ТАВРИЧЕСКИЙ ВЕСТНИК ИНФОРМАТИКИ И МАТЕМАТИКИ

№ 1 ' 2008

МЕЖДУНАРОДНОЕ НАУЧНО-ТЕОРЕТИЧЕСКОЕ ИЗДАНИЕ
КРЫМСКИЙ НАУЧНЫЙ ЦЕНТР НАЦИОНАЛЬНОЙ АКАДЕМИИ НАУК
И МИНИСТЕРСТВА ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ
ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. ВЕРНАДСКОГО

ОСНОВАН В 2002 ГОДУ

Свідоцтво про державну реєстрацію друкованого засобу масової інформації

КВ №7826 від 04.09.2003

Згідно до постанови ВАК України від 30 червня 2004 р. 3—05/7, перелік №4,журнал "Таврійський вісник информатики та математики" внесено до переліку журналів ВАК України, у яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів кандидата й доктора наук за спеціальностями "Теоретичні основи інформатики та кібернетики", "Математичне моделювання та обчислювальні методи", "Математичне і програмне забезпечення обчислювальних машин і систем", "Системний анализ і теорія оптимальних рішень".

**КРЫМСКИЙ НАУЧНЫЙ ЦЕНТР НАЦИОНАЛЬНОЙ АКАДЕМИИ НАУК
И МИНИСТЕРСТВА ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ
ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. ВЕРНАДСКОГО**

АДРЕС РЕДАКЦИИ:
Крымский научный центр Национальной Академии наук
и Министерства образования и науки Украины
Украина, Крым, г. Симферополь, пр-т Вернадского, 2, 95007

ДЛЯ ПЕРЕПИСКИ:
факультет математики и информатики ТНУ
Украина, Крым, г. Симферополь, пр-т Вернадского, 4, 95007

Тел. гл.редактора: (0652) 63-75-42
Тел. редакции: (0652) 602-466
e-mail (гл.редактор): donskoy@ccssu.crimea.ua
e-mail (для переписки): twim_taurida@mail.ru
сайт журнала: www.twim.crimea.edu

**Журнал публикует оригинальные и обзорные статьи
по вопросам теоретической и прикладной информатики и математики**

Ведущие тематические разделы:

Функциональный анализ и его приложения

Интегральные, дифференциальные уравнения и динамические системы

Нелинейный анализ и его применение

Спектральные и эволюционные задачи

Математические проблемы гидродинамики

Дискретная оптимизация

Математическая логика, теория алгоритмов и теория сложности вычислений

Математические модели и методы прогнозирования

Машинное обучение и извлечение закономерностей

Дедуктивные системы и базы знаний

Знаниеориентированные и гибридные математические модели принятия решений

Синтез моделей принятия решений при неполной начальной информации

Вычислительная математика

Математическая теория, алгоритмы и системы распознавания образов

**Печатается по решению научно технического Совета
КНЦ НАН и Министерства образования и науки Украины
Протокол №7 от 15 мая 2008 г.**

Содержание

УДК 519.23

ZERO-INFLATED BOOSTED ENSEMBLE FOR SMALL COUNT PROBLEM

© Borisov A.E., Tuv E.V.

Intel corporation
Nizhnhy Novgorod, Russia

e-mail: alexander.borisov@intel.com, tuv.eugene@intel.com

Abstract. The article introduces a new approach for modeling "small count data" where distribution of the response variable is assumed to follow the zero-inflated Poisson (ZIP) model. ZIP model based on boosted ensemble is introduced. It combines and extends ZIP tree and gradient boosting tree (GBT) methods. Our algorithm, called ZIP-GBT, is at first introduced from theoretical perspective in the framework of Friedman's gradient boosting machine. Then it is compared empirically on two real data sets and two artificial data sets versus single tree approach (ZIP-tree). It is shown that ZIP-GBT outperforms ZIP tree in most cases both in terms of cross validated ZIP-likelihood and ZIP distribution parameters prediction.

Introduction

The analysis of count data is the primary interest in many areas including public health, epidemiology, sociology, psychology, engineering, and agriculture. Poisson distribution is typically assumed to model the distribution of the rare event counts. The Poisson regression model is commonly used to explain the relationship between the count (non-negative integer) response and input variables (predictors). However, it is often the case that the outcome of interest contains excess number of zeros which cannot be explained correctly by the standard Poisson model.

Lambert [1] successfully proposed a mixture of the distribution with a point mass at zero and a Poisson distribution, called zero-inflated Poisson (ZIP) regression, to handle zero-inflated count data in a number of defects in a manufacturing process. After Lambert [1] successfully introduced the zero-inflated Poisson (ZIP) model, many extensions or modified ZIP models were elaborated. For example Wang [18] proposed Markov zero-inflated Poisson regression (MZIP), Li et.al [3] introduced multivariate ZIP models, Lee and Jin [2] proposed a tree-based approach for Poisson regression, Chiogna and Gaetan [11] used semi-parametric ZIP in animal abundance studies, Hsu [13] proposed a weighted ZIP, and Famoye and Singh [12] used zero-inflated generalized Poisson (ZIGP) regression model when the count data is over-dispersed. ZIP regression is not only applied in the manufacturing, but it is also widely used in many other areas such as public health, epidemiology, sociology, psychology, engineering, agriculture, etc. ([17], [16], [14], [10], [19]).

In data mining, tree-based model is one of the most popular and common methods used for approximating target functions, in which a function can be learned by splitting the data set into subsets based on an response-attribute value test. This process is repeated on each derived subset in a recursive manner and is represented by a tree model. Each terminal node is assigned a response value. A popular method of tree-based regression and

classification is called CART (Classification and Regression Tree) [9,4]. In 2006, Lee and Jin [2] introduced ZIP-tree model. They modified CART algorithm splitting criteria by using the zero-inflated Poisson (ZIP) likelihood error function instead of residual sum of squares. Each terminal node of ZIP tree is assigned its own ZIP distribution parameters (zero inflation probability p and Poisson distribution parameter λ).

Further development of the idea of using trees for ZIP regression leads to using a tree ensemble instead of a single tree. Ensemble methods are very popular in literature and widely used in practice, especially parallel (Random Forest, or RF, see [7,8]) and boosted tree ensembles (like AdaBoost or GBT[20,21]). Tree ensembles are shown to have smaller prediction error (bias) than a single tree; parallel ensembles (RF) also offer more stability (smaller variance).

In this paper, we propose a boosted ensemble approach similar to GBT that fits ZIP distribution parameters p, λ using two tree ensembles. The algorithm minimizes ZIP log-likelihood loss function by gradient descent method similar to the one proposed by Friedman for multi-class logistic regression (MCLRT) [21]. Our algorithm uses the log-link function for λ and the logit-link function for p as proposed in [1] for standard ZIP regression.

1. Previous work : ZIP regression and ZIP tree

Lambert [1] used ZIP distribution for response variable y, where Poisson distribution parameters depend on the values of input variables:

$$y_i \sim \begin{cases} 0, \text{ with probability } p_i, \\ Poisson(\lambda_i), \text{ with probability } 1-p_i, \end{cases} \quad i = 1 \dots n,$$

where n is the number of samples. This model implies that

$$P(y_i = k) = P(p_i, \lambda_i, k) = \begin{cases} p_i + (1-p_i)e^{-\lambda_i}, k = 0, \\ (1-p_i)e^{-\lambda_i}\lambda_i^k/k!, k = 1, 2, \dots, \end{cases}$$

where parameters λ, p are obtained from the linear combinations of inputs via log- and logit-link functions :

$$\log(\lambda_i) = \beta x_i, \text{ and } \log it(p_i) = \log\left(\frac{p_i}{1-p_i}\right) = \gamma x_i.$$

Here x is input feature vector (for simplicity of notation we always assume that a "dummy" variable $x = 1$ is added as the first input variable to take the intercept term into account), β, γ are vectors of coefficients (same for each data point) to be fit. The ZIP model is usually fitted using the maximum likelihood estimation method.

Log-likelihood can be maximized using Newton-Raphson method, but usually EM algorithm is used because it is more robust and computationally simpler. The authors study the behavior of their algorithm on AT&T Bell Labs soldering data. Another article[6] applies the same ZIP regression model to DMFT(decayed, missing and filled teeth) data. They use piecewise constant model for p parameter. Both articles consider using mixture models (most popular is mixture of Poisson and negative binomial distributions), but

both authors claim that such models are more difficult to fit and usually provide worse predictions than ZIP models.

In 2006, Lee and Jin [2] used the ZIP likelihood as new splitting criteria for decision tree. They modified CART (classification and regression tree) algorithm by using negative ZIP likelihood as an impurity measure in a node. Negative ZIP likelihood of the data in node T can be expressed as

$$L_{ZIP}(T) = L_{ZIP}(p, \lambda, y) = -n_0 \cdot \log\left(p + (1-p)e^{-\lambda}\right) - (n - n_0) \cdot (\log(1-p) - \lambda) - \\ - \sum_{x_i \in T} y_i \cdot \log\lambda + \sum_{x_i \in T, y_i > 0} \log(y_i!),$$

where p, λ are estimates of Poisson distribution parameters in node T.

The new splitting criterion is based on the difference of the ZIP likelihood in the left-child node and the right-child node from the ZIP likelihood in the parent node. The expression for split weight can be written as $\phi(s, T) = L_{ZIP}(T) - L_{ZIP}(T_L) - L_{ZIP}(T_R)$, where T is the parent node, T_L, T_R are left and right children of T, s is the split in node T. The same best split search strategy can be applied as in CART.

Parameter λ for a tree node is estimated using zero-truncated Poisson distribution:

$$\frac{\lambda}{1 - e^{-\lambda}} = \bar{y} = mean(y_i | y_i > 0, x_i \in T).$$

After λ parameter is obtained, p can be estimated from the known proportion of zero-class samples in the node :

$$p = \frac{n_0/n - e^{-\lambda}}{1 - e^{-\lambda}},$$

where n_0 is the number of zero count samples and n is total number of samples.

2. Boosting framework and ZIP boosted ensemble

Trees usually provide robust models for complex target functions (not limited by linearity assumptions) and are not sensitive to noise and outliers. They also allow working with mixed type data (both numeric and categorical predictors) and handle missed values in a natural way. CART trees are also very fast to fit (they do not require complex matrix operations as MLE problem). That is why trees are widely used in real life applications where data sets are mixed-type, large in number of samples and predictors, and noisy (both input variables and the response).

However, a single tree often has low predictive power (especially if an underlying target model is complex and multivariate) and is not stable to small fluctuations in the data. So different authors proposed using ensembles of trees for regression and classification problems (L. Breiman introduced parallel ensembles, or Random Forests [7,8], and J.H. Friedman introduced boosted ensembles [20, 21]). Ensembles have much higher predictive accuracy and generalization ability while keeping all advantages of the single tree. So ensemble methods become more and more popular as "off-the-shelf" approach and often provide as good results as best state-of-art methods.

Random Forest, a parallel ensemble, is a set of trees, with each of the trees build on a different (random) subsample of training data. In each node when searching for best split

only a small subset of input variables is selected randomly. Prediction from a set of trees is obtained using averaging prediction over trees in regression or voting in classification.

Gradient boosting, in its general form, constructs an additive regression (or logistic regression) model by sequentially fitting a simple parameterized function (a base learner that can be a tree or any other model) to current "pseudo-residuals" at each iteration. The pseudo-residuals are the gradient of the loss functional minimized with respect to the current parameter values, with respect to the model values at each training data point evaluated at the current step. Let's describe gradient boosting framework more formally.

Suppose we have a training sample $\{y_i, x_i\}_{i=1...n}, x_i = \{x_{i1}, \ldots, x_{im}\} \in X, y_i \in Y$, where n is the number of samples, m is the number of input variables. Our goal is to find a function $F^*(x) : X \to Y$ that minimizes expected value of the specified loss function $L(y, F(x))$ over the joint distribution of x, y values :

$$F^*(x) = \arg\min_{F(x)} E_{x,y} L(y, F(x)).$$

Here the expectation term cannot usually be computed directly as the joint distribution of x, y is not known. So in practice it is replaced with expected risk, i.e. :

$$F^*(x) = \arg\min_{F(x)} \sum_{i=1}^{n} L(y_i, F(x_i)).$$

Boosting uses an additive model to approximate $F^*(x)$: $F^*(x) = \sum_{m=0}^{M} h(x, a_m)$, where function $h(x, a_m)$ is some simple function ("base learner") of parameter vector a. Base learner parameters $a_m, m = 1 \ldots M$ are fit in forward stepwise manner. It starts from some initial approximation $F_0(x)$, then proceeds as follows :

$$a_m = \arg\min_{a} \sum_{i=1}^{n} L\left(y_i, F_{m-1}(x_i) + h(x_i, a)\right), \tag{1}$$

$$F_m(x) = F_{m-1}(x) + h(x, a_m).$$

Gradient boosting solves optimization problem (1) using the stepwise steepest descent method. The function $h(x, a)$ is fit by least squares:

$$a_m = \arg\min_{a} \sum_{i=1}^{n} (\tilde{y}_{im} - h(x_i, a))^2$$

to the current "pseudo-residuals" or "pseudo-response" :

$$\tilde{y}_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}. \tag{2}$$

Gradient tree boosting is the specialization of this approach to the case where base learner is a CART regression tree :

Algorithm 1 : Gradient tree boosting

1. $F_0(x) = \arg\min_{\gamma} \sum_{i=1}^{n} L(y_i, \gamma)$
2. For $m = 1$ to M do:
3. $\tilde{y}_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}, i = 1 \dots n$
4. $\{R_{lm}\}_{l=1\dots L} = L-$ terminal node tree
5. 5. $\gamma_{lm} = \arg\min_{\gamma} \sum_{x_i \in R_{lm}} L(y_i, F_{m-1}(x_i) + \gamma)$
6. $F_m(x) = F_{m-1}(x) + \nu \cdot \gamma_{lm} I(x \in R_{lm})$
7. End for

Here γ_{lm} is the response (mean) in node R_{lm}. Parameter ν is "shrinkage rate" or "regularization parameter" that controls learning rate of the algorithm. Smaller values for shrinkage (0.01-0.1) have proven to reduce over-fitting, thus allowing building models with better generalization ability. Usually only random part of samples (about 60%) are used to learn a tree on step 4 (bootstrapping). This speeds up model building and also reduces over-fitting.

A particularly interesting case of the algorithm 1 is a two-class logistic regression (that also has multi-class generalization that we will omit). It is derived from gradient tree boosting framework when using CART tree as a base learner, and negative binomial log-likelihood as the loss function.

Assume that response is binary, $y \in \{-1, 1\}$, and the loss function is negative binomial log-likelihood : $L(y, F) = \log(1 + \exp(-2yF))$, where F is a two-class logistic transform : $F(x) = \frac{1}{2}\log\left[\frac{\Pr(y=1|x)}{\Pr(y=-1|x)}\right]$. So each tree approximates log-odd of class 1 probability, and the pseudo-response derived from formula (2) or step 3 of Algorithm 1 is $\tilde{y}_{im} = 2y_i/(1 + \exp(2y_i F_{m-1}(x_i))$.

Optimization problem on step 5 cannot be solved in closed form, so single Newton-Raphson step approximation is used :

$$\gamma_{lm} = \sum_{x_i \in R_{jm}} \tilde{y}_{im} = \sum_{x_i \in R_{jm}} |\tilde{y}_{im}| \cdot (2 - |\tilde{y}_{im}|) \tag{3}$$

To increase the robustness of GBT algorithm, influence trimming can be applied when selecting samples for building a subsequent tree. Suppose we want to estimate the response in a terminal node on step 5 of Algorithm (1) via equation

$$\sum_{x_i \in R_{lm}} \partial L(y_i, F_{m-1}(x_i) + \gamma)/\partial\gamma = 0.$$

Influence of i-th sample on the solution can be gauged by the second derivative of the loss function, i.e

$$w_i = w(x_i) = \partial^2 L(y_i, F_{m-1}(x_i)+\gamma)/\partial\gamma^2|_{\gamma=0} = \partial^2 L(y_i, f)/\partial f^2|_{f=F_{m-1}(x_i)} = |\tilde{y}_{im}|\cdot(2-|\tilde{y}_{im}|).$$

When building subsequent tree, we omit all observations with $w_i < w_{l(\alpha)}$, where $l(\alpha)$ is the solution to $\sum_{i=1}^{l(\alpha)} w_{(i)} = \alpha \cdot \sum_{i=1}^{N} w_i$ (here weights $w_{(i)}$ are w_i's sorted in ascending order), and α is usually chosen in [0.05,02] range.

Influence trimming not only speeds up the tree construction, but also improves robustness of Newton-Raphson method step (equation 4), preventing small denominator values for a tree node, because denominator is proportional to the sum of sample influences in the node. Influence trimming

Now we are ready to derive our own algorithm for the ZIP regression problem using negative ZIP-likelihood as a loss function, and CART model as a base learner. We use two ensembles of trees to approximate transformed Poisson distribution parameters (p, λ). We use the same transformation (link function) as used by Lambert[1], i.e log-link for p and logit-link for λ :

$$\mu = \log\left(p/(1-p)\right), p = e^{\mu}/(1+e^{\mu}),$$
$$\nu = \log(\lambda), \lambda = e^{\nu}.$$

So the first ensemble fits model for $\mu(x)$, the second one – for $\nu(x)$. Initial value for ν is estimated from zero truncated Poison distribution of the response :

$$\frac{\lambda_0}{1-e^{-\lambda_0}} = \bar{y} = mean(y_i|y_i > 0), \nu_0 = \log(\lambda_0),$$

then μ_0 as

$$p_0 = \frac{n_0/n - e^{-\lambda_0}}{1-e^{-\lambda_0}}, \mu_0 = \log it(p_0),$$

where n_0 is the number of zero-class samples $(y_i = 0)$.

The loss function to be minimized takes the form

$$L(y,p,\lambda) = \sum_{i=1}^{n} L(y_i, p_i, \lambda_i) = -\sum_{y_i=0} \log\left(p_i + (1-p_i)e^{-\lambda_i}\right) -$$
$$-\sum_{y_i>0}\left(\log(1-p_i) - \lambda_i\right) - \sum_{y_i>0} y_i \log\lambda_i + \sum_{y_i>0}\log(y_i!),$$

where we denoted $p_i = p(x_i), \lambda_i = \lambda(x_i)$ to simplify notation. Last term is not dependent on the model and can be dropped. In other terms,

$$L(y,p,\lambda) = L(y,\mu,\nu) = \sum_{i=1}^{n} L(y_i,\mu_i,\nu_i) =$$
$$= -\sum_{y_i=0}\left(\log(e^{\mu_i} + \exp(-e^{\nu_i}) - \log(1+e^{\mu_i})\right) + \sum_{y_i>0}\left(\log(1+e^{\mu_i}) + e^{\mu_i}\right) - \sum_{y_i>0} y_i\nu_i,$$

where $\mu_i = \mu(x_i), \nu_i = \nu(x_i)$.

Pseudo-responses are calculated as follows. Pseudo-response for p-ensemble is

$$\tilde{\mu}_{im} = -\left[\frac{\partial L(y_i,\mu_i,\nu_i)}{\partial\mu_i}\right]_{\mu_i=\mu_{m-1}(x),\nu_i=\nu_{m-1}(x)} =$$

$$= \left[\frac{\partial L(y_i,p_i,\lambda_{m-1}(x_i))}{\partial p_i}\right]_{p_i=p_{m-1}(x)} \cdot \left[\frac{\partial p(\mu_i)}{\partial\mu_i}\right]_{\mu_i=\mu_{m-1}(x)} =$$

$$= \begin{cases} \frac{(e^{-\lambda_i}-1)}{p_i+(1-p_i)e^{-\lambda_i}} \cdot p_i(1-p_i), & y_i = 0, \\ 1/(1-p_i) \cdot p_i(1-p_i) = p_i & y_i > 0, \end{cases}$$

where $p_i = p_{m-1}(x_i), \lambda_i = \lambda_{m-1}(x_i)$. Here pseudo-response is expressed in terms of p_i, λ_i to simplify notation.

Pseudo-response for λ-ensemble is derived in the same way (note that $\frac{\partial p(\mu)}{\partial \mu} = e^{\mu}/(1+e^{\mu})^2 = p(1-p)$, $\frac{\partial \lambda(\nu)}{\partial \nu} = e^{\nu} = \lambda$):

$$\tilde{\nu}_{im} = \begin{cases} \lambda_i e^{-\lambda_i} / \left(p_i/(1-p_i) + e^{-\lambda_i}\right), & y_i = 0, \\ \lambda_i - y_i, & y_i > 0. \end{cases}$$

Then node response optimization problem on step 5 of algorithm 1 is solved via single step of Newton-Raphson as in Friedman's two class logistic regression. Unfortunately in our case Hessian (second derivative) can be negative sometimes, although such occasions are rare and possibly indicate over-fitting or "self-contradictory" data i.e a case when data points with similar x values have very different (p, λ) values. Negative Hessian means that the target function is not concave and thus cannot be approximated by 2-nd order polynomial. In such case we use one step of steepest descent instead of Newton-Raphson step. Second derivatives for p-tree (which are summands in denominator in formula (4)) are:

$$\partial^2 L(y_i, \mu_{m-1}(x_i) + \gamma, \nu_{m-1}(x_i))/\partial \gamma^2|_{\gamma=0} = \partial^2 L(y_i, \mu_i, \nu_{m-1}(x_i))/\partial \mu_i^2|_{\mu_i = \mu_{m-1}(x_i)} =$$
$$= \tilde{\tilde{\mu}}_{im} = \begin{cases} -\frac{(1-p_i)^2 e^{-\lambda_i} - p_i^2}{(p_i+(1-p_i)e^{-\lambda_i})^2} \cdot p_i(1-p_i) \cdot (1-e^{-\lambda_i}), & y_i = 0, \\ p_i(1-p_i), & y_i > 0. \end{cases}$$

Same for λ-tree:

$$\partial^2 L(y_i, \mu_{m-1}(x_i), \nu_{m-1}(x_i) + \gamma)/\partial \gamma^2|_{\gamma=0} = \partial^2 L(y_i, \mu_{m-1}(x_i), \nu_i)/\partial \nu_i^2|_{\nu_i = \nu_{m-1}(x_i)} =$$
$$= \tilde{\tilde{\nu}}_{im} = \begin{cases} \lambda_i(1-p_i) \cdot \frac{1-p_i+p_i e^{\lambda_i} \cdot (1-\lambda_i)}{(p_i+(1-p_i)e^{\lambda_i})^2}, & y_i = 0, \\ \lambda_i, & y_i > 0. \end{cases}$$

The formula (4) for "optimal" response in p-tree terminal node will look like ($n(R_{jm})$ is the count of training samples in node R_{jm}):

$$\gamma^1_{lm} = \begin{cases} \sum_{x_i \in R_{jm}} \tilde{\mu}_{im} / \sum_{x_i \in R_{jm}} \tilde{\tilde{\mu}}_{im}, \sum_{x_i \in R_{jm}} \tilde{\tilde{\mu}}_{im} > \varepsilon = 10^{-6}, \\ \sum_{x_i \in R_{jm}} \tilde{\mu}_{im} / n(R_{jm}), \text{otherwise.} \end{cases}$$

same for λ-tree :

$$\gamma^2_{lm} = \begin{cases} \sum_{x_i \in R_{jm}} \tilde{\nu}_{im} / \sum_{x_i \in R_{jm}} \tilde{\tilde{\nu}}_{im}, \sum_{x_i \in R_{jm}} \tilde{\tilde{\nu}}_{im} > \varepsilon, \\ \sum_{x_i \in R_{jm}} \tilde{\nu}_{im} / n(R_{jm}), \text{otherwise.} \end{cases}$$

There are several tricks that we use to improve the numeric stability of the algorithm. To prevent μ_i, ν_i from causing numerical overflow or underflow we simply threshold them by a reasonable constant ($log(FLT\_MAX/2)$ for example). We also adopted influence trimming strategy to prevent very small Hessian by absolute value in a tree node. We found that one cannot remove samples with negative loss function second derivative because it can harm severely the performance of the algorithm. However one can trim samples with

second derivative small by absolute value in p-tree. So we do no influence trimming for λ-tree (as small absolute value of the second derivative of the loss function is not likely to happen there), and do influence trimming with weights $w_i = p_i(1 - p_i)$ for p-tree in the same way as it is described earlier for two-class logistic regression.

3. Evaluation

First we validate our algorithm and compare its performance with our implementation of ZIP tree on two artificial data sets. Both data sets are generated from a known model for ZIP distribution parameters (p, λ) with a small amount of random noise added, i.e

$$p = p(x_1, x_2) \cdot (1 + \varepsilon \cdot u_1), u_1 \in U(-1, 1),$$
$$\lambda = \lambda(x_1, x_2) \cdot (1 + \varepsilon \cdot u_2), u_2 \in U(-1, 1).$$

Then response value y_i is generated from ZIP distribution with parameters $(p_i = p(x_{1i}, x_{2i}), \lambda_i = \lambda(x_{1i}, x_{2i}))$. In all three experiments three values for noise level $\varepsilon = 0, 0.2, 0.5$ are used.

The first data set uses linear model for (p, λ) :

$$p = 0.2 + 0.6 \cdot (0.3x_1 + 0.7x_2),$$
$$\lambda = 1.5 + 7 \cdot (0.6x_1 + 0.4x_2).$$

The second data set uses more complex highly nonlinear model

$$\text{logit}(p) = 2\sin(20x_1) + 3x_2 \cdot (x_2 - 0.5),$$
$$\log(\lambda) = \sin(30x_1) + 3x_2.$$

For each model we report the base error (error for the best constant model), training error, and cross-validation error (5-fold), where error is average negative ZIP log-likelihood, and average absolute difference (on the training set) between "true" and "predicted" parameters (p, λ), we also report average relative difference for λ parameter. Three last numbers show how well ZIP distribution parameters are approximated by the model. In all experiments the model complexity (which is the pruning step for the tree and the number of iterations for GBT) is selected using best CV error. Size of all data sets is 10000 samples.

For artificial data sets, the following parameters are used :
ZIP TREE : tree_depth = 6, min_split = 50, min_bucket = 20.
ZIP GBT : nit = 1000, tree_depth = 3, min_split = 400, min_bucket = 200, shrinkage = 0.01, infl_trimming = 0.1.
Here tree_depth is a maximum tree depth (node is not split if it is at the specified depth), min_split is a minimum size of the node that will be split (if it has less observations it is NOT split), min_bucket is a minimum size of the terminal node (split is not accepted if it creates a terminal node with smaller size), nit = is a maximum number of iterations for an ensemble, shrinkage is the ν parameter (regularization) on step 6 of Algorithm 1, infl_trimming is α threshold for influence trimming.

Base error column in the following table shows negative ZIP log-likelihood for the best constant model, train and CV-error are train and 5-fold cross-validation errors (negative ZIP log-likelihood also), δp is average absolute difference in predicted

p parameter ($\delta p = \sum_{i=1}^{n} |p(x_{1i}, x_{2i}) - \hat{p}(x_{1i}, x_{2i})|/n$ where $\hat{p}(x_1, x_2)$ is prediction from the model), $\delta\lambda$ is an average absolute difference in predicted λ parameter ($\delta\lambda = \sum_{i=1}^{n} |\lambda(x_{1i}, x_{2i}) - \hat{\lambda}(x_{1i}, x_{2i})|/n$), $\delta\lambda_{rel}$ is an average relative difference in predicted λ parameter ($\delta\lambda_{rel} = \sum_{i=1}^{n} |1 - \hat{\lambda}(x_{1i}, x_{2i})/\lambda(x_{1i}, x_{2i})|/n$).

Table 1. Comparison of ZIP tree and ZIP GBT on two artificial data sets.

Data	Noise(ε)	Base error	Model	Train error	CV error	δp	$\delta\lambda$	$\delta\lambda_{rel}$	Best step
LINEAR	0	1.801	TREE	1.663	1.690	0.043	0.355	0.074	16
			GBT	1.653	1.675	0.027	0.182	0.038	413
	0.2	1.859	TREE	1.707	1.736	0.043	0.416	0.092	15
			GBT	1.702	1.721	0.032	0.179	0.040	284
	0.5	1.873	TREE	1.744	1.775	0.040	0.441	0.093	13
			GBT	1.733	1.754	0.032	0.234	0.049	319
NON-	0	2.920	TREE	1.535	1.675	0.146	3.105	0.403	49
LINEAR			GBT	1.360	1.413	0.058	1.844	0.255	999
	0.2	3.037	TREE	1.594	1.735	0.156	3.027	0.423	35
			GBT	1.425	1.492	0.064	1.810	0.247	999
	0.5	3.310	TREE	1.774	1.925	0.154	3.112	0.394	42
			GBT	1.577	1.663	0.073	1.812	0.253	998

This table shows that GBT is always superior to a single tree in terms of train error, CV error and ZIP distribution parameters prediction error. One can see that over-fitting (difference between CV and train errors) is much smaller for GBT, especially for bigger noise levels and more complex models.

Then we compared performance of ZIP GBT to ZIP tree on two public available real-life data sets. The first one is SOLDER, which is a part of rpart R free package, the second is DMFT (decayed, missing and filled teeth) data set used in [6]. On SOLDER data set, ZIP GBT is much better than a single tree in terms of cross-validated log-likelihood, on DMFT GBT is only slightly better. Parameters of both algorithms were adjusted manually to minimize cross-validation error :
ZIP TREE : tree_depth = 6, min_split = 15, min_bucket = 10.
ZIP GBT : nit = 1000, tree depth = 3, min_split = 30, min_bucket = 20, shrinkage = 0.02(0.005 for DMFT), infl_trimming = 0.1.

It can be seen that GBT has much smaller CV error on SOLDER data set and a little smaller on DMFT data set.

Conclusion

This article introduces gradient boosting model for small-count regression problem, where response is assumed to follow ZIP distribution. This model uses gradient tree boosting concepts introduced by Friedman for regression and classification and extends

Table 2. Comparison of ZIP tree and ZIP GBT on real-life data.

Data	Base error	Model	Train error	CV error	Best step
SOLDER	4.464	TREE	2.493	2.714	9
SOLDER	4.464	GBT	1.510	1.818	765
DMFT	1.789	TREE	1.525	1.577	8
DMFT	1.789	GBT	1.499	1.564	660

them to ZIP model. It is shown that the algorithm performance (both in terms of log-likelihood value and prediction of ZIP distribution parameters as function of inputs) is superior to the performance of ZIP tree.

The algorithm can be adapted to different problems using different link functions. Further analysis of the algorithm performance and comparison to other small count data models on large real-life data that comes from Intel manufacturing processes is of great interest.

References

1. *D. Lambert.* Zero-inflated Poisson regression with an application to defects in manufacturing// Technometrics, 34(1), pp. 1-14, 1992.
2. *S. Lee and S. Jin.* Decision tree approaches for zero-inflated count data// Journal of applied statistics, 33(8), pp. 853-865, 2006.
3. *C. Li, J. Lu, and J. Park.* Multivariate zero-inflated Poisson models and their applications// Technometrics, 41(1), pp. 29-38, 1999.
4. *Hastie, R. Tibshirani, and J. Friedman.* The Elements of Statistical Learning. Springer, New York , 2001.
5. *D. Bohning, E. Dietz, P. Schlattman, L. Mendonca, and U. Kirchner.* Testing parameter of the power series distribution of a zero inflated power series model// Statistical Methodology, 4, pp. 393-406, 2007.
6. *D, Bohning, E, Dietz ,P, Schlattman, L, Medonca and U,Kirchner.* The zero-inflated Poisson model and the decayed, missing and filled teeth index in dental epidemiology// J.R.Statist.Soc A(1999) 162, part 2, pp.195-209.
7. *L. Breiman.* Bagging predictors// Machine Learning, 24, pp. 123-140, 1996.
8. *L. Breiman.* Random forests// Machine Learning, 45, pp. 5-32, 2001.
9. *L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone.* Classification and regression trees. Chapman and Hall/CRC, New York, 1998.
10. *Y.B. Cheung.* Zero-inflated models for regression analysis of count data: a study of growth and development// Statistics in Medicine, 21, pp. 1461-1469, 2002.
11. *M. Chiogna and C. Gaetan.* Semiparametric zero-inflated poisson models with application to animal abundance studies// Environmetrics, 18, pp. 303-314, 2007.
12. *F. Famoye and K.P. Singh.* Zero-inflated generalized Poisson regression model with an application to domestic violence data.// Journal of Data Science, 4, pp. 117-130, 2006.
13. *C. Hsu.* A weighted zero-inflated poisson model for estimation of recurrence of adenomas// Statistical Methods in Medical Research, 16, pp. 155-166, 2007.
14. *K. Hur, D. Hedeker, W. Henderson, S. Khuri, and J. Daley.* Modeling clustered count data with excess zeros in health care outcomes research// Health Services and Outcomes Research Methodology, 3, pp. 5-20, 2002.

15. *C. Li, J. Lu, and J. Park.* Multivariate zero-inflated Poisson models and their applications// Technometrics, 41(1), pp. 29-38, 1999.
16. *C.S. Li, J.C. Lu, J. Park, K. Kim, P.A. Brinkley, and J.P. Peterson.* Multivariate zero-inflated Poisson models and their applications. // Technometrics, 41(1), pp. 29-38, 1999.
17. *R. Ramis Prieto, J. Garcia-Perez, M. Pollan, N. Aragones, B. Perez-Gomez, and G. Lopez-Abente.* Modelling of municipal mortality due to haematological neoplasias in Spain // Journal of epidemiology and community health, 61(2), pp. 165-171, 2007.
18. *P. Wang.* Markov zero-inflated poisson regression models for a time series of counts with excess Zeros// Journal of Applied Statistics, 28(5), pp. 623-632, 2001.
19. *K. K. Yau and A.H. Lee.* Zero-inflated poisson regression with random effects to evaluate an occupational injury prevention programme// Statistics in Medicine, 20(19), pp. 2907-2920, 2001.
20. *J.H. Friedman.* Greedy function approximation : a gradient boosting machine// Technical report, Dept. of Statistics, Stanford University, 1999.
21. *J.H. Friedman.* Stochastic gradient boosting// Computational Statistics and Data Analysis, 38(4), pp. 367-378, 2002.

Статья поступила в редакцию 25.04.2008

УДК 681.3: 519.68

TIME SERIES CLASSIFICATION THROUGH HETEROGENEOUS FEATURE SELECTION

© Martyanov V.Ju., Eruhimov V.L.

Intel corporation
Nizhnhy Novgorod, Russia

e-mail: *Vladimir.Martyanov@intel.com*

Abstract. The paper investigates a generic method of time series classification. A heterogeneous set of features is extracted from each signal, including statistical moments, wavelets, Chebyshev polynomials, PCA, and DTW-based features. An ensemble of boosted trees is learned on a subset of this set of features. Particle filtering is used to choose a good feature subset and parameters of the learning engine based on cross-validation error.

Introduction

The problem of time series classification (TSC) has experienced a burst in the number of publications during the last decade. Various domains such as computer vision, medicine, biology, manufacturing and many others generate an enormous amount of signal data that can be used to segment or predict important events such as human actions or diseases or manufacturing tool malfunctions. There is a strong need for a generic TSC engine that, having small or no knowledge about the domain, can learn to classify signals from labeled samples. About a decade ago, [4] introduced a measure called Dynamic Time Warping (DTW) that is based on matching two signals with dynamic programming. Together with one-nearest-neighbor (1NN) it remains a competitive method for time series classification [22]. A large group of papers is devoted to extracting generic features from signals and transforming a TSC problem into a classical machine learning problem of predicting signal class from a given feature set. A list of features includes SVD (Singular Value Decomposition) features [19], DFT (Discrete Fourier Transform) [2], coefficients of the decomposition into Chebyshev Polynomials [8], DWT (Discrete Wavelet Transform) [9, 21], PLA (Piecewise Linear Approximation) [17], ARMA (AutoRegression Moving Average) coefficients [10], various symbolic representations [15, 13, 20]. An excellent review of the TSC techniques is given in [16].

Each of the methods has its own faults. Euclidean/DTW based methods suffer from the curse of dimensionality – 1-NN is known to perform poorly on high-dimensional problems (i.e. long signals) [14]. Generic feature extraction methods balance between low-dimensional signal representations that have less information and thus often possess lower predictive capabilities, and high dimensional representations that are hard to learn from (although here we are not restricted to 1-NN, as in the case of Euclidean/DTW distance). Also, each generic feature is salient for only a subset of TSC problems – each TSC problem has its own classification-optimal set of features.

Recent advances in feature selection methods [5, 1] enable us to dicuss the following generic approach to a TSC problem. We pick several generic feature sets – statistical moments, wavelets, Chebyshev coefficients, PCA coefficients, and the original values of

signal. Then we run Gradient Boosting Trees (GBT) [11, 12] with imbedded feature weighting scheme. We show that although the extracted features supplement each other and removal of one feature types may result in a considerably large increase of test error for some TSC tasks, due to limited number of training samples, learning the model on an appropriate feature subset (suited to this particular TSC dataset) can be advantageous, as compared to learning it on all available features. We show that Particle Filtering (PF) can be successfully used to select a subset of features and simultaneously optimize model parameters. The results are obtained on publicly available UCR datasets [18].

1. Machine Learning on Massive Sets of Features

This section provides a description of Gradient Boosting Trees that we used for supervised learning, and a feature selection algorithm that we used to reduce the dimensionality of the learning problem.

1.1. **Gradient Boosting Trees.** Gradient Boosting Trees (GBT) [11], [12] has been proven to be among the most accurate and versatile state-of-the-art learning machines. GBT is an iteratively learned serial ensemble where every new tree is fitted to the generalized residuals of the current ensemble. GBT builds shallow trees using all variables (on a subsample of the training data), and hence, it can handle large datasets with a moderate number of inputs. A modification of GBT [5] suggests a different ensemble learning strategy so that processing of very high dimensional datasets is feasible with almost no loss in prediction accuracy.

Our implementation of GBT is very close to [12] with feature weighting [5] on top of it. Each tree was trained on a randomly chosen 60% portion of the training dataset, the probability threshold was equal to 0.5.

1.2. **GBT with dynamic feature weighting.** The main idea of this approach is to apply the variable-sampling technique used by Random Forest [6] for high-dimensional problems. We sample with replacement a small subset S from a set M of all input variables but keep only distinct elements. We sample S for each split in each tree independently and a standard greedy optimization is used to select a feature from S to split on. However, as shown in the [5], the uniform sampling used in the Random Forest could cause a significant performance degradation for sequentially boosted trees. We sample with probabilities that depend on feature importance and both are updated as a new tree is added to the ensemble. The sampling probability for a variable x_i at iteration l (where each iteration corresponds to learning a new tree so that at iteration l we already have an ensemble of l trees) is proportional to the corresponding weight

$$p(x_i, l) = w(x_i, l) / \sum_j w(x_j, l) \tag{1}$$

Weights have two components

$$w(x_i, l) = I(x_i, l) + SVI(x_i, l), \tag{2}$$

where $SVI(x_i, l) = \sum_{j=1}^{l} VI(x_i, j)$, $VI(x_i, j)$ is the influence of the i-th variable in the j_{th} tree:

$$VI(x_i, T) = \sum_{t \in T} \Delta I(x_i, t) \tag{3}$$

where $\Delta I(x_i, t) = I(t) - p_L I(t_L) - p_R I(t_R)$ is the decrease in impurity due to an actual (or potential) split on variable x_i at a node t of the tree T [7]. $I(x_i, l)$ is a contribution of the initial influence $I(x_i, 0)$ for the i_{th} variable at the l_{th} iteration. We used exponentially decreasing initial influences

$$I(x_i, l) = I(x_i, 0) \cdot (1 - |S| / |M|)^{\alpha l}, \tag{4}$$

where α is an adjustable parameter controlling how fast initial weights decrease (empirically chosen in range 0.5-2, equal to 1 throughout this paper). Here, $I(x_i, 0)$ represents prior knowledge about the variable influences, and this governs the sampling weights for a number of initial iterations. In this paper we used $I(x_i, 0)$ equal to the sum of variances of log-odds for response classes.

It is obvious that $I(x_i, l)$ decreases and $SVI(x_i, l)$ grows with the number of iterations l. Therefore, for sufficiently low initial influences, the learned variable importance will dominate the sampling weights after a number of iterations. Sampling with replacement (versus without) reduces the computational time up to 5 times [5]. However, it poses additional challenges related to potential "overweighting"effect for a small group of influential variables preventing other relevant variables from entering the model. This effect could be controlled by a gradual transition from initial to learned importance based upon the weights.

Here is the formal description of the dynamic feature weighting algorithm.

Gradient boosting trees with dynamic feature selection

1. Set $I(x_i, 0)$ for $i = 1, \ldots, n = |M|$ to the initial response deviation. Initialize $w(x_i, 0) = I(x_i, 0)$. Set current residuals (responses) to output variable values. Set $SVI(x_i, 0) = 0$.
2. Fit the next GBT tree to the current residuals using $p(x_i, l) = w(x_i, l) / \sum_j w(x_j, l)$ as the selection weights. At each tree node, a small fixed number $n_0 << n$ of variables is selected with replacement using selection probabilities $p(x_i, l)$ and the best split is searched only amongst this subset. l is the current iteration number.
3. Calculate variable importance $VI(x_i, l)$ on the i-th variable as in (3).
4. Calculate $SVI(x_i, l+1) = SVI(x_i, l) + VI(x_i, l)$. Update variable weights as $w(x_i, l+1) = I(x_i, 0) \cdot (1 - S/M)^{\alpha \cdot (l+1)} + SVI(x_i, l+1)$.
5. Update residuals with the difference between the predicted values and the old residuals.
6. Return to step 2 if the maximum iteration number is not exceeded ($l < l_{max}$).

1 Raw values,
2 Wavelet coefficients,
3 Chebyshev coefficients,
4 PCA coefficients,
5 Statistical moments,
6 DTW distances to base signals,
7 DTW+1NN predicted class.

2. Selecting the optimal feature classes

We extracted several types of features from each time series:

First we want to make sure that each of the feature classes helps to improve classification accuracy on at least one dataset. Here we are assessing the quality of only large feature classes – raw, wavelets, chebyshev and PCA – that, being irrelevant to the response, can decrease classification accuracy. In order to evaluate the contribution of each of the feature classes, we compare the performance of two models: the model learned on all features F and the model learned on all features except for a specific feature class F_X. Each model is learned 10 times with different subsampling sets.

The results are summarized in Table 1. The columns corresponding to feature types show the median of the ratio of test errors ϵ_X to the test error on the full feature set ϵ. We run a t-test for each set of experiments checking if $\epsilon_X/\epsilon > 1$. The values in bold correspond to p-values less than 0.05 indicating that the given type of features is important for prediction on the given dataset. Table 1, columns 2–6, shows that all feature types except for PCA provide a statistically significant change in test errors. Although it is hard to explain the influence of a specific feature type on the response through a wide variety of UCR datasets, we could speculate that PCA features are less important because most of UCR datasets do not have enough samples to robustly estimate more than few PCA components.

We have found that for each feature class X there is at least one UCR dataset where the removal of feature set $F_X$has a negative impact on test error (not statistically significant for PCA). However we simultaneously get a decrease of test error on other datasets. Although the GBT model with dynamic feature weighting can handle massive sets of features, the number of training samples is usually limited, and filtering out irrelevant feature types prior to model construction can increase accuracy. Exhaustive search of the subset of feature classes that minimizes CV errors is too expensive. Also our experience of working with GBT indicates that sometimes the choice of GBT parameters N and ν is crucial. We solve both optimization problems with one algorithm based on particle filtering with simulated annealing.

3. Particle Filtering

The objective of the Particle Filtering (PF) algorithm is to optimize cross-validation error as a function of GBT model complexity N, shrinkage ν and feature subsets $\hat{F} \subset F$. Since iterating through all possible feature classes is infeasible we limit our search to classes of features so that $\hat{F}$ can be represented as $\hat{F} = F_{X_1} \cup F_{X_2} \cup .. \cup F_{X_k}$, where each F_{X_i}

Table 1. Medians of test errors on various feature sets normalized by the test error ϵ on F. Columns 2–6: for each feature type X we plot test error ϵ_X on feature set $F \backslash F_X$. "Random" column: test error ϵ_{RND} on random set of features. "1NN" column: the test error ϵ_{DTW} of DTW+1NN method. Boxes where t-test indicates that $\epsilon_X > \epsilon$ (columns 2–6) or $\epsilon_{RND,DTW}/\epsilon > 1$ (last two columns) with p-values greater than 0.05 are in bold.

Dataset	Raw	Wavelets	Chebyshev	PCA
ECG	0.789	**1.508**	0.829	0.950
Yoga	0.999	**1.025**	0.999	0.996
Two_patterns	1.000	2.750	1.000	1.000
wafer	0.925	**1.487**	**1.359**	1.093
Synthetic_Control	1.000	1.000	1.000	1.000
Swedish_Leaf	1.013	0.994	**1.066**	1.000
OSU_Leaf	0.926	1.000	**1.137**	0.978
Face(all)	1.017	1.008	**1.186**	1.004
Gun_Point	0.888	0.854	1.000	0.894
CBF	0.564	0.820	0.822	1.079
Trace	0.000	0.000	0.000	0.000
Face_Four	0.774	**1.524**	0.762	1.083
Lighting2	**1.192**	**1.069**	0.969	1.000
Lighting7	**1.323**	1.000	0.938	0.967
Olive_Oil	0.600	1.000	1.000	1.000
Coffee	0.000	**Inf**	0.000	0.000
Fish	1.000	1.063	1.000	0.920
Beef	0.837	**1.300**	0.905	0.900
Adiac	1.010	**1.034**	1.007	0.993

represents a class of features such as raw, wavelets etc. Each feature class is removed or added to a feature set with a fixed probability on a resampling phase. Particle weight is proportional to the difference between cv error of the particle and cv error averaged over all particles on the current iteration. Each weight is also multiplied by an "overlapping" factor that gives priority to the particles whose classification errors are consistently lower than errors observed on the previous iteration. The details of the method are summarized in Algorithm 1.

Algorithm 1: Particle filtering with simulated annealing

Notation:
- $1(x)$ is a step function: $1(x) = 1$ if $x > 0$, otherwise 0
- ln is natural logarithm
- $U(a, b)$ denotes a real number sampled from a uniform distribution in $[a, b]$
- X denotes a class of features $f_X \subset F$ (wavelets, raw, etc)
- F denotes the full feature set

1. Initialize: $f_{sa} = f_0 = 0.99$, $\beta = 50$, $t_{max} = 10$, $t = 0$, $q = 5$, $m = 20$, $p = 0.2$, $N_{min} = 30$, $N_{max} = 2000$, $\nu_{min} = 0.01$, $\nu_{max} = 0.5$
2. Initialize m particles with input parameters $N = \#\text{samples}/3, \nu = 15/N, f = F$
3. Evaluate particle weights:
   For each particle p_i
   calculate q-fold cv error $\{\epsilon_i^{(k)}\}_{k=1..q}$, $\epsilon_i = \sum_{k=1}^{q} \epsilon_i^{(k)}$
   calculate the overlap factor γ_i (equal to 1 for $t = 1$):
   calculate the average $E(\epsilon_{min})$ and standard deviation $Std(\epsilon_{min})$ of the errors distribution of the maximal weight particle from the previous iteration $\{\epsilon_{min}^{(k)}\}$
   Calculate the number J of CV fold errors that fall below
   $$\epsilon_b = E(\epsilon_{min}) - 3 \cdot Std(\epsilon_{min}): J = \sum_{k=1}^{q} 1(\epsilon_b - \epsilon_i^{(k)}).$$
   $\gamma_i = (J + 1)/q$
   EndFor
   Calculate average error $\bar{\epsilon} = \frac{1}{m} \sum_{1}^{m} \epsilon_i$
   For each particle p_i calculate weight $w_i = (\bar{\epsilon} - \epsilon_i)1(\bar{\epsilon} - \epsilon_i)\gamma_i$.
4. Resample particles
   For each $i = 1..m$
   Sample a number j from $1..m$ with probabilities proportional to w_i
   Set a new particle $p_i^{(n)}$ with $\{N_i^{(n)}, \nu_i^{(n)}, f_i^{(n)}\}$, equal to parameters of p_j
   Resample particle parameters:
   $N_i^{(n)} = N_i^{(n)} + \exp U(ln(N_{min}), ln(N_{max}))f_{sa}1(U(0, 1) - 0.5)$
   $\nu_i^{(n)} = \nu_i^{(n)} + \exp U(ln(\nu_{min}), ln(\nu_{max}))f_{sa}1(U(0, 1) - 0.5)$
   For each feature class X
   If $U(0, 1) < p$ then do
   [If $f_X \subset f_i^{(n)}$ then remove f_X from $f_i^{(n)}$ otherwise add f_X to $f_i^{(n)}$]
   EndFor
   $f_{sa} = f_0^{\beta t}$
   $t = t + 1$
   If $t > t_{max}$ End otherwise Goto 3

We test our TSC method on several UCR datasets. We run the algorithm on each dataset 10 times with different GBT random seed to reduce possible effect of fluctuations. The results are summarized in Table 2. One can see that GBT model with parameters and feature subset optimized by Particle Filtering is almost always notably superior to one built without such optimization. Test error values of simple DTW+1NN classifier are listed for comparison.

Table 2. Test errors.

Dataset	Average test error on all features	Average test error with PF	DTW+1NN error
Beef	0.167	0.13	0.467
CBF	0.0392	0.0186	0.004
Coffee	0.0214	0.00357	0.179
ECG200	0.068	0.052	0.12
Face(all)	0.14	0.191	0.192
Face(four)	0.124	0.0557	0.114
Fish	0.167	0.147	0.160
Gun-Point	0.0793	0.0793	0.087
Lightning-2	0.251	0.131	0.131
Lightning-7	0.290	0.256	0.288
OliveOil	0.2	0.17	0.167
OSU Leaf	0.395	0.355	0.384
Swedish Leaf	0.101	0.107	0.157
Synthetic Control	0.025	0.012	0.017
Trace	0.0	0.0	0.01
Two Patterns	0.0075	0.0	0.0015
Wafer	0.017	0.00393	0.005
Yoga	0.161	0.163	0.155

Conclusions

This work deals with TS classification problems. The proposed approach creates a massive number of features including original signals, by-class warped signals, wavelet and chebychev decomposition coefficients of warped signals, summary statistical moments of warped signals, and even predicted by DTW-1-NN labels used as input features. Gradient boosting of trees with imbedded dynamic feature weighting capable of handling hundreds of thousands predictors is then used for classification. Model parameters and the subset of features that the model is trained on are optimized using Particle Filtering. A set of experiments on UCR datasets show that this combination provides a superior learner relative to the well know state of the art approach. The future work will concentrate on

refining of this approach for important industrial applications and porting the methodolgy to the time series regression problems.

References

1. B. A., K.Torkkola, and T. E. *Best Subset Feature Selection for Massive Mixed-Type Problems*, volume 4224/2006, pages 1048–1056. Springer, 2006.
2. R. Agrawal, C. Faloutsos, and A. Swami. Efficient similarity search in sequence databases. In *In proceedings of the 4th Int'l Conference on Foundations of Data Organization and Algorithms*, pages 69–84, 1993.
3. Anonymous.A. Signal classification through massive feature extraction from warped signals. In *Submitted to ECML*, 2007.
4. D. J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. *In Working Notes of the Knowledge Discovery in Databases Workshop*, pages 359–370, 1994.
5. A. Borisov, V. Eruhimov, and E. Tuv. Dynamic soft feature selection for tree-based ensembles. In I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh, editors, *Feature Extraction, Foundations and Applications*. Springer, New York, 2005.
6. L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
7. L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth, Belmont, MA, 1984.
8. Y. Cai and R. T. Ng. Indexing spatio-temporal trajectories with chebyshev polynomials. In *SIGMOD*, 2004.
9. K. Chan and A. W. Fu. Efficient time series matching by wavelets. In *In proceedings of the 15th IEEE Int'l Conference on Data Engineering*, pages 126–133, 1999.
10. K. Deng, A. Moore, and M. Nechyba. Learning to recognize time series: Combining arma models with memory-based learning. In *IEEE Int. Symp. on Computational Intelligence in Robotics and Automation*, volume 1, pages 246 – 250, 1997.
11. J. Friedman. Greedy function approximation: a gradient boosting machine. Technical report, Dept. of Statistics, Stanford University, 1999.
12. J. Friedman. Stochastic gradient boosting. Technical report, Dept. of Statistics, Stanford University, 1999.
13. P. Geurts. Pattern extraction for time series classification. In *In proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 115–127, Freiburg, Germany, Sep 3-7 2001.
14. T. Hastie, R. Tibshirani, and J. Friedman. *The elemetns of statistical learning: Data mining, inference, prediction*. Springer, 2001.
15. Y. Huang and P. S. Yu. Adaptive query processing for time-series data. In *In proceedings of the 5th Int'l Conference on Knowledge Discovery and Data Mining*, pages 282–286, San Diego, CA, Aug 15-18 1999.
16. E. Keogh. Data mining and machine learning in time series databases, 2004.
17. E. Keogh, S. Chu, D. Hart, and M. Pazzani. An online algorithm for segmenting time series. In *In Proceedings of IEEE International Conference on Data Mining*, pages 289–296, 2001.
18. E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana. The ucr time series classification/clustering homepage, 2006.
19. F. Korn, H. Jagadish, and C. Faloutsos. Efficiently supporting ad hoc queries in large datasets of time sequences. In *In proceedings of the ACM SIGMOD Int'l Conference on Management of Data*, pages 289–300, 1997.

20. J. Lin, E. Keogh, S. Lonardi, and B. Chiu. A symbolic representation of time series, with implications for streaming algorithms. In *In proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, San Diego, CA, June 13 2003.

21. I. Popivanov and R. J. Miller. Similarity search over time series data using wavelets. In *In proceedings of the 18th Int'l Conference on Data Engineering*, pages 212–221, San Jose, CA, Feb 26-Mar 1 2002.

22. C. A. Ratanamahatana and K. E. Everything you know about dynamic time warping is wrong. In *Third Workshop on Mining Temporal and Sequential Data, in conjunction with the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

23. H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Acoustics, Speech, and Signal Proc.*, ASSP-26:43–49, 1978.

Статья поступила в редакцию 25.04.2008

УДК 519.23

'SPLIT AND PEEL' RULE INDUCTION METHOD

© [1]Treebushny D., [1]Kotkov V., [2]Chikalov I.

[1]Institute of Mathematical Machines and System Problems
NAS Ukraine,
Prospekt Glushkova, 42, Kiev, Ukraine, 03680 GSP
e-mail: *{dima, kotkovv}@env.com.ua*

[2]Intel Corporation,
Nizhny Novgorod Lab,
30 Turgeneva st., Nizhny Novgorod, Russia, 603024
e-mail: *igor.chikalov@intel.com*

Abstract. Patient Rule Induction Method (PRIM) [2] is a rule learning procedure that seeks to locate bumps: regions in the feature space where an output variable has substantially higher values than its mean value in entire input domain. Though accepted by many practical researches the original PRIM may perform poorly on datasets containing multiple bumps. The paper proposes an addition to classical PRIM: a splitting procedure that replaces peeling to process a multimodal bump. Performance of the new method is compared with the classical algorithm on an artificial dataset simulating fault analysis problem.

Introduction

Patient rule induction method (PRIM) was proposed by Friedman and Fisher as an algorithm of optimization of expected function value. Several problems of optimization, classification, and clustering can be formulated in such a form. PRIM generates interpretable solutions – associative rules describing hypercubes in an input space. A distinctive feature of PRIM is patience – unlike other rule induction algorithms (CART [1], RIPPER [3], CN2 [4]) PRIM comes to a solution through multiple iterations. This improves precision as misdirected iterations are compensated on later stages, makes the solution more stable to small changes in data and increases a search breadth – more input variables have a chance to participate in the solution.

We applied PRIM to the analysis of root causes of yield loss in semiconductor manufacturing. While performing the experiments we discovered a property of PRIM that complicates work with multiple bumps in data. To overcome this we implemented box splitting procedure that separates bumps.

The rest of paper is organized as follows. Chapter 1 gives basic notions, describes essential details of PRIM and describes a problematic situation with multiple bumps. Chapter 2 describes the box splitting procedure and all modifications that are necessary to incorporate it in PRIM. Chapter 3 experimentally compares the modified algorithm with original PRIM on a synthetic data set modeling failure analysis problem.

1. Bump hunting

1.1. **Problem statement.** Let $\mathbf{x} = (x_1, x_2, ..., x_p)$ be input variables (real valued or categorical) and X_j be a set of possible values (domain) of x_j for $j = 1, \ldots, p$. We will call $X = X_1 \times X_2 \times \ldots \times X_p$ *input space*. Let y be a real valued output variable and $D = \{d^i = (\mathbf{x}^i, y^i), i = 1, ..., n\}$ be a random sample taken from an unknown probability

distribution $p(y, \mathbf{x})$. For given D the goal is to find such a sub-region $R \subset S$ that the mean expected output value in R is substantially higher than the mean output value in the whole input space S. We will focus on the problem of bump hunting i.e. generating constraints on input variables that caused output value to be high. It imposes two restrictions on R: its description must be interpretable by an expert and it should be representative, i.e. contain enough samples from D.

Let us call *elementary constraint* on a variable x_j any subset $s_j \subset X_j$, such as

$$s_j = \begin{cases} [a_j,\, b_j], & \text{if } x_j \text{ is numeric;} \\ \{s_{j1},\, \ldots,\, s_{jm}\}, & \text{if } x_j \text{ is categorical.} \end{cases}$$

A *box* $B = s_1 \times s_2 \times \ldots \times s_p$ is a combination of elementary constraints on all input variables. We will state that a variable x_j *participates* in the box B if $s_j \neq X_j$. For interpretability purpose R must be a box or a union of small number of boxes, i.e. $R = \bigcup_{k=1}^{K} B_k$.

Two important characteristics of a box are *output mean* and *support*. For a box B we estimate the support as $\beta_B = \frac{1}{N} |\{x^i \in B\}|$ and the output mean as $\bar{y}_B = \frac{1}{N\beta_B} \sum_{d^i \in B} y^i$. For given β_0 the problem is to find a box $B_1 = \arg\max_{b \in B, \beta_b \geq b_0} \bar{y}_b$. To find multiple bumps one should remove from D samples covered by B_1 (they are considered as "explained") and repeat the process until the mean output value of the current box becomes lower than some threshold.

1.2. **Patient rule induction method.** PRIM iteratively builds a set of boxes according to the following algorithm [2]:

1. build a single box;
2. perform box post-processing in order to simplify its description;
3. remove all data samples covered by the current box;
4. perform 1-3 until the specified number of boxes is reached or mean value of the current box is lower than a specified threshold.

A key step of the algorithm is *top-down peeling* and *bottom-up pasting* procedures, which build a single box. Top down peeling starts from a box that covers all data. At each step a small subbox b within the current box B is removed. The subbox b is chosen from a class of eligible subboxes $C(b)$ such as it maximizes some criterion $I(b)$ i.e. $b^* = \arg\max_{b \in C(b)}(I(b))$.

The set $C(b)$ contains several subboxes for each input variable. A real valued input x_j provides two subboxes: $b_{j+} = \{\mathbf{x} | x_j < x_{j(\alpha)}\}$ and $b_{j-} = \{\mathbf{x} | x_j > x_{j(1-\alpha)}\}$, where $x_{j(\alpha)}$ is α-quantile of distribution of samples $\{\mathbf{x}^i \in B\}$ by x_j. Parameter α is called *peeling fraction*; it regulates the algorithm patience and is typically set to $0.05 \div 0.10$. A categorical input x_j contributes to $C(b)$ a subbox $b_{jm} = \{\mathbf{x} | x_j = s_{jm}\}$ for each value s_{jm} encountered in B.

Three criteria I(b) differing in patience degree are considered:

1. $I(b) = \bar{y}_{B-b} - \bar{y}_B$: directly targets increase in output mean in B, the most greedy
2. $I(b) = \bar{y}_B - \bar{y}_b$: minimizes output mean of peeled subbox, i.e. rejects the “worst“ part of data, most patient;

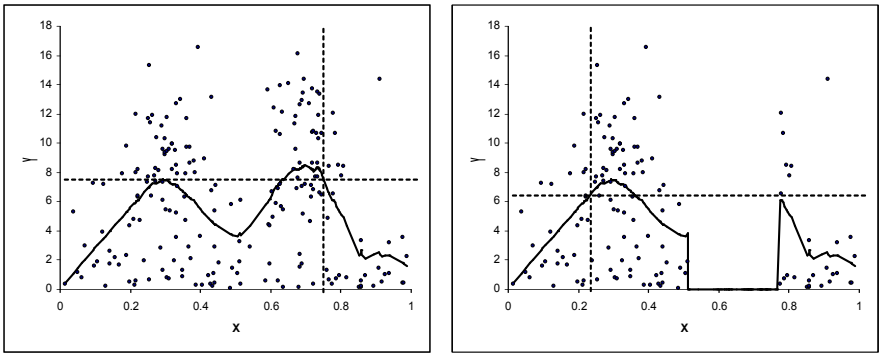


Fig. 1. (a) Scatter plot of an example data set. Dots are data samples, the solid line is a average of y by window of size $\beta_0 N$. Dashed vertical line is the box bound after peeling, grayed rectangle is the reported box after pasting. (b) The second box built after removing data contained in the first box is biased too.

3. $I(b) = \bar{y}_{B-b} - \bar{y}_b$: a sum of two previous criteria, maximizes difference between the output mean of the peeled and remained subboxes.

We used criteria 2 because it had shown best results in experiments.

Top-down peeling iteratively cuts the box until its support falls lower than a specified threshold or none of eligible subboxes increases the output mean. Bottom-up pasting is applied just after top-down peeling. It is an inverse procedure that enlarges at each step the box B by adding a subbox b^* that maximizes output mean. The class of subboxes eligible for pasting, is defined analogously to those used for peeling. A numeric variable x_j participating in B provides two subboxes that extend upper and lower condition on x_j respectively in order to cover extra $\alpha\beta_B$ samples. A categorical variable participating in B provides a subbox for each of its value not represented in B. Bottom-up pasting is over when the target mean cannot be increased by adding subboxes to B.

1.3. **Multiple bumps problem.** In case of multiple bumps PRIM can "fall between two stools". Let us demonstrate it by an example.

For the sake of simplicity assume there is a single real valued input variable x and a real valued output y. Figure 1 shows the scatter plot of a data sample and of y on x and the running average with centered window of $\beta_0 N$ samples which is used to provide spatial references. At the beginning, PRIM alternately peels outer slopes of the two peaks until reaches top of the left peak. Then it continues to peel the left face of the cube until the support threshold is reached. When peeling is over, pasting adds a part of the cut outer slope of the right cube and stops when the added cube is lower than the resulted cube mean. The box center does not coincide with the peak, thus the box corresponds to a non-optimal solution.

The problem remains after the first box is removed. "Leftovers"from the first box misdirect the algorithm in the same way and cause cutting off the outer slope of the second bump (Figure 1b).

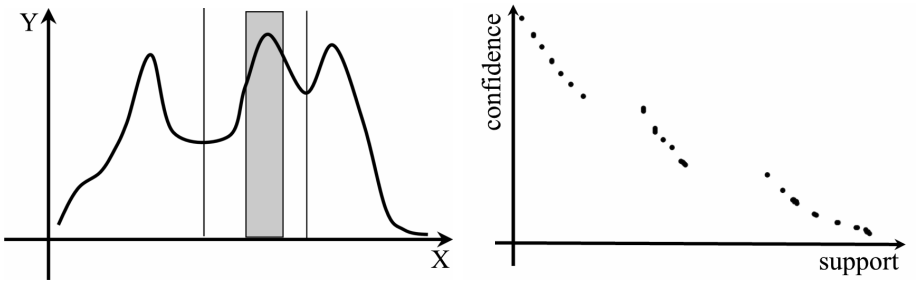


Fig. 2. Combination of peeling and splitting: (a) smoothed output curve: split points are marked with vertical lines. (b) peeling trajectory.

Let us describe the problem in general. The top-down peeling procedure can be viewed as steepest ascent: each iteration produces a step that is estimated to provide the greatest local increase to the objective function. In case of complex objective function criteria each step is seldom the optimal one in terms of leading to the ultimate solution. If each step has its own irregular bias, it is likely to be compensated by increasing steps number. This is not the case for multimodal distribution that causes a regular bias for many consecutive steps until the leading mode is localized. This introduces an error in the solution that can't be compensated by subsequent peeling steps and bottom-up pasting.

2. Modification of PRIM algorithm

The goal is to modify top-down peeling procedure in order to detect that the processed box contains multiple bumps, then to split it into two subboxes and chose one of them to continue peeling. The three topics discussed below are whether to apply splitting or peeling at the current step, how to choose the split point and which of the two halves to use further.

2.1. **Splitting criteria and choice of split point.** Algorithm searches for a split point that separates modes of conditional distribution $p(y|\mathbf{x})$. For each real valued variable x_j participating in cube B the algorithm splits elementary constraint $s_j = [a_j, b_j]$ into bins containing equal number of samples. Then the bin which has minimal mean output value is chosen to be a *splitting bin*. Decreasing of a bin size improves resolution, but increases the variance of the estimate, so we assume the bin size to be equal to a peeling fraction. The decision whether to perform splitting or peeling at the current step is taken by comparing the mean output value at the splitting point with subboxes eligible for peeling. If the mean output in the splitting bin is lower than the mean output in all eligible subboxes then splitting is performed. In other words splitting is performed when valleys between modes of conditional distribution of y over some variable become lower than the mean output value at the cube edges. It leads to a smooth peeling trajectory as shown in Figure 2.

The splitting bin is removed from the data set because it is actually a good candidate for peeling. Note that only minor changes are required in the original peeling procedure: modification only extends only a set of eligible subboxes.

2.2. **Choice of subbox to continue peeling.** As PRIM builds boxes iteratively, a rejected good candidate will be most likely located at a subsequent pass. Thus the primary target for the choice criterion is resistance to outliers. We have used a simple criterion that considers guaranteed optimization result: such a box is chosen that delivers maximal mean output value over contiguous bins covering at least $\beta_0 N$ samples.

2.3. **Modification of top down peeling procedure.** To integrate proposed changes in PRIM, the top-down peeling procedure should be changed in the following way.

1. For each real input variable x_j the algorithm splits an interval $[a_j,\, b_j]$, and constructs subboxes $b_{ji} \equiv \{\mathbf{x} \in B | t_{j(i-1)} \le x_j \le t_{ji}\}$, $a_j = t_{j0} < t_{j1} < \ldots < t_{jn_j} = b_j$ so that $\beta_{b_{jk}} \approx \alpha$ (sometimes exact equity cannot be reached due to a finite sample size and coinciding values). All subboxes b_{ji} join $C(b)$. The set of eligible subboxes provided by categorical variables remains unchanged.
2. If either the leftmost or the rightmost subbox b_{jk} is chosen for removal ($k = 0$ or $k = n$), the original peeling procedure is performed. If b_{jk} is in the middle of the interval ($0 < k < n$), define $B_l = \{\mathbf{x} | \, x_j \in [t_{j0},\, t_{j(k-1)}]\}$ and $B_r = \{\mathbf{x} | \, x_j \in [t_{jk},\, t_{jn}]\}$, $Q_l = \max\limits_{1<i<k-l-1} \bar{y}_{bji \bigcup \ldots \bigcup bj(i+l)}$ and $Q_r = \max\limits_{k<i<n-l} \bar{y}_{bji \bigcup \ldots \bigcup bj(i+l)}$. If $Q_l > Q_r$ make B_l the current box; otherwise make B_r the current box.

3. Test Results

The dataset that we have used for testing simulates semiconductors manufacturing data. A data sample corresponds to a lot: several units that are processed together at each operation. It contains five numeric variables $N_1, N_2, \ldots N_5$ describing quantitative characteristic (date, physical characteristics of process) and a categorical variable C_6 with 5 levels describing qualitative characteristics (material type, vendor, machine). A numeric response variable characterizes yield loss – a number of failed units in a lot.

A sample is drawn from a mixture of distibutions: a *base sample* characterizes a normal operation mode and three bumps characterize different failures. The base sample contains 44000 samples drawn from $5D$ Gaussian distribution by $N_1, \ldots, N_5$ with random mean vectors and random covariance matrix. Values for C6 are independently drawn from a multinomial distribution with a predefined level probability. Each bump sample contains 2000 samples drawn from $1D$ Gaussian distribution on variables participating in bump and uniform distribution on other variables. C_6 participates in one bump – its level probabilities have been changed for that case. Four categorical variables $C_7, \ldots, C_{10}$ were added to the data set that have different number of levels (from 2 to 10) and are not correlated with the response.

The response variable is drawn from a beta distribution with different parameters for the base sample and bumps. Table 1 contains distribution parameters for the base sample and all bumps.

Each algorithm is requested to report 3 boxes of support 0.03. Peeling fraction is set to be 0.01. Results are shown in the Table 2.

One can see that unlike the original algorithm the modified algorithm correctly reported all three bumps.

Table 1. Variable distribution in the test sample

Variables	Base sample	Bump 1	Bump 2	Bump 3
N_1	Mixture of	$Unif(0,1)$	$N(0.5, 0.06)$	$N(0.8, 0.06)$
N_2	50 $5D$ Gaussians,	$N(0.2, 0.06)$	$N(0.5, 0.06)$	$N(0.8, 0.06)$
N_3	with random mean	$N(0.2, 0.06)$	$Unif(0,1)$	$N(0.8, 0.06)$
N_4	vectors and	$N(0.2, 0.06)$	$Unif(0,1)$	$Unif(0,1)$
N_5	covariation matrix.	$Unif(0,1)$	$N(0.5, 0.06)$	$Unif(0,1)$
C_6	Mult(0.15, 0.2, 0.25, 0.2, 0.2)	Mult(0.2, 0.2, 0.2, 0.2, 0.2)	Mult(.004, .5, .004, .004, .488)	Mult (0.2, 0.2, 0.2, 0.2, 0.2)
Response	$beta(0.1, 10)$	$beta(1, 10)$	$beta(1, 10)$	$beta(1.5, 10)$

Table 2. Reported boxes.

PRIM	Optimized PRIM
$N_1 \in (0.079, 0.84]$, $N_2 \in (0.14, 0.87]$, $N_3 \in (0.08, 0.93]$, $N_4 \in (0.12, 0.94]$, $C_6 = 1$	$N_1 \in (0.01, 1.01]$, $N_2 \in (0.04, 0.34]$, $N_3 \in (0.04, 0.33]$, $N_4 \in (0.11, 0.27]$
$N_1 \in (0.38, 0.65]$, $N_2 \in (0.41, 0.69]$, $N_3 \in (0.03, 0.98]$, $N_4 \in (-0.09, 1.22]$, $N5 \in (0.38, 0.67]$, $C6 \in (2, 5)$	$N_1 \in (0.28, 0.62]$, $N_2 \in (0.4, 0.61]$, $N_5 \in (0.39, 0.68]$, $C_6 \in (2, 5)$
$N_1 \in (0.09, 0.89]$, $N_2 \in (0.15, 0.86]$, $N_3 \in (0.03, 0.92]$, $N_5 \in (-0.21, 0.90]$, $C_6 = 5$	$N_1 \in (0.65, 0.96]$, $N_2 \in (0.65, 0.88]$, $N_3 \in (0.66, 1.06]$

Conclusion

We proposed a modification of PRIM algorithm that overcomes the problem dealing with multiple bumps. Experimental results show that the modified algorithm correctly performs separating of multiple bumps and does not suffer from leftovers when building subsequent boxes.

Acknowledgements

This work was done in the frame of the partner contract P216 "Descriptive Supervised Optimization in High Dimensional Mixed Type Data" funded by Intel Corporation through Science and Technology Center of Ukraine (STCU).

References

1. *Breiman L., Friedman J., Olshen R. and Stone C.* Classification and Regression Trees. CityWadsworth, CityplaceBelmont, StateMA, 1984.
2. *Friedman J., Fisher N.* Bump-hunting in high-dimensional data // Statistics and Computing, V. 9, 1999, P. 123-143.
3. *Cohen W.* Fast Effective rule induction //Proceedings of the Twelfth International Conference on Machine Learning (ML95), Tahoe City, CA, USA.
4. *Clark P., Niblett. T.* The CN2 Induction Algorithm. //Machine Learning, V. 3(4), 1989, P. 261-283.

Статья поступила в редакцию 25.04.2008

УДК 681.3:51

ТЕХНОЛОГИЯ ВЫЯВЛЕНИЯ ИЗМЕНЕНИЙ И ОБНОВЛЕНИЯ ЦИФРОВЫХ КАРТ ГОРОДСКОГО КАДАСТРА НА ОСНОВЕ КОСМИЧЕСКИХ СНИМКОВ ВЫСОКОГО РАЗРЕШЕНИЯ

© Абламейко С.В., Крючков А.Н., Соболь Л.Н., Апарин Г.П.

Объединенный институт проблем информатики
Сурганова 6, Минск, Беларусь

e-mail: *abl[lab210, aparin]@newman.bas-net.by*

Abstract. The paper describes the technology of remote sensing data processing to solve the tasks of automated identification of city building up changing to update the digital maps of city cadastre, on the basis of space high resolution pictures. The examples of developed technology implementation are given.

Введение

Современное развитие науки и производства связано с получением и обработкой больших объемов информации. В области наук о Земле и окружающем ее пространстве проблема адекватного восприятия человеком больших объемов информации при принятии тех или иных решений традиционно решалась путем картографического отображения различной пространственно-привязанной информации в определенной системе проекций и условных знаков. Существующая потребность оперативного отображения и анализа меняющейся информации об окружающем мире и развитие компьютерных технологий ее обработки способствовали появлению нового вида картографической продукции – цифровых карт местности (ЦКМ) и основанных на их использовании геоинформационных систем (ГИС).

С течением времени в силу происходящих изменений на местности информация в ЦКМ «устаревает» и перестает соответствовать ее фактическому состоянию. Особенно быстрому изменению подвержены ЦКМ городского кадастра (ЦКГК). Под термином «городской кадастр» понимается всеобъемлющая информация об объектах застроенной территории города (зданиях, инженерных сооружениях, транспортных коммуникациях и др.) [1]. Для обеспечения актуальности данных ЦКМ необходимо их по мере необходимости обновлять – приводить информационное содержание к современному фактическому состоянию объектов и явлений на местности. Объемы работ и трудовые затраты по обновлению ЦКМ из-за их старения постоянно растут. Одним из способов обновления ЦКМ, выполняемого с меньшими трудозатратами, чем другими, является обновление по цифровым снимкам (ЦС) с данными ДЗЗ [2].

В настоящее время в развитых странах при решении комплекса задач обновления ЦКМ на основе данных ДЗЗ широко используются зарубежные программные ГИС-пакеты, такие как ArcGIS, ArcView и MapInfo (США), ПАНОРАМА (Россия), SICAD/open (Германия), WinGIS (Австрия) и др. Однако использование зарубежных пакетов ограничено такими факторами как стоимость, закрытость форматов представления данных, невозможность их расширения и дополнения и др. Кроме того,

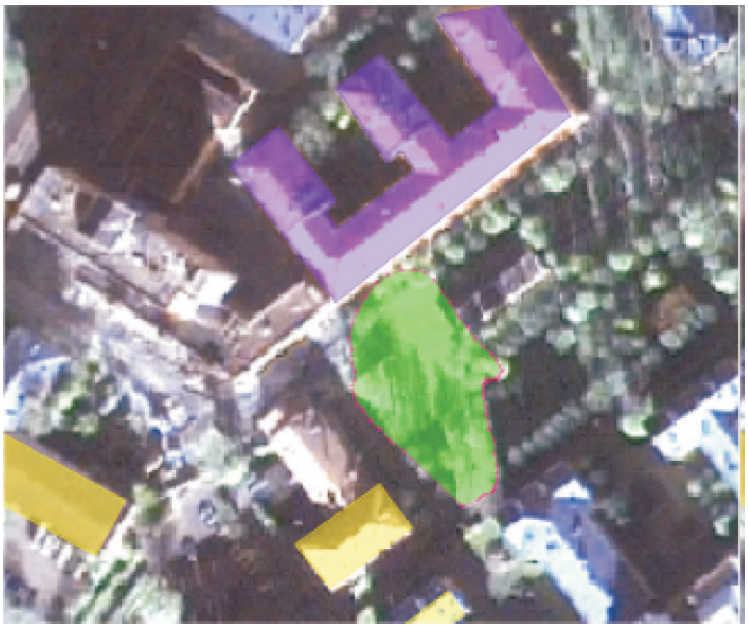


Рис. 1. Фрагмент цифрового космоснимка высокого разрешения

как правило, системы государственного управления накладывают жесткие ограничения на защиту информации и сертификацию программных продуктов. Хотя практика показывает, что наряду с отечественными разработками иногда целесообразно использовать и зарубежные пакеты.

В последнее время из-за большой важности и актуальности задач ведения городского кадастра все интенсивнее развиваются технологии обновления ЦКГК с использованием аэрокосмоснимков [3]. Способ обновления ЦКГК по ЦС высокого разрешения (рис. 1) обладает по сравнению с другими способами тем преимуществом, что использует самую актуальную информацию о местности, позволяющую в более короткие сроки обеспечить необходимую точность и информативность цифровых карт. Современные спутниковые изображения ЦС высокого разрешения предлагают широкий спектр метрических характеристик для обновления картографических материалов и для более точного нанесения геометрических характеристик на ЦКГК.

1. Технология выявления изменений и обновления ЦКГК

Обновление ЦКГК выполняется по одиночным космическим ЦС или фотодокументам, составленным по материалам космической съемки с помощью цифровых фотограмметрических систем (ЦФС). В процессе обновления контурной части содержание карты приводится в соответствие со снимком, устраняются обнаруженные ошибки в изображении форм рельефа (если рельеф был получен на ЦФС). В процессе обновления из ЦКГК исключаются отсутствующие на снимке объекты, включаются вновь появившиеся объекты, корректируется форма или семантика существующих объектов (рис. 2).

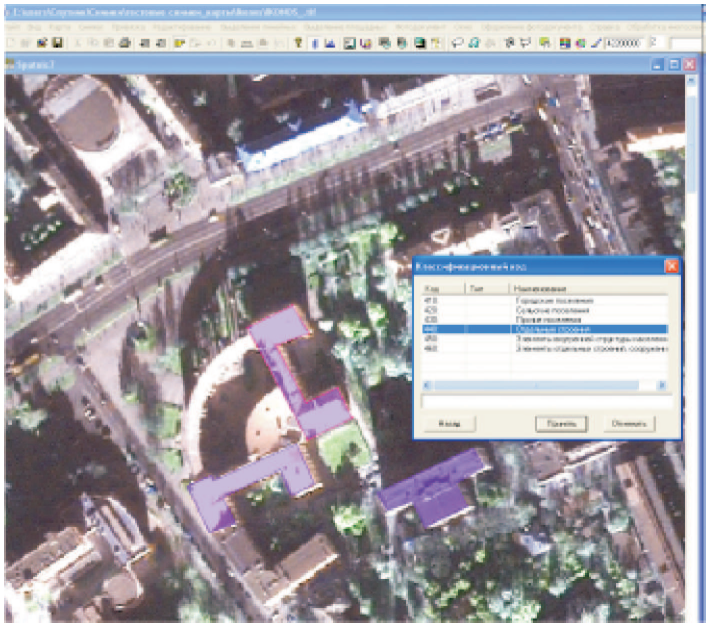


Рис. 2. Копия экрана: результат операции интерактивного выявления изменений в городской застройке

Особую актуальность представляет задача оперативного мониторинга городской застройки с целью актуализации карт городского кадастра для принятия решений по регистрации вновь появившихся или незарегистрированных объектов недвижимости. Такая задача оперативно решается с помощью ЦС высокого и сверхвысокого разрешения, получаемых с различных спутниковых систем .

Технология выявления изменений и обновления ЦКГК включает следующие операции:

- запрос и получение из базы данных исходных материалов (ЦС, ЦКГК, цифровых матриц рельефа (ЦМР));
- предварительная обработка ЦС;
- привязка ЦС к ЦКГК;
- автоматическое и интерактивное дешифрирование объектов ЦС;
- формирование массива изменений;
- формирование результатов дешифрирования
- автоматизированное внесение изменений в ЦКГК;
- контроль и редактирование обновленной ЦКГК;
- помещение ЦКГК в базу данных.

Запрос в базу данных обеспечивает получение исходных материалов: ЦКГК, ЦС, ЦМР, покрывающих обновляемую ЦКГК, необходимых для трансформирования ЦС к ЦКГК. Для обновления может быть использован цифровой фотоплан. В случае отсутствия в базе данных ЦМР, она может быть построена с заданным шагом сетки по имеющимся в ЦКГК горизонталям, отметкам высот и урезам воды. При загрузке ЦКГК может осуществляться их конвертация из обменных форматов (Shapefile, SXF

и Mid/Mif) во внутренний формат обработки F20S. При этом осуществляется приведение карт в единую проекцию и систему координат (поддерживаются: локальная система координат, геодезическая; проекция Гаусса-Крюгера (эллипсоид Красовского), проекция UTM (эллипсоид WGS-84)).

Предварительная обработка снимка включает функции по обработке ЦС, необходимые для улучшения изображения с целью дальнейшего дешифрирования снимков. В этот блок входят функции контрастирования, коррекции, подавления шумов, предварительной увязки снимков различных диапазонов и источников, получение синтезированных изображений и др.

Привязка ЦС и ЦКГК предназначена для формирования математической модели пространственного преобразования изображений ЦС и ЦКГК в системы координат друг друга. Задача решается путем определения параметров пространственного преобразования по известным координатам опорных точек, измеренных одновременно на снимке и на карте. В качестве таких опорных точек используются характерные точки на контурах объектов городской застройки, надежно опознающиеся на изображениях ЦС и ЦКГК. В состав программного блока привязки входят модули: формирования массива опорных точек; полиномиальных преобразований 1-3 порядка; полиномиально-триангуляционных преобразований; проективных преобразований; анализа геометрических искажений изображения ЦС.

При привязке снимков к ЦКГК в качестве опорных точек выбираются наиболее информативные для сопоставимых надежно распознанных объектов (рис. 3, рис. 4):
- зданий;
- инженерных сооружений;
- транспортных коммуникаций и др.

При привязке снимка к карте осуществляется контроль привязки как визуально, так и по отклонению в опорных точках. В дальнейшем параметры пространственных преобразований используются в блоке трансформирования векторной модели объектов обновления.

Операция выделения изменений на ЦКГК выполняется с использованием средств автоматической и интерактивной классификации объектов ЦС с использованием ЦКМ и базы эталонов [3]. В процессе дешифрирования осуществляется поиск и обнаружение на ЦС объектов местности с заданными параметрами яркости, размера и конфигурации.

Для выделения протяженных контурных объектов (классы объектов дорожной и гидрологической сети) используется интерактивный подход, который позволяет производить сегментацию объектов и осуществлять ручную коррекцию результатов обработки.

На аэрокосмических снимках с высоким разрешением (менее 2 м на пиксель) линейные объекты представлены набором однородных по яркости регионов с приблизительно постоянной толщиной, ограниченные двумя параллельными границами, на которых направление градиента имеет противоположные направления.

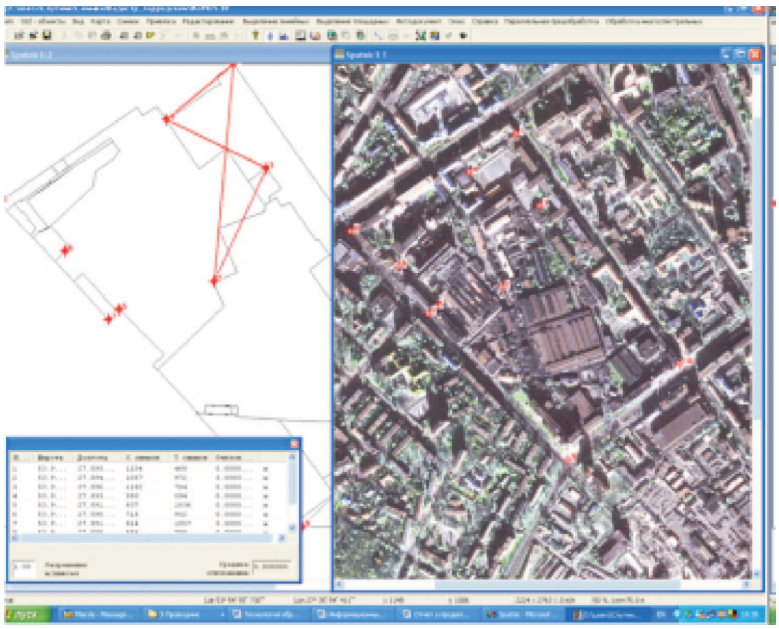


Рис. 3. Франмент карты на район интереса города Минска с фрагментом снимка на тот же район со снятыми опорными точками

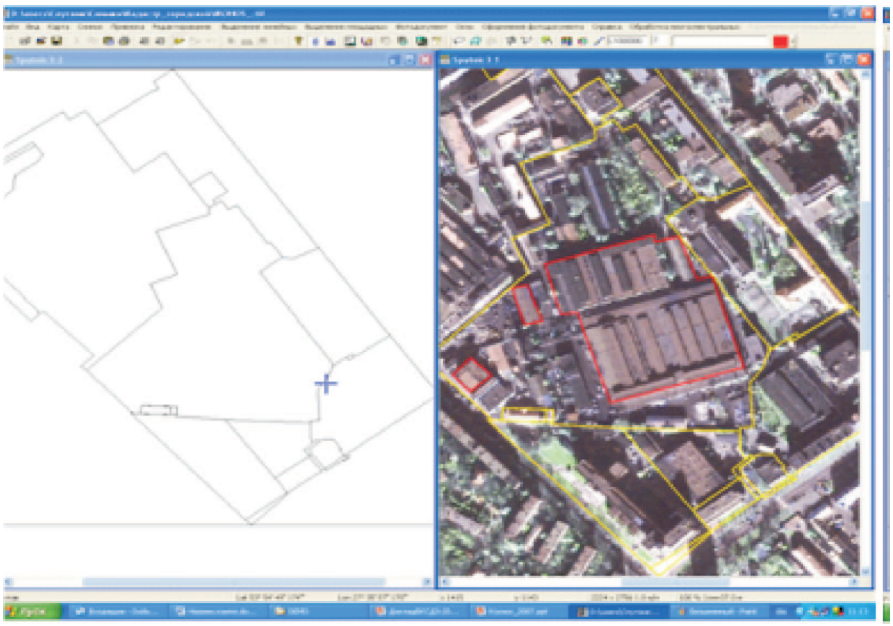


Рис. 4. Копия экрана операции выделения объектов городской застройки и присвоения им семантических характеристик

Кроме того выделение объектов городского кадастра может осуществляться путем ручного ввода на панхроматическом и многозональном изображении. По результатам обработки формируются объекты заданного класса в векторном формате с

назначением им семантических характеристик, которые могут редактироваться средствами специального графического редактора. Полученный массив изменений трансформируется в систему координат ЦКМ и передается в картографический блок для обновления ЦКМ.

После внесения изменений в ЦКМ, выполняется контроль обновленной ЦКГК (метрического описания, семантики, правильность приписания высот) и ее редактирование с помощью картографического редактора. После редакторских работ ЦКМ помещается в базу данных.

2. Экспериментальная проверка

Для проверки технологии обновления карт городского кадастра использовался снимок IKONOS с разрешением 1 пиксел/метр и карта района города Минска масштаба 1:5000 в геодезической системе координат проекции Гаусса-Крюгера (см. рис. 3 и 4).

Заключение

Разработанная технология выявления изменений и обновления цифровых карт городского кадастра на основе результатов дешифрирования космоснимков высокого разрешения может быть использована для решения различных комплексных геоинформационных задач, связанных с обнаружением интересующих пользователя объектов и ситуаций, выявлением изменений на местности, с получением количественных и качественных характеристик этих изменений с целью их анализа и принятия решений по обновлению цифровых карт.

Список литературы

1. *Лесных И.В. и др.* Городской кадастр – Новосибирск: СГГА, 2000. – 120 с.
2. *Абламейко С.В., Крючков А.Н.* Информационные технологии создания и обновления цифровых и электронных карт местности // Информатика. – 2004. – № 2. – С. 86-93.
3. *Абламейко С.В., Крючков А.Н., Соболь Л.Н.* Комплекс технической подготовки данных ДЗЗ: технология обработки информации // Сб.тезисов 6-й Украинской конференция по космическим исследованиям. – Евпатория: НЦУИКС, 2006, С. 148.

Статья поступила в редакцию 30.04.2008

УДК 004.5+004.657

ИНТЕЛЛЕКТУАЛИЗАЦИЯ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА БАЗЫ ДАННЫХ

© Акимов О.М., Шапцев В.А.

Тюменский Государственный Университет
НИИ ИИС
ул. Семакова, 10, г. Тюмень, 625000, Россия

e-mail: *akimov-oleg@ya.ru*

Abstract. A task is examined to translate a query, formulated in human language, in a standard request for the language of the SUBD. In the article it is described as that can be executed on the basis of conceptual graphs. A NL-query analyses and interprets an interface and presents him as a conceptual graph. Then this graph is modified information from the base of knowledge about a subject domain and structure of DB. A resulting graph becomes the basis of some SQL-query forming.

Введение

Усиливающаяся тенденция к хранению информации в СУБД, с одной стороны, и широкое распространение глобального Интернета и средств доступа к нему, с другой, делают вопрос быстрого и полного поиска информации, хранящейся в БД, одним из актуальных в современных информационных технологиях. Особенно остро этот вопрос стоит для класса пользователей, не являющихся специалистами в области информационных технологий, составляющие большинство современных пользователей Интернета.

Кроме того, большинство пользователей чаще всего:

- даже не специалисты в предметной области приложения или web-ресурса;
- не знают языка запросов SQL;
- не располагают структурой базы данных;
- редко используют расширенный поиск приложения или web-сервиса, предпочитая использовать быстрый поиск по одному-двум словам.

Соответственно такой поиск часто не дает полные и необходимые пользователю результаты. Естественный язык в данном контексте – предпочтительный способ общения человека с компьютером. Человек владеет своим родным языком куда лучше, чем искусственным – будь то языки программирования и разметки или языки визуального взаимодействия, например, язык кликов и окошек. Нам не приходится, зайдя в обычный магазин, рыться в каталоге товаров, мы просто спрашиваем у продавца-консультанта «а где...?» или «а у вас есть...?». Пользователя информационной системы также чаще всего интересует не функционал и возможности интерфейса, а информация необходимая ему.

Узким местом в разработке компьютерных систем, понимающих естественный язык, является получение адекватных по сути и объему знаний о предметной области. Эта проблема менее сложна при создании приложений, ориентированных на использование БД. Несмотря на то, что БД обычно содержат гигантские объемы информации, она (информация) посредством модели данных имеет достаточно

регулярную структуру, с одной стороны, и связана с узкой предметной областью, с другой. Более того, семантика БД четко определена.

В статье описывается метод, позволяющий перевести запрос с естественного языка на язык SQL. Данный подход создан на основе метода описанного в [4] для английского языка, но с модификацией для русского. Также приводится описание планируемого эксперимента, целью которого является проверка работоспособности и эффективности полученной методики.

1. Роль концептуальных графов в понимании естественного языка

Основная задача естественно-языкового интерфейса (ЕЯ-интерфейса): перевести запрос на естественном языке в стандартный запрос на языке СУБД. Но этот перевод невозможно совершить напрямую, т.к. слова и даже целые выражения естественного языка обычно имеют несколько смыслов и требуют дополнительной интерпретации. Поэтому вопрос сначала необходимо перевести на более выразительный язык представления, в частности, язык концептуальных графов.

Концептуальный граф (КГ) – это конечный, связанный, двудольный граф G [1, 4]:

$$G = G(C, R, A),$$

где $C = C(c_1, c_2, ..., c_n)$ – множество понятий, $R = R(r_1, r_2, ..., r_m)$ – множество концептуальных отношений, $A = A(a_1, a_2, ..., a_l), a_i = (c_j, r_k), c_j \in C, r_k \in R$.

Каждый концептуальный граф представляет собой одно высказывание или предложение. Он задает смысл этого предложения. Например, на рисунке 1 изображен граф предложения «Памятник находится в Тюмени». При графическом изображении графов прямоугольниками обычно обозначают понятия, концепты, а эллипсами – концептуальные отношения.

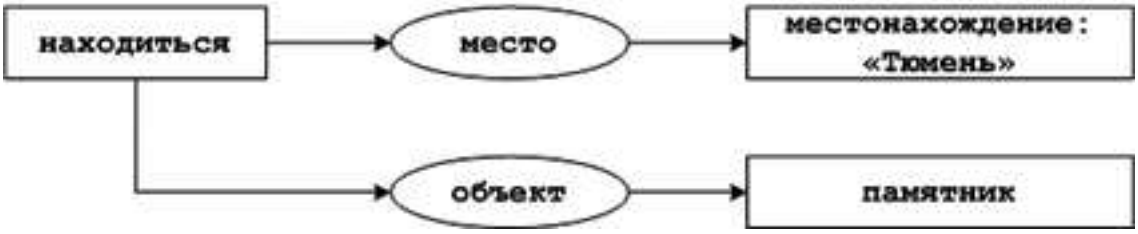


Рис. 1. Концептуальный граф предложения «Памятник находится в Тюмени»

Для более удобного представления КГ в тексте существуют, так называемые, языки описания, среди которых наиболее популярен CGIF (Conceptual Graph Interchange Form) [2, 3]. Граф, изображенный на рисунке 1, в формате CGIF запишется следующим образом:

$$[\text{Находиться} : *x][\text{Местонахождение} : \text{Тюмень} * y][\text{Памятник} : *z] \\ (\text{Место } ?x?y)(\text{Объект } ?x?z) \qquad (1)$$

Здесь x, y, z – метки, присваиваемые понятиям.

При построении КГ особо важная роль отводится глаголам, поскольку они определяют отношения между подлежащим, дополнением и другими компонентами предложения. Каждый глагол можно представить с помощью падежного фрейма [4], определяющего следующие данные.

1. Лингвистические отношения (агент, объект, инструмент и т.д.), соответствующие данному глаголу.
2. Ограничения на значения, которые могут присваиваться любому компоненту падежного фрейма.
3. Используемые по умолчанию значения компонентов падежного фрейма.

Падежный фрейм для глагола *находиться* в формате CGIF выглядит следующим образом:

$$[\text{Находиться} : *x][\text{Местонахождение} : *y][\text{Объект} : *z]$$
$$(\text{Место } ?x?y)(\text{Объект } ?x?z) \qquad (2)$$

Кроме падежных форм семантический интерпретатор, строящий концептуальный граф, использует иерархию понятий. Например, понятие *объект* является более общим, чем понятие *памятник*, т.е. понятие *памятник* является подтипом понятия *объект*. Эта информация также хранится в базе знаний о предметной области в виде подобного концептуального графа:

$$[\text{Памятник} : *x][\text{Объект} : *y](\text{Подтип}?x?y) \qquad (3)$$

В настоящее время практически нет систем, позволяющих автоматически строить КГ по входному тексту на русском языке [3]. Но подобные англоязычные системы продвинулись дальше, и один из методов построения концептуального графа в узкой предметной области описан в [4]. Модификация этого метода для русского языка описана ниже.

Сначала, входное предложение разбирает синтаксический анализатор, строит дерево синтаксического разбора. Это дерево передается семантическому интерпретатору. Дальнейшая последовательность действий семантического интерпретатора выглядит следующим образом:

1. Главным элементом дерева является глагол, сказуемое. Интерпретатор в базе знаний находит падежный фрейм соответствующий этому глаголу.
2. По дереву определяется подлежащие предложения. В падежном фрейме оно соответствует понятию *объект*, связанному с глаголом через концептуальное отношение *объект*. Но слово, являющееся подлежащим, не обязательно слово *объект*, оно может быть совершенно другим. Поэтому, используя иерархию понятий и операции ограничения и объединения, существующие для КГ, можно связать падежный фрейм и понятие, являющиеся подлежащим предложения.
3. Похожие действия происходят и с прямым дополнением предложения, и с остальными элементами дерева.

В итоге, полученный концептуальный граф представляет значение предложения. Например, входной запрос: «Какой памятник находится в Тюмени?», с помощью (2)

и (3) примет следующий вид:

$$[\text{Находиться}: *x][\text{Местонахождение}: \text{Тюмень} * y][\text{Памятник}: ? * z]$$
$$(\text{Место } ?x?y)(\text{Объект } ?x?z) \quad (4)$$

2. Связь концептуального графа с БД

Вернемся к ЕЯ-интерфейсу. Синтаксический анализатор и семантический интерпретатор позволяют перевести ЕЯ запрос на язык концептуальных графов. Дальше из него необходимо получить SQL-запрос для БД. При этом необходимо решить, где выполнять поиск в БД, какие выбрать имена полей и ограничения для запроса. Этой информации нет в исходном запросе, но она есть в БД, вернее в сведениях об организации БД.

В реляционной БД данные связаны отношениями между сущностями различных доменов. Обычно такую взаимосвязь двух сущностей представляют в виде диаграммы «сущность-связь». На рисунке 2 представлены отношение *object_location* и диаграмма «сущность-связь», отображающая взаимосвязь двух сущностей: объекты (*object*) и местонахождение (*location*).

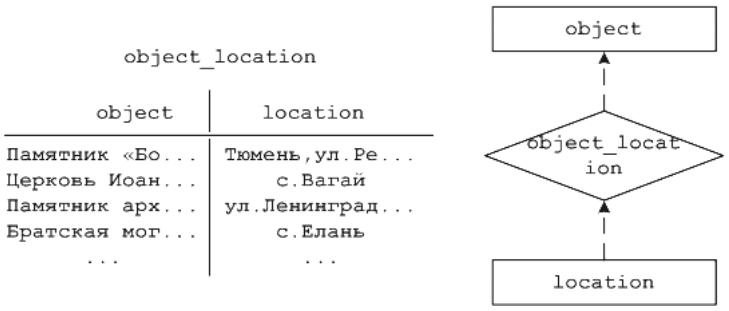


Рис. 2. Отношение из БД и диаграмма «сущность-связь»

Сущности из БД и понятия из базы знаний о предметной области прямо взаимосвязаны. Поэтому концептуальный граф, представляющий падежный фрейм глагола, можно расширить добавлением диаграммы «сущность-связь».

Тогда расширенный концептуальный граф G_{db} примет вид:

$$G_{db} = G_{db}(C, R, A, T, D),$$

где $T = T(t_1, t_2, ..., t_s)$ – множество отношений, таблиц, или представлений БД, $D = D(d_1, d_2, ..., d_v), d_i = (c_j, c_k, t_p), c_j \in C, c_k \in C, t_p \in T$ – представляет собой диаграмму «сущность-связь».

Формат описания CGIF также можно расширить, включением информации о БД. И тогда запись базы знаний для глагола *находиться* можно представить в таком виде:

$$[\text{Находиться}: *x][\text{Местонахождение}: *y][\text{Объект}: *z]$$
$$(\text{Место } ?x?y)(\text{Объект } ?x?z)\{\text{object\_location } ?y?z\} \quad (5)$$

Объединяя теперь входной запрос (4) и запись базы знаний для глагола (5), получаем расширенный КГ запроса:

$$[\text{Находиться} : *x][\text{Местонахождение} : \text{Тюмень} * y][\text{Объект} : ? * z]$$
$$(\text{Место}\ ?x?y)(\text{Объект}\ ?x?z)\{\text{object\_location}\ ?y?z\} \qquad (6)$$

Как видно из (6) построить SQL запрос к базе данных по такому КГ не составляет особого труда.

3. Об эксперименте

В настоящее время заканчивается работа над экспериментальным прототипом ЕЯ-интерфейса для информационной системы «Реестр ОКН». Это разработка НИИ ИИС ТюмГУ [5] для Комитета по охране и использованию объектов историко-культурного наследия администрации Тюменской области. «Реестр ОКН» представляет собой региональный цифровой информационный ресурс государственного реестра объектов культурного наследия народов РФ и поддерживает деятельность сотрудников государственных органов охраны памятников истории и культуры.

Целью экспериментальной работы с прототипом является проверка работоспособности подхода, описанного выше, и оценка границ его эффективного применения. В эксперименте планируется задействовать до 10 пользователей, работающих с этой информационной системой и являющихся специалистами в предметной области. В то же время будет организована работа с системой около 10 пользователей из вспомогательного персонала, знающего лишь только назначение системы.

Все участники эксперимента будут вводить в специальном поле интерфейса заданную совокупность из 20-25 запросов на естественном (русском) языке. Они же оценят результаты, выдаваемые системой, путем ввода в специальную таблицу-протокол по трехбалльной шкале. Все заданные и возможно придуманные пользователями запросы, соответствующие им концептуальные графы, расширенные концептуальные графы и итоговые SQL-запросы вместе с оценками пользователей будут фиксироваться в отельной БД.

Количественные характеристики плана эксперимента позволят обеспечить относительную статистическую устойчивость его результатов. Полученные в БД эксперимента данные ожидает тщательный многовариантный анализ.

Заключение

На текущем этапе исследования заявленной проблемы установлено следующее.

Во-первых, использование ЕЯ-интерфейса снижает барьер освоения приложения или web-ресурса. Теперь пользователю для поиска информации становится достаточно ввести запрос в поле ввода. Это актуально для современных пользователей Интернета, большинство из которых не являются специалистами не только в области ИТ, но и в предметной области приложения.

Во-вторых, использование изложенной методики позволяет путь до нужных данных сократить до минимума, фактически до одного шага. Это обеспечивает дополнительную привлекательность и быстродействие ресурса.

В-третьих, в нашем случае пользователь сосредотачивается на том, что ему необходимо, что он или она хочет найти, а не на том, как же это сделать. В итоге, пользователь будет решать интересующие его задачи, а не разбираться в функционале и возможностях интерфейса информационной системы.

Результаты эксперимента покажут справедливость этих гипотез.

Список литературы

1. A World of Conceptual Graphs – http://conceptualgraphs.org/
2. *John F. Sowa* Conceptual Graphs – http://www.jfsowa.com/cg/cgstand.htm
3. *Богатырев М.Ю., Латов В.Е., Столбовская И.А.* Применение концептуальных графов в системах поддержки электронных библиотек // Электронные библиотеки: перспективные методы и технологии, электронные коллекции. Тезисы постерных докладов Девятой Всероссийской научной конференции RCDL'2007 (Переславль-Залесский, Россия, 15-18 октября 2007 г.). Переславль-Залеский: Изд-во «Университет города Переславля», 2007 г., С. 104–110.
4. *Люгер Дж.Ф.* Искусственный интеллект: стратегии и методы решения сложных проблем, 4-е издание.: Пер. с англ. – М.: Издательский дом «Вильямс», 2003 г. – 864 с.
5. *Шапцев В.А., Широков А.В. и др.* Информационные системы поддержки деятельности // Материалы II-й Всероссийской конференции «Муниципальные информационные системы». – Екатеринбург, октябрь 2004 г. – С. 45-56. – http://www.egd.ru/index.php?menu_id=23102&show_id=23192.

Статья поступила в редакцию 20.04.2008

УДК 5019.7

МЕТОДЫ АНАЛИЗА И СИНТЕЗА ИНФОРМАЦИОННОЙ СИСТЕМЫ РАСПОЗНАВАНИЯ ОБРАЗОВ

© Амиргалиев Е.Н., Амиргалиева С.Н.

Казахстан, Алматы, КазНТУ им. К.И.Сатпаева, КБТУ

e-mail: *amir_ed@mail.ru*

Abstract. In the given work development of information system conceptual model of recognition and classification on the basis of mathematical methods and models of image recognition and classification, intended for the analysis and processing of multidimensional data is considered. The methodology of working out of information system is realised with use of the object-oriented approach and carried out in language of modelling of complex program systems – the unified language of modelling UML. The considered concept of construction of information system can be used in different subject domains, as the modern concept of working out of similar information systems. For realisation of design decisions it is used CASE-means Rational Rose.

Введение

В быстроразвивающейся сфере разработки объектно-ориентированных приложений становится все труднее и труднее создавать и поддерживать приложения, обладающим высоким качеством, укладываясь при этом в разумные временные рамки. Унифицированный язык моделирования (Unified Modeling Language, UML) появился как ответ на потребность в универсальном языке объектного моделирования, который могла бы использовать любая компания. UML – это технология моделирования сложных программных систем, принятая в современная технология в индустрии информационных технологий. Это метод детального описания архитектуры системы. С помощью нотации UML легче создавать и сопровождать систему, вносить в неё требуемые изменения и совершенствовать ее далее.

Язык UML, пришедший на смену многочисленным системам нотации и методикам проектирования, предложил нотацию для описания объектно-ориентированных моделей, которая стала промышленным стандартом. Однако для эффективного применения нотации UML необходимо сочетать ее с каким-либо методом объектно-ориентированного анализа и проектирования.

В описываемом методе сочетаются прецеденты использования, статическое моделирование, и диаграммы последовательности событий, которые встречаются в нескольких методах. Применяемая нотация основана на UML. В ходе моделирование прецедентов определяется функциональное требование к системе в терминах актеров и прецедентов. Статическая модель предлагает статический взгляд на информационные аспекты системы. Класс определяется в терминах своих атрибутов и взаимоотношений с другими классами. Результатом динамического моделирования является динамический взгляд на систему. Уточняются сформулированные ранее прецеденты с целью показать взаимодействие объектов, участвующих в каждом из них. Разрабатываются диаграммы кооперации и последовательности, отражающие кооперацию

объектов в каждом прецеденте. Зависящие от состояния аспекты системы описывается с помощью диаграмм состояний, причем для каждого объекта составляется своя диаграмма.

В качестве математического обеспечения разрабатываемой системы рассмотрены математический аппарат решения задачи распознавания и классификации, задачи групповых классификации и оптимизационные модели, использующие различные виды функциоалов качества.

1. Постановка задачи классификации Z_k

Пусть задана начальная информация I, $\{S\}$ – множество допустимых объектов и каждый объект $S_i \in \{S\}$, $i=1,..., m$ характеризуется n-мерным вектором, координаты которого называются признаками, взятыми из алфавита признаков, т.е.

$Si=(\alpha i1,\alpha i2,....,\alpha in)$, $i=1,..., m$, n – число признаков, $\alpha ij \in Mj$.

Требуется построить алгоритм классификации A для множеств $\{I(S)$, $S\}$, классифицирующий исходное множество объектов S на ряд непересекающихся классов (кластеров), Kj, $j=1,..., l$, так чтобы объекты, принадлежащие к одному классу (кластеру) были сходными (близкими), в то время как объекты, принадлежащие различным классам, были в некотором определенном смысле непохожими (удаленными):

$$A\{I(S), S\} = \bigcup_{j=1}^{l} K_j, K_i \bigcap K_j = \emptyset,$$

если $i \neq j$, $Ki \neq \emptyset$, i, $j=1$, $2,..., l$.

Построение кластеров можно рассматривать как задачу распознавания образов без учителя, учитывая, что на заданном множестве объектов, как правило, отсутствует всякая информация, касающаяся числа классов и структуры классов.

При построении алгоритмов по принципу минимума расстояния, отыскание кластеров и определение эталонов являются вопросами первостепенной важности. При построении таких алгоритмов, как правило, используются два подхода. Один из них – эвристический, и в его основе лежит интуиция и опыт. Второй подход предусматривает минимизацию или максимизацию некоторого выбранного показателя качества классификации.

В моделях групповых классификации введена метрика, используемая в пространстве классификаций. Рассмотрим задачу синтеза групповой классификации Z_C. Пусть $A_1, ...,\ A_m \in \{A\}$ – исходный набор алгоритмов решения задачи классификации Z_k для множества объектов $M = \{S_1, ..., S_n\}$.

Результатом применения алгоритмов A_i к множеству $(M,\ J(M))$ являются классификации $K_i(M) \in \Re(M)$, $\Re(M)$- пространство классификаций конечного множества объектов M, элементами которого являются отдельные классификации. Пусть определена метрика $d(K', K")$ в $\Re(M)$ и $\varphi(K) = \sum_{i=1}^{m} d(K, K_i)$, $K_i = K_i(M)$, $K \in \Re(M)$.

Тогда основная задача группового синтеза (групповых классификации) Z_C состоит в следующем. Найти классификацию $K^*(M) \in \Re(M)$, минимизирующую функционал $\varphi(K)$, т.е.

$$\varphi(K^*) = \min\ \varphi(K),\ \ K \in\ \Re(M).$$

Рассмотрим пространство классификаций $\Re(M)$ множества M. Известно, что по любой классификации $K \in\ \Re(M)$ можно построить соответствующее ей бинарное отношение R, которое является отношением эквивалентности на множестве M. Также, любому отношению эквивалентности $R(M)$ на множестве M однозначно соответствует классификация $\Re(M)$ для M.

Будем рассматривать стандартное представление классификаций $K(M)$ 87 $\Re(M)$ в виде конечного множества классов $K_i(M)$ – подмножеств множества M, т.е. $K(M) = \{K_1(M), K_2(M), ..., K_l(M)\}$ и значит для описания классификации достаточно перечисления номеров объектов, попавших в каждый из классов. Причем, как будет показано ниже, способ нумерации классов может быть произвольным для полученной любой классификации, и не влияет на результат их сравнения. При таком представлении классификаций для решения задачи Z_C нужно иметь метрику в $\Re(M)$, которая бы достаточно полно отражала реально существующие в данном пространстве расстояния, и исследовать свойства пространства $\Re(M)$, являющегося структурой.

Обозначим через $K^l(M)$ множество классификаций M на l, $1 \le l \le n$ классов, т.е. $\bigcup_{l=1}^{n}\ K^l(M) = \Re(M)$. Пусть $K_n(l)$- произвольная классификация из $K^l(M)$. Зададим отображение $d : K(M) \times K(M) \to Z$ с помощью следующей формулы:

$$d(K_n^u(l_u), K_n^\nu(l_v)) = 2n - \sum_{j=1}^{l_u} \max_{l\le i\le l_v} \left\{\ \left| K_{n,j}^u(l_u) \bigcap K_{n,i}^v(l_v)\ \right|\ \right\} -$$
$$- \sum_{i=1}^{l_\nu} \max_{l\le j\le l_u} \left\{\ \left|\ K_{n,i}^v(l_v) \bigcap K_{n,j}^u(l_u)\ \right|\ \right\}\ ,$$

где $K_n^t(l_t) = \left\{\ K_{n,1}^t(l_t), ..., K_{n,l_t}^t(l_t)\ \right\},\ \ 1 \le l_t \le n,\ \ t \in \{u, v\}$.

Введенная метрика обладать свойствами метрики[3].

Концептуальная модель информационной системы распознавания и классификации. Концептуальная схема разработанной информационной системы распознавания и классификации показана на рисунке 1.

Кратко опишем функциональные назначения подсистем, входящих в состав системы: Подсистема ***Справка-Help*** представлена как справочная система. Показывает справку о функциональных возможностях, как отдельных подсистем, так и системы в целом; ***Управление*** – подсистема представляет программный интерфейс процесса управления проектируемой системы; ***База данных*** –Совокупность ряда таблиц для хранения данных необходимых для системы; Подсистема ***предварительной обработки*** предназначена для предварительной обработки исходных данных: определения незаполненных данных и определение оптимальных подсистем признаков в описании объектов; определение информативных признаков;

Модели и алгоритмы классификации осуществляет работу алгоритмов классификации, составляющих базовый набор для системы; **Модели группового синтеза** представляет работу алгоритмов группового синтеза, реализованных в рамках системы; **Визуализация результатов** представляет результаты работы,

Рис. 1. Концептуальная модель системы распознавания и классификации

как подсистем, так и системы в целом для пользователей в нужном формате; **Анализ и оценка результатов** предназначена для анализа и оценки результатов на основе показателей выбранных видов функционалов качества и представляет основу для выработки рекомендации об использовании различных вычислительных схем; Подсистема ***ГИС моделирование*** представляет возможности использования разработанных методов, алгоритмов группового синтеза в рамках геоинформационного моделирования.

На рисунке 2 приведена функциональная модель информационной системы распознавания и классификации с применением методологии функционального анализа и проектирования системы SADT – (Structured Analysis & Design Technique). На диаграмме указана схема передачи управляющих параметров входных и выходных данных для каждой подсистемы.

В рамках информационной системы, в зависимости от постановок задач пользователя, можно осуществить оптимизационные процедуры, используя конкретный вид функционалов качества. Схема проведения оптимизационных процедур осуществляет подсистема *Анализ и оценка* (Рисунок 3).

В рамках данной схемы возможны следующие модели оптимизации (применяются различные комбинации алгоритмов, моделей группового синтеза, функционалов качества): одноуровневая модель $M1\ \{(M\ k,\ Aj,\ F\ t):\ k=1,\ \ldots,\ N;\ j=1,\ldots,\ M;\ t=1,\ldots,\ N.\}$; двухуровневая модель $M2\ \{(M\ k,\ Aj,\ Z\ ci\ ,\ F\ t):\ k=1,\ \ldots,\ N;\ j=1,\ldots,\ M;\ i=1,\ldots,\ T;\ t=1,\ldots,\ K.\}$.

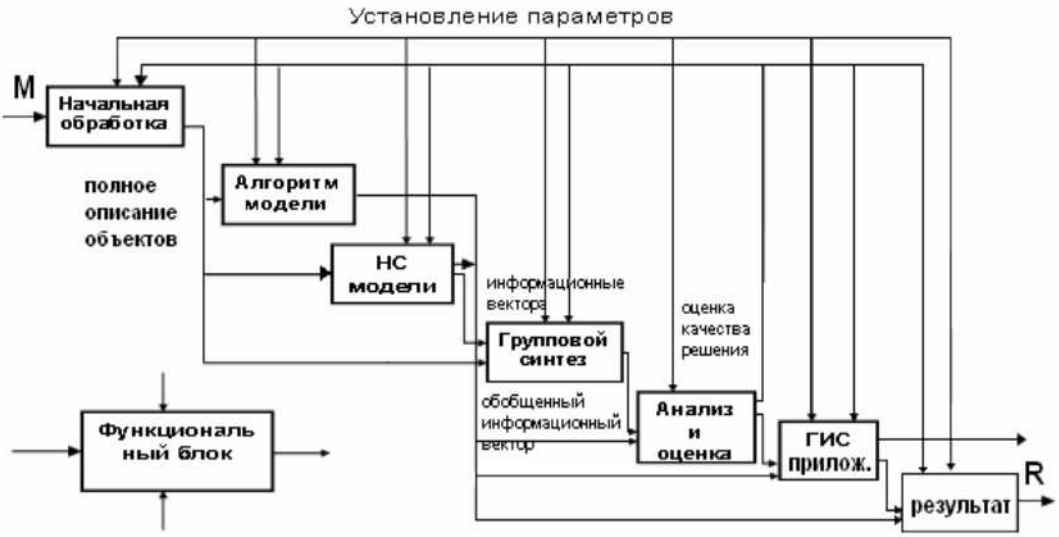


Рис. 2. Функциональная модель системы распознавания и классификации

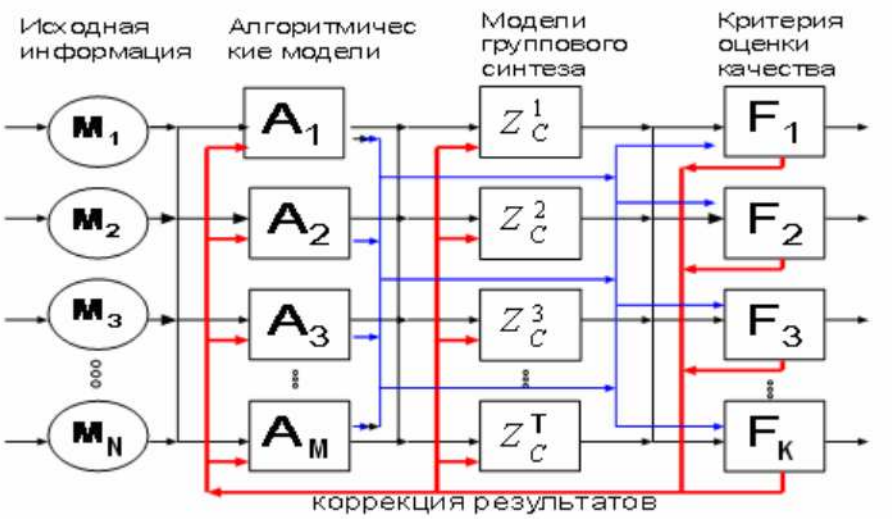


Рис. 3. Схема оптимизационных процедур (двухуровневая модель). Здесь: MN – множество исходных данных, AM – алгоритмы классификации, Z_C^T- модели группового синтеза, FK – функционалы качества.

В процессе моделирования системы использованы следующие виды диаграмм (нотации) UML: *диаграммы прецедентов* (*use case diagram*) представляет функциональность и поведение разрабатываемой системы, позволяет определить требования к системе, определить действующие в системе объекты и основные задачи, выполняемые этими объектами (отображение сценариев); *диаграммы классов (class diagram)* обеспечивают статическое проектное представление системы; *диаграммы кооперации*

(collaboration diagram) описывают взаимодействие объектов, абстрагируясь от последовательности передачи данных, отражаются все принимаемые и передаваемые сообщения конкретного объекта и типы этих сообщений; *диаграммы последовательности (sequence diagram)* позволяют определить последовательность передачи сообщений между объектами, показывают поток сообщений; *диаграммы состояний (state diagram)* предназначены для отображения состояний объектов системы, имеющих сложную модель поведения.

Для управления функционированием системы создается, так называемый *сеанс обработки,* диаграмма прецедентов которой показана на рис. 4.

Сеанс обработки – выбор вычислительного процесса и формирование совокупности входных и выходных данных для определенной подсистемы и сам процесс обработки данных. *Входные данные* – исходные данные, список алгоритмов и параметров. *Исходные данные* – подмножество входных данных подсистемы, которое является общим для всех алгоритмов этой подсистемы. Структура исходных данных регламентируется подсистемой. *Список алгоритмов* – подмножество алгоритмов подсистемы, которые выполняют обработку в данном сеансе. *Параметры* – подмножество входных данных подсистемы, которое не является общим для всех ее алгоритмов. Параметры группируются по алгоритмам. Каждый алгоритм объявляет свой набор параметров и их структуры. *Выходные данные* – совокупность кодов возврата и результатов. Каждый выполнявший обработку алгоритм ассоциируется с кодом возврата и результатом. *Код возврата* – числовое значение, обозначающее статус окончания алгоритма.

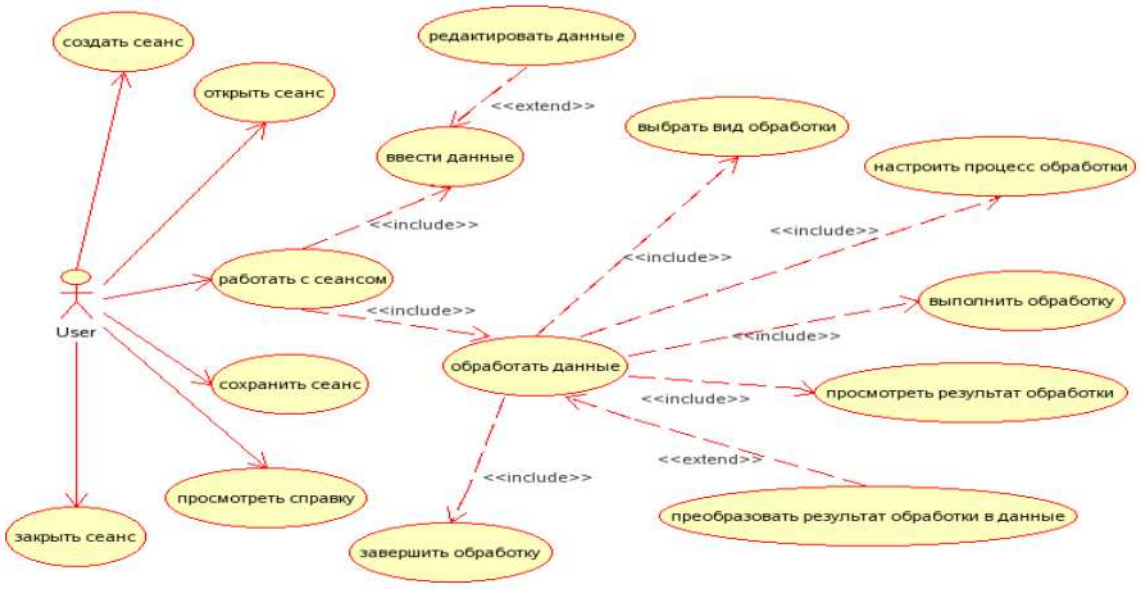


Рис. 4. Диаграмма прецедентов для сеанса обработки

Представим описания некоторых прецедентов, приведенных на рисунке 4.

Создать сеанс – *с*оздание нового сеанса. *Предусловие* – система незапущена или сеанс находится в состояний *новый. Основной поток событий: п*ользователь, запуская программу, или, используя элементы управления пользовательского интерфейса запущенной программы, создает *новый сеанс .пользователя. Постусловие* – система находится в состоянии *Новый сеанс.*

Открыть сеанс – открытие предварительно созданного сеанса**.** *Предусловие-с*еанс в состояний *Новый* или *Рабочий. Основные потоки событий:* 1)пользователь выбирает файл сеанса; 2) файл открывается для чтения; 3)сеанс инициализируется данными из файла; 4) файл закрывается. *Альтернативный поток событий-р*азличные ошибки ввода-вывода. Выводится соответствующее сообщение. *Постусловие* – *с*еанс переходит в состояние *Рабочий.*

Работать с сеансом – работа с сеансом. *Предусловие- с*еанс в состоянии *Новый* или *Рабочий. Основные потоки событий:*1) вводит данные; 2) обрабатывает данные; 3) просматривает результаты обработки. *Альтернативный поток событий – р*азличные ошибки при работе соответствующих подсистем. Выводится соответствующее сообщение. *Постусловие* – сеанс переходит в состояние *Работающий*, а затем в *Рабочий.*

Также по указанной схеме описываются и другие прецеденты сеанса обработки, указанные на диаграмме прецедентов (Рисунок 4).

Заключение

Таким образом, в данной работе мы показали некоторые аспекты моделирования информационной системы распознавания и классификации с применением унифицированного языка моделирования. Для реализации проектных решений использованы инструментарий CASE- средства Rational Rose.

Разработанная информационная система применена для решения реальных прикладных задач из области гидрогеологии, экологического мониторинга с использованием данных дистанционного зондирования и геологии.

Список литературы

1. *Г. Буч* Объектно-ориентированный анализ и проектирование. – М.: «Издательство Бином», 1999.
2. *Уэнди Боггс* Rational Rose & UML. – М.: «Издательство Лори», 1999, – 480с.
3. *Айдарханов М.Б., Амиргалиев Е.Н.* Алгоритмические основы построения систем классификации. – Алматы, 1998. – 100с.
4. *Вендров А.М.* Проектирование программного обеспечения ЭИС. – М.: 2000, - 290 с

Статья поступила в редакцию 27.04.2008

УДК 5019.7

МЕТОДЫ АНАЛИЗА И ПРОЕКТИРОВАНИЯ СИСТЕМЫ СИНТЕЗА ИСКУССТВЕННОЙ РЕЧИ

© Амиргалиев Е.Н., Мусабаев Р.Р.

КазНТУ им. К.И. Сатпаева, Казахстан, Алматы

e-mail: : *amir_ed@mail.ru, rmusab@gmail.com* ,

Abstract. In the given work the practical experience of speech synthesis algorithms realization and application are examined. Human-machinery speech interfaces, designated for the linguistics courseware is the subject of one of the most effective application of the given algorithms.

Введение

В данной работе рассматриваются вопросы создания информационной системы анализа и синтеза искусственной речи, некоторые проблемы которой приведены в предыдущих статьях [1, ?]. Главная задача систем синтеза речи – это преобразование входной текстовой информации в её выходное звуковое представление на одном из естественных языков. В самом оптимальном варианте, данные системы призваны заменить человека в процессе чтения текстов и выступить в роли автоматизированных дикторов, которые готовы неустанно читать тексты любых объёмов и любой сложности на любых языках. Под естественной речью понимаются звуковые и мимические составляющие речи человека. Входная текстовая информация обычно представляется в виде электронного текста (Рис. 1).

Данные алгоритмы в основном применяются при построении систем с человеко-машинными интерфейсами. Особенно их применение оправдано, если система оперирует текстовой информацией, подлежащей озвучиванию, а данная информация является переменной во времени. Классическим примером систем данного типа выступают системы контроля и управления параметрами технологических процессов, в которых активное участие принимают человек и ЭВМ [2].

Синтез речи относится к группе речевых алгоритмов [3], к которой также можно отнести: распознавание речи, анализ текста на естественном языке, алгоритмы автоматического извлечения информации и знаний из текстов, системы машинного перевода и др. Все эти алгоритмы тесно связаны между собой. Во многих из них применяются сходные технологические подходы. К примеру, задача синтеза речи является обратной к задаче её распознавания. При этом алгоритмы распознавания часто применяются на этапе построения систем синтеза для автоматизированного выделения элементарных звуков речи или сегментных границ.

Группа речевых алгоритмов относится к приоритетным направлениям научных исследований во многих странах мира. Главным образом, это связано со значительными качественными изменениями, которые они привносят и могут привнести в повседневную жизнь человека. Для примера, можно сказать, что только правительство Японии ежегодно тратит порядка 40 млн. долларов на разработки по информатизации японского языка.

Значимость применения речевых алгоритмов при построении человеко-машинных интерфейсов подчёркивает тот факт, что многие современные учёные связывают само появление человека (Homo sapiens) и резкий скачёк в его развитии с появлением у наших предков способности говорить [4]. Можно сказать, что речь – это катализатор развития человечества, позволивший ему за относительно короткое время осуществить резкий скачёк вперёд и получить всё то, что мы имеем сегодня. Главное, что даёт нам речь – это возможность общаться между собой и передавать опыт от одного поколения другому. Общий период эволюции предков человека насчитывает сотни миллионов лет. Возникновение же человека в современном виде (Homo sapiens) и появление у него способности говорить произошло примерно 200 тыс. лет назад. При этом первая письменность, которая возникла на Земле – шумерская. Возникла она примерно от 4 до 5 тысяч лет назад [5]. Из приведённой хронологии видно, что в течение большей части периода своего развития человек пользовался только исключительно устной речью. Отсюда можно сделать следующий вывод: человеческий мозг должен быть более приспособлен к усвоению устно-речевой информации, нежели визуально-текстовой.

В определённой степени речевое восприятие текстовой информации у здорового человека более первично, чем визуальное и стоит выше по иерархии. Главным образом, это связано с тем, что человек изначально учится говорить, а уже только значительно позже приобретает навыки чтения. Существует также проблема неграмотных людей, просто не умеющих читать. Человек приобретает навыки чтения относительно своей уже сформировавшейся и развитой устной речи. В данном случае исключение составляют люди родившиеся глухими. У таких людей процесс обучения чтению происходит путём прямого формирования образных ассоциативных связей с читаемым текстом.

Другая важная особенность: человеческий интеллект устроен таким образом, что способен воспринимать и обрабатывать информацию только в последовательном виде, в виде последовательностей образов или паттернов [6]. Человек не может, моментально увидев лист с печатным текстом, сразу прочитать его и усвоить информацию, содержащуюся в нём. Ему необходимо последовательно слово за словом, предложение за предложением его прочитать. По мере чтения, у него возникают последовательности, состоящие из различных образов. В процессе последовательного чтения у человека также последовательно возникают прямые автоассоциативные образы, связанные с текстом. Общий же смысл текста воспринимается как набор ассоциативных связей более высокого порядка между элементарными образами его составляющими. Для чтения текста человеку также необходимо прикладывать определённые моральные и механические усилия: двигать глазами, фокусировать зрачок, организовывать процесс чтения в рамках установленных порядков и правил.

Иначе происходит процесс восприятия устной речи. Устная речь изначально представлена в последовательной форме в виде последовательностей звуков, слов, предложений. Последовательно также изменяются ударения, интонации и скорость чтения. Восприятие устноречевой информации с помощью слуха происходит у человека практически самопроизвольно на уровне рефлексов. Это подтверждается тем

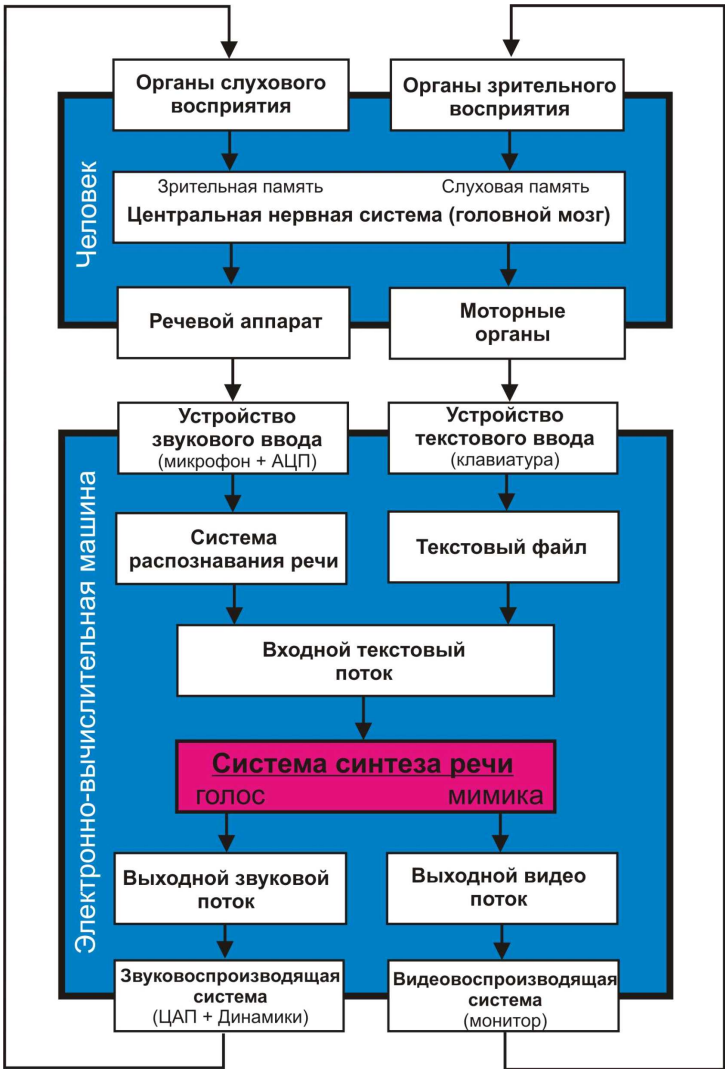


Рис. 1. Схема взаимодействия человека и ЭВМ с применением речевых человеко-машинных интерфейсов.

фактом, что даже если человек не желает воспринимать информационную составляющую речи, восприятие всё равно происходит помимо его воли и единственный способ препятствовать восприятию – это физически устраниться от источника звукового сигнала. В комфортной обстановке при оптимальных параметрах звучания (приемлемый уровень громкости, скорости чтения, отсутствие посторонних шумов) человеку требуются минимальные усилия для восприятия и усвоения информации представленной в устноречевой форме.

Важные различия заключаются также между восприятием устной речи с сопутствующей ей визуальной информацией и восприятием той же речи без сопутствующей визуальной информации. Типичным примером для первого случая является кинофильм, а для второго – радиопередача. В случае просмотра кинофильма человек

мысленно погружается в представляемые ему внешние визуальные и звуковые образы, воспринимая информацию практически такой, какой она ему преподносится. При этом мозговая активность находится на относительно низком уровне. В случае же прослушивания радиопередачи, человеку необходимо «включать» собственное воображение для воссоздания излагаемой картины, ему приходится постоянно извлекать некие образы, хранящиеся в собственной памяти. Другие образы могут возникать самопроизвольно из автоассоциативной памяти. При прослушивании радиопередачи активность головного мозга намного выше, чем при просмотре кинофильма. Данную особенность восприятия человеком информации нужно учитывать при создании обучающих систем с применением ЭВМ и речевых человеко-машинных интерфейсов.

Письменная же речь в общем смысле является «побочным» эффектом устной речи и носит второстепенный характер. Для человека наиболее естественно и эффективно получать и усваивать информацию именно в устной форме. Это связано главным образом с тем, что при устной форме общения между человеком и человеком или между человеком и машиной звуковая информация из слухового нерва попадает непосредственно в ту часть головного мозга человека, которая отвечает за речевые функции. При письменном же общении информация из зрительного нерва сначала попадает в визуальные области головного мозга, обрабатывается там, а уже только потом оттуда попадает в ту часть головного мозга, которая ответственна за речевые функции. Неэффективность второго варианта заключается в наличии большего количества звеньев, через которые должна пройти информация, а также в самом факте задействования менее приспособленной к речевым функциям зрительной памяти. Опыты показывают, что усваивать речевой материал наиболее эффективно задействуя только слуховую память без участия зрительной. В определённой степени зрительная память мешает нам в усвоении речевой информации. Это утверждение главным образом подтверждается общеизвестным фактом, что выучить иностранный язык можно в короткие сроки (от 6 мес. до 1 года) методом «интенсивного погружения в устную языковую среду». Напротив, если изучать язык самостоятельно с применением только письменной литературы (различных словарей и самоучителей), то этот процесс может затянуться на десятилетия, при этом возникает существенный риск неправильного изучения языка и формирования некорректного произношения. Данный подход обосновывает общее применение речевых технологий в информационных системах, а также частное их применение при построении обучающих систем. В данном случае в качестве «обучаемого» может выступать как человек, так и машина. В общем можно сказать, что главным плюсом зрительного восприятия информации является его скорость, а слухового восприятия – более высокая эффективность усвоения и качество самой информации. Человек может быстрым взглядом посмотреть на лист бумаги с печатным текстом, за считанные секунды пробежать глазами между его строк и сразу же мысленно представить себе общий смысл излагаемого в нём текста. Однако при таком быстром чтении возникает риск неправильного или неполного понимания смысла текста, излагаемой в нём информации, а также искажённого её представления.

Особенно эффективно при построении систем с человеко-машинным интерфейсом использовать связку технологий синтеза и распознавания речи. На Рис. 1 видны обратные связи, которые свидетельствуют о возможности осуществления контроля над системой человек-машина при использовании одних только речевых интерфейсов. Например, при условии наличия качественного синтезатора речи ЭВМ может обучать человека иностранному языку и при этом производить контроль правильности произношения иностранных слов с помощью алгоритмов распознавания речи. Авторами данного доклада ведутся разработки подобных систем с целью построения лингвистических тренажеров для обучения английскому и казахскому языкам.

Главная задача, которую приходится решать при реализации вышеописанных систем является задача качественного синтеза речи. Наиболее универсальным решением данной задачи может выступить программная реализация алгоритма синтеза речи в виде универсального модуля с целью последующего его использования в различных системах. Авторами доклада успешно произведена реализация данных алгоритмов для синтеза любых слов на английском языке по их текстовому либо транскрипционному представлению. При этом основной упор делался именно на качество осуществляемого синтеза, а также на широкий диапазон регулирования параметров синтеза. Главным побуждающим мотивом для разработки данной системы послужил произведённый анализ существующих синтезаторов речи и опыт их использования при построении лингвистических тренажеров, который показал их крайне низкое качество и очень узкий диапазон регулируемых параметров (скорость и интонация произношения). Главным образом, данная ситуация объясняется ориентированностью существующих синтезаторов на беглое чтение художественной литературы. При таком подходе многие звуки часто «проглатываются», а слова произносятся нечётко, и нет особой необходимости в широком диапазоне регулирования. Такой подход делает существующие синтезаторы малопригодными для систем обучения иностранным языкам, где на первом плане стоит именно качество синтеза.

В настоящее время для реализации алгоритмов синтеза речи широко применяются электронно-вычислительные машины (ЭВМ). До появления ЭВМ алгоритмы синтеза речи воспроизводились с применением физических моделей речевого тракта человека и продувания через них воздушного потока. Рассматриваемый алгоритм синтеза также не является исключением и реализован с применением программных и аппаратных средств ЭВМ с тем, чтобы задействовать все вытекающие отсюда преимущества:

1. Возможность использования уже имеющегося в наличии широкого спектра существующих программных и аппаратных средств ЭВМ для обработки и анализа звука.
2. Использовать развитые инструментальные средства и новейшие технологии для разработки программного обеспечения (ПО).
3. Производить синтез в режиме реального времени.
4. Передавать результаты синтеза по сетям и каналам связи на значительные расстояния за незначительное время.

5. Встраивать реализацию данного алгоритма в другие системы с целью повышения их функциональности и эффективности.
6. Распространять программную реализацию алгоритма среди миллионов существующих пользователей персональных ЭВМ путём простого копирования.

Так к первому пункту можно отнести множество существующих программ для редактирования оцифрованного звука. Наиболее известные из них: Sound Forge, Cool Edit Pro, Wave Lab. Эти программные средства обладают большим набором функциональных возможностей и позволяют осуществлять качественную обработку и редактирование звука. Их можно использовать для решения следующих задач:

1. Запись исходного звукового материала с микрофона.
2. Обработка и редактирование уже записанного звукового материала.
3. Выделение фрагментов различной длины из исходного звукового материала.
4. Нормализация и регулирование параметров звучания (громкость, скорость, высота и тембр звука).
5. Удаление дефектов звучания и шумов.
6. Наложение различных звуковых эффектов (реверберация, эхо и др.).
7. Сохранение звука в любой из доступных аудио-форматов.
8. Преобразование из одного аудио-формата в другой.
9. Использование средств для частотного и спектрального анализа звуков.

Однако здесь нужно заметить, что при построении систем синтеза речи часто приходится обрабатывать записанную речь, общая продолжительность звучания которой может достигать десятки часов. При этом звукозапись может быть фрагментирована на десятки тысяч аудио-файлов (по предложениям, словам, фонемам и т. п.). И обработка такого огромного массива данных с помощью стандартного ПО является довольно трудоёмким процессом. Данные программы имеет смысл использовать для выделения небольшого и ограниченного количества звуковых фрагментов. Например, при разработке микросегментного синтезатора речи, когда необходимо выделить только ограниченное количество базовых периодических микросегментов речи. Если же стоит задача построения высококачественного аллофонного синтезатора, то для одного только английского языка придётся выделить и проклассифицировать порядка нескольких тысяч звуковых фрагментов. Здесь уже не обойтись без собственной разработки специализированного редактора, который позволяет в автоматическом или в полуавтоматическом режиме производить выделение и классификацию звуковых фрагментов. Программное средство подобного класса было успешно нами разработано и использовано при построении системы дифонного синтеза [1].

Очень интересно решение, которое применяется в современных высококачественных синтезаторах речи на этапе выделения и классификации аллофонной базы: применяется не ручное выделение фрагментов, а задействуются специализированные распознаватели речи, которые автоматически фрагментируют и классифицируют речь диктора по заранее известному тексту и его транскрипции.

Алгоритмы синтеза речи программным методом потенциально могут быть применены практически к любому из языков, на которых говорит современное человечество. Каждый из этих языков в свою очередь имеет собственный словарный запас, лексические, фонетические и граматические особенности. Наиболее оптимальным методом решения задачи синтеза речи может стать разработка специального машинного языка, с помощью которого в универсальной форме будут описываться правила синтеза для каждого из человеческих языков с учётом его специфики.

Нужно сказать, что те вычислительные мощности и аппаратные ресурсы, которыми обладают современные ЭВМ, в достаточной степени позволяют обеспечить качественный синтез речи в режиме реального времени. Основная же сложность возникает на этапе проектирования и программной реализации данных алгоритмов.

Идеальным результатом считается такое преобразование текстовой информации, в результате которого получаемое на выходе звуковое представление речи воспринимается случайно выбранным человеком как естественная речь. В ходе прослушивания искусственно синтезированного звукового фрагмента, у человека не должны самопроизвольно возникать предположения о его неестественности и неразборчивости. Конечно, человек принимающий участие в данном опыте должен быть носителем того же языка, на котором осуществляется синтез. А вся информативная составляющая звукового фрагмента должна быть воспринята и понята человеком в максимально возможной степени. Машина должна максимально приблизиться по качеству к человеку-диктору. Это и есть та самая «идеальная планка», к которой мы стремились при разработчике своей системы синтеза речи.

Однако при оценке качества синтеза, нужно отчётливо понимать, что синтез речи– это результат высшей нервной деятельности человека [6]. И достижение указанных результатов возможно только при использовании алгоритмов аналогичных или подобных по функционалу человеческому мозгу. Пока же, в силу текущих достижений науки и техники мы можем только максимально близко приближаться к указанным результатам по качеству и возможностям синтеза. В системах синтеза речи всегда будут присутствовать вполне определённые ограничения, накладываемые функциональностью используемых алгоритмов и аппаратных мощностей.

Заключение

В работе получены следующие результаты:

1. Разработан программный модуль синтеза произвольных английских слов.
2. Отработаны два различных алгоритма синтеза: дифонный и микросегментный.
3. Реализована система выделения и автоматической классификации дифонов.
4. Разработана уникальная методика обучения иностранным языкам основанная на особенностях звукового восприятия речи человеком.
5. Модуль синтеза речи успешно применён при постороении системы автоматического обучения иностранному языку.

Список литературы

1. *Е.Н. Амиргалиев, Р.Р. Мусабаев.* Информационные технологии искусственного синтеза речи // Вестник КазНПУ. – 2007. – № 4. С. 26-34.
2. *Р.Р. Мусабаев.* Опыт использования технологии синтеза речи при построении системы контроля технологического процесса для нефтеперерабатывающего производства // Проблемы инновационного развития нефтегазовой индустрии: Сб. тр. междунар.науч.-практ. конф. – Алматы: КБТУ, 2008.
3. *Е.Н. Амиргалиев, Р.Р. Мусабаев.* Некоторые направления и задачи обработки лингвистических данных // Вестник КазНТУ. – 2007. – № 6. С. 182-187.
4. *А. Ф. Элфорд.* Загадка возникновения Homo Sapiens. Сборник «Тайное и явное», СПб., 2003 г. С. 182-187.
5. *И. Т. Канева.* Шумерский язык. Центр «Петербургское Востоковедение», 2006 г.
6. *Д. Хокинс, С. Блейксли.* Об интеллекте. Вильямс, 2007 г.

Статья поступила в редакцию 27.04.2008

УДК 004.93

МНОГОМОДАЛЬНАЯ ИДЕНТИФИКАЦИЯ ЛИЧНОСТИ ПО ФОРМЕ ЛАДОНИ И ГОЛОСУ

© Бакина И.Г., Местецкий Л.М.

Московский Государственный Университет им. М.В. Ломоносова
Факультет Вычислительной Математики и Кибернетики
Ленинские горы, 1, г. Москва, 119234, Россия

e-mail: *irina_msu@mail.ru, L.Mest@ru.net*

Abstract. The paper demonstrates the method of improving the quality of the one-mode method for personal identification by a palm shape on the base of combining it with a less reliable recognition mathod (by uttered word). Theoretical and experimantal estimates proving the efficiency of the proposed method are given.

Введение

На данный момент существует достаточно много одномодальных методов биометрической идентификации человека: анализ радужной оболочки глаз, дактилоскопия, распознавание по форме ладони, подписи, геометрии лица, голосу человека и т.д. Однако, в любом из методов идентификации есть ошибки, и задача повышения качества распознавания остается актуальной. Один из возможных вариантов решения этой проблемы — комплексный анализ по нескольким биометрическим признакам.

В данной работе рассматривается метод двумодальной идентификации личности, основанный на распознавании человека по форме ладони и голосу этого человека. Их интеграция рассматривается с точки зрения комбинирования классификаторов, когда каждый из методов обучается отдельно по своей модальности. По терминологии [3] *под модальностью понимается набор признаков, относящийся к одной группе измерений (например, параметры голоса или параметры формы ладони). Распознавание по форме ладони обладает относительно высоким, но недостаточным уровнем правильной идентификации. Однако, как показывают теоретические оценки и проведенные эксперименты, в совокупности с менее надежным методом распознавания, таковым является распознавание по голосу, качество идентификации может быть улучшено.*

1. Распознавание человека по форме ладони

Существует несколько способов описания изображений ладоней, на основе которых строятся различные критерии их сравнения. Один из таких критериев рассматривается в работе [2]. Он базируется на так называемом циркулярном разложении ладони (*Рис.1a*), описывающем ширину пальцев. Другой критерий, основанный на сравнении формы гибких объектов (*Рис.1b*), предложен в [3]. Здесь измеряется площадь симметрической разности объектов при их наложении друг на друга и "шевелении", при котором обеспечивается максимальное совпадение формы. Использование гранично-скелетного представления формы объектов позволяет построить еще

несколько количественных мер для сравнения формы ладоней. В работе [4] рассмотрены критерии, основанные на измерении длины пальцев, ширины ладони, ширины пальца вдоль его оси, а также график искривления самой этой оси. Ширина пальца описывается в виде зависимости радиуса окружности, вписанной в силуэт пальца, от расстояния между ее центром и кончиком пальца. Искривление пальца представляется в виде функции, описывающей отклонение серединной оси пальца от прямой линии.

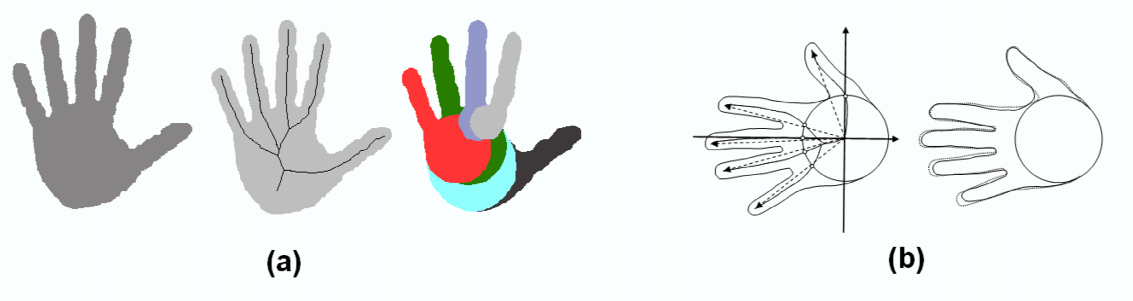


Рис. 1. (a) Растровый образ, его скелет и циркулярное разложение; (b) Представление ладони в виде гибкого объекта и сравнение таких объектов.

В данной работе за основу был взят метод распознавания человека по форме ладони, изложенный в [9]. Он включает в себя следующие основные шаги:

1. Построение гранично-скелетного представления тестового изображения ладони.
2. Сравнение полученного представления с эталонными образцами из имеющейся базы (сравнение происходит по нескольким критериям).
3. Определение Парето-оптимальных эталонных образцов ладоней (”ближайших”).
4. Классификация среди ”ближайших”.

Для интеграции данного метода с распознаванием человека по голосу выполнение всех четырех шагов не требуется, достаточно определить лишь множество ”ближайших”, не проводя дальнейшую классификацию. Фактически, рассматривается неполный вариант метода, выходом которого является список персон, чьи изображения ладони попали в число ”ближайших”, и число таких попаданий.

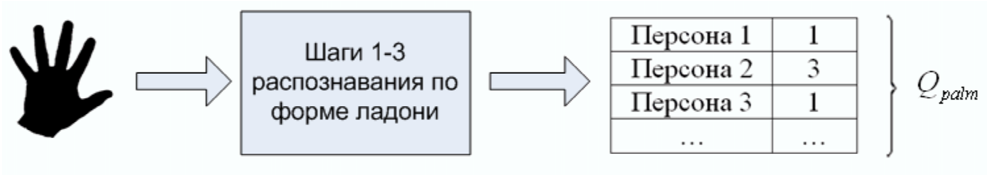


Рис. 2. Схема работы неполного метода распознавания человека по форме ладони.

Для удобства назовем полученный список персон $\mathbb{Q}_{palm}$. Заметим, что в случае, когда метод отклоняет предъявленное изображение ладони, $\mathbb{Q}_{palm} = \varnothing$.

В качестве метода распознавания по форме ладони мог быть рассмотрен любой из перечисленных выше. Единственное условие — метод должен работать с базой эталонных изображений ладоней и на выходе выдавать список, составленный из наиболее похожих изображений. Выбор метода [9] обусловлен тем, что он позволяет анализировать сразу несколько критериев, и потому множество "ближайших" изображений может быть определено более корректно.

Эксперименты по распознаванию ладоней с использованием этих критериев показали, что они довольно хорошо позволяют определить круг наиболее похожих образцов из базы эталонов. Вместе с тем, ошибки классификации остаются достаточно большими. Особенно это сказывается на ошибках ложного узнавания (*False Acceptance Rate*) в случае, когда объем базы эталонов достаточно велик.

2. Распознавание человека по голосу

Распознавание человека по голосу рассматривается в данной статье с точки зрения распознавания человека по произнесенному парольному слову. Каждая персона характеризуется парольным словом, при этом парольные слова разных людей могут совпадать. Например, в качестве парольного слова может использоваться фамилия человека. Имеется база эталонных записей парольных слов: аудиозапись парольного слова каждого человека хранится в отдельном файле в формате ".wav".

В данной работе не ставится задача распознавания человека по произносимой речи. Запись парольного слова представляет собой достаточно короткий звуковой сигнал, поэтому методы, основанные на построении модели диктора (например, метод квантования векторов [5, 6]), здесь неприменимы.

Сравнение двух записей парольных слов осуществляется на основе метода динамического выравнивания [7]. С этой целью дискретный звуковой сигнал разбивается на перекрывающиеся фреймы длины $N_t = 512$ измерений с перекрытием в 100 измерений. Дальнейшей обработке подвергаются лишь фреймы, амплитуда сигнала в которых больше заданного порога. К каждому из полученных фреймов применяется оконная функция Хэмминга, затем фрейм описывается вектором признаков $\vec{f} = (f_1, f_2, ..., f_{12})^T$, составленным из первых 12 кепстральных коэффициентов (0-ой коэффициент исключается из рассмотрения, так как он содержит информацию только об энергии сигнала). Кепстральные коэффициенты задаются следующим образом: если известен дискретный звуковой сигнал $s(t)$, а N_t — число временных отсчетов в записи, то n-ый кепстральный коэффициент может быть вычислен по формуле:

$$f_n = \mathrm{Re}\left[\frac{1}{N_t}\sum_{l=0}^{N_t-1}\left\{\ln\left(\sum_{t=0}^{N_t-1}\left|s(t)\exp(-i\frac{2\pi lt}{N_t})\right| + 1\right)\right\}\exp(i\frac{2\pi ln}{N_t})\right], n = \overline{0, N_t}$$

Расстояние между фреймами F_1 и F_2, описываемых векторами-признаками $\vec{f^1} = (f_1^1, f_2^1, \ldots, f_{12}^1)^T$ и $\vec{f^2} = (f_1^2, f_2^2, \ldots, f_{12}^2)^T$ соответственно, определяется по формуле:

$$D(F_1, F_2) = \sqrt{\sum_{i=1}^{12} (f_i^1 - f_i^2)^2}$$

Как уже упоминалось, в качестве меры близости двух звуковых сигналов рассматривается расстояние, полученное по методу динамического выравнивания.

В работе также рассматривалось признаковое описание звукового сигнала на основе спектральных коэффициентов, однако для данного метода они оказались недостаточно информативными. Их использование приводило к значительному увеличению ошибок распознавания по сравнению с случаем, когда в качестве признакового описания рассматривались кепстральные коэффициенты.

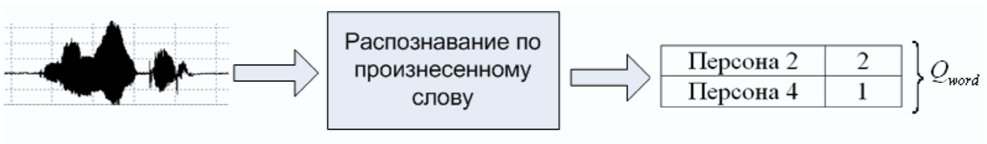


Рис. 3. Схема работы метода распознавания человека по произнесенному слову ($k = 3$).

Общая схема работы метода заключается в следующем. При произнесении парольного слова звуковой сигнал сравнивается со всеми эталонными образцами из имеющейся базы, после чего они упорядочиваются по возрастанию меры близости с тестовым образцом. Из полученного списка отбираются k наиболее похожих эталонных записей, и строится множество персон $\mathbb{Q}_{word}$, аналогичное $\mathbb{Q}_{palm}$. В отличие от распознавания по форме ладони множество $\mathbb{Q}_{word}$ не может быть пустым (так как отбор похожих персон происходит всегда).

Эксперименты показывают, что при сравнении тестового образца с базой эталонов, вероятность того, что персона попадет в начало упорядоченного списка, весьма высока. Это позволяет рассчитывать на низкий уровень ошибок ложного отказа (*False Rejection Rate*). Однако в случае, когда персона не представлена в базе, для нее все равно найдется достаточно похожий эталон. Это происходит потому, что две записи парольного слова одного человека зачастую оказываются менее похожими, чем записи парольных слов двух разных людей (с точки зрения меры близости, рассмотренной выше). Поэтому не удается уверенно классифицировать ситуацию, когда тестовый образец принадлежит "чужой" персоне. Это приводит к очень высокому уровню ошибок ложного узнавания.

3. Комбинирование классификаторов

Целью комбинирования классификаторов является преодоление указанных недостатков каждого из описанных методов.

Для распознавания человек предъявляет системе свою ладонь и произносит парольное слово. Отдельно по каждому из этих биометрических признаков происходит распознавание, и формируются два множества $\mathbb{Q}_{palm}$ и $\mathbb{Q}_{word}$, описанные выше. Далее ищется пересечение этих множеств $\mathbb{Q} = \mathbb{Q}_{palm} \bigcap \mathbb{Q}_{word}$, при этом возможны следующие варианты:

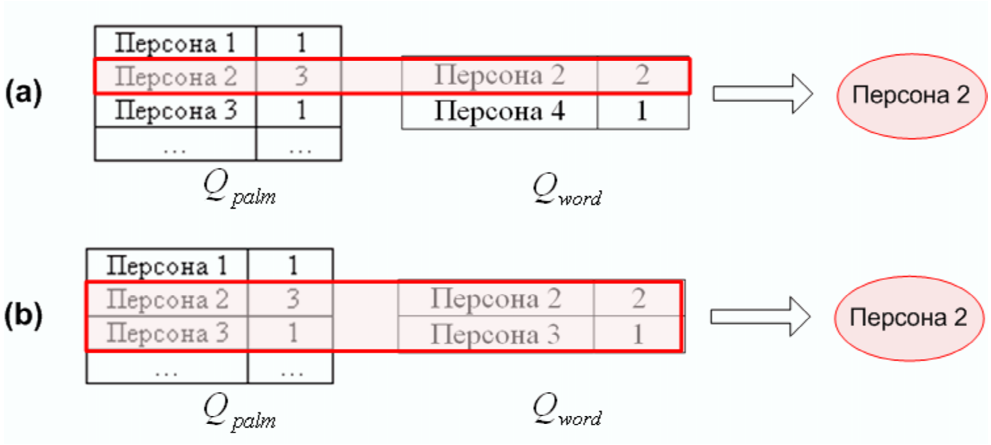


Рис. 4. Комбинирование классификаторов: (a) $|\mathbb{Q}| = 1$; (b) $|\mathbb{Q}| > 1$.

1. $\mathbb{Q}_{palm} = \varnothing$, и в этом случае человек признается как "чужой".
2. Ситуация $|\mathbb{Q}| = 1$ представлена на *Рис.4a*. Человек идентифицируется как персона, попавшая в пересечение.
3. В случае $|\mathbb{Q}| > 1$ считается общее число попаданий в группу "ближайших" по ладошке и по голосу одновременно. Например, для ситуации на *Рис.4b* для *Персоны 2* эта величина равна $n_1 = 3+2 = 5$, а для *Персоны 3* - $n_2 = 1+1 = 2$. Так как $n_1 > n_2$, то человек идентифицируется как *Персона 2*. В случае, когда для нескольких персон эти величины совпадают, система признает человека как "своего", но идентифицировать отказывается.

Обозначим через FRR_{palm} и FAR_{palm} соответственно *False Rejection Rate* и *False Acceptance Rate* для неполного метода распознавания человека по форме ладони, а через FRR_{word} и FAR_{word} - по произнесенному слову. Пусть u_o — "свой" человек, а u_s — "чужой". Для рассмотренных методов распознавания эти величины определяются следующим образом:

1. $FRR_{palm} = \mathsf{P}(\{u_o \notin \mathbb{Q}_{palm}\})$;
2. $FAR_{palm} = \mathsf{P}(\{\mathbb{Q}_{palm} \neq \varnothing\} |\ u_s)$;
3. $FRR_{word} = \mathsf{P}(\{u_o \notin \mathbb{Q}_{word}\})$;
4. $FAR_{word} = \mathsf{P}(\{\mathbb{Q}_{word} \neq \varnothing\} |\ u_s) = 1$, так как множество всегда непустое.

Оценим FAR и FRR для рассматриваемой двумодальной системы. Как отмечалось выше, человек определяется "чужим" в случае, когда он не попал в пересечение списков $\mathbb{Q} = \mathbb{Q}_{palm} \bigcap \mathbb{Q}_{word}$ персон для обоих методов. Поэтому:

$$FRR = \mathsf{P}(\{u_o \notin \mathbb{Q}\}) = \mathsf{P}(\{u_o \notin \mathbb{Q}_{palm} \bigcap \mathbb{Q}_{word}\}) =$$
$$= 1 - \mathsf{P}(\{u_o \in \mathbb{Q}_{palm}\}) \cdot \mathsf{P}(\{u_o \in \mathbb{Q}_{word}\}) =$$
$$= FRR_{palm} + FRR_{word} - FRR_{palm} \cdot FRR_{word} \quad (1)$$

"Чужой" признается "своим" в случае, когда множество оказывается непустым, оценка для FAR имеет следующий вид:

$$FAR = FAR_{palm} \cdot \frac{k}{n} \quad (2)$$

В этой формуле k — число отбираемых эталонных записей парольных слов, а n — число персон в базе. Здесь предполагается, что вероятности попадания парольных слов в k "ближайших" равны, определяются лишь количеством парольных слов в списке и не зависят от самих слов.

4. Эксперименты

Рассматривалась группа людей из 24 человек. Для каждого из них 12 раз производились снимок ладони и запись парольного слова. Полученные пары вида "изображение – запись" разделялись на две части: эталоны и контроль. К эталонам были отнесены 7 из сформированных пар, оставшиеся 5 — к контролю.

При оценивании FRR рассматривались все эталонные и контрольные пары: каждая контрольная пара подавалась на вход программе один раз, после этого подсчитывалось число пар, ошибочно отнесенных к классу "чужой".

При оценивании величины FAR также рассматривались все эталонные и контрольные пары. Однако сравнение предъявляемой контрольной пары происходило лишь с эталонными парами, не относящимися к тому же человеку. В эксперименте подсчитывалось число пар, ошибочно пропущенных системой.

Результаты эксперимента представлены в *Таблице 4*. В последних двух колонках содержатся теоретические и практические оценки FRR и FAR. Теоретические оценки получены по формулам (1) и (2).

Таблица 1. *Результаты эксперимента.*

	Распознавание по ладошке	Распознавание по слову	Теоретическая оценка	Экспериментальная оценка
FRR	2.5%	5.8%	8.2%	8.33%
FAR	30%	100%	3.75%	3.33%

Из таблицы видно, что использование менее надежного метода распознавания позволяет существенно снизить FAR, правда, за счет некоторого увеличения FRR.

Заключение

Проведенные эксперименты и полученные теоретические оценки показывают, что предложенный в работе метод комбинирования классификаторов действительно позволяет повысить надежность распознавания. Прослеживается значительное снижение величины FAR, что согласуется с полученными теоретическими оценками. Наблюдаемое при этом увеличение FRR оказывается не настолько весомым.

Конечно, полученные оценки FAR и FRR все еще остаются значительными. Это объясняется тем, что исходные одномодальные методы распознавания, по форме ладони и произнесенному слову, изначально обладают высоким уровнем ошибок ложного узнавания и ложного отказа. В данной работе большее внимание уделялось изучению не самих методов распознавания, а проверке гипотезы о том, что предложенный подход к их комбинированию приводит к улучшению качества идентификации. В дальнейшем планируется обратиться к анализу указанных методов и поработать над увеличением надежности каждого из них в отдельности.

Работа выполнена при поддержке РФФИ, гранты 08-01-00670 и 08-07-00305-а.

Список литературы

1. *Mestetskiy L.M.* Shape comparison of flexible objects similarity of palm silhouettes // Proceedings of the 2nd International conference on computer vision theory and applications (VISAPP 2007), Volume IFP/IA, Barcelona, Spain, 2007, P .390-393.
2. *Mestetskiy L.M., Semenov A.B.* Palm shape comparison based on fat curves // Proceedings of 7th International conference on Pattern recognition and image analysis: new information technologies, St.Petersburg, 2004, P .788-791.
3. *Местецкий Л.М.* Сравнение изображений гибких объектов на основе нормализации // Труды 17 международной конференции ГРАФИКОН-2007, Москва, ВМК МГУ, 2007, С. 203-210.
4. *Mekhedov I.S., Mestetskiy L.M.* Construction of a classifier for person biometric identification using a palm shape //Proceedings of the Ninth International conference on Pattern recognition and information processing (PRIP'2007), Volume I, Minsk, Belarus, May 22-24, 2007, P. 290-294.
5. *Evgeny Karpov* Real-Time Speaker Identification // University of Joensuu, Department of Computer Science, Master's Thesis.
6. *Tomi Kinnunen, Ismo Karkkainen and Pasi Franti* Is Speech Data Clustered? — Statistical Analysis of Cepstral Features // In EUROSPEECH-2001, 2627-2630.
7. *Sergios Theodoridis, Konstantinos Koutroumbas* Pattern Recognition // second edition, Elsevier 2003.
8. *Татарчук А.И., Елисеев А.П., Моттль В.В* Комбинирование классификаторов и потенциальных функций в многомодальном распознавании образов // Доклады Всероссийской конференции ММРО-13, 2007, С. 220-222.

Статья поступила в редакцию 25.04.2008

УДК 62-50

ПРОГРАММНО-АЛГОРИТМИЧЕСКИЙ КОМПЛЕКС СТРУКТУРНО-КЛАССИФИКАЦИОННОГО АНАЛИЗА СЛОЖНООРГАНИЗОВАННЫХ ДАННЫХ [1]

© Бауман Е.В., Дорофеюк А.А., Дорофеюк Ю.А., Киселёва Н.Е.

Институт проблем управления РАН

Abstract. In this paper the conception of the structure-ranging analysis methods application is discussed, among them were discussed such methods as: parameter structuring; objects classification; dynamic ranging analysis; piecewise approximation of complex dependences. This conception is realized in the man-machine system with intellectual interface for user. During the complex development a special attention was given to problems, in which researched objects have clear-cut areal structure.

Введение

В работах [1-4] предложены принципы построения и использования алгоритмов структурного анализа сложноорганизованных данных. В работе предлагается концепция применения разнообразных методов структурно-классификационного анализа, включающая методы: структуризации набора исходных параметров; структуризации множества объектов с помощью алгоритмов автоматической классификации; динамического классификационного анализа, позволяющие анализировать поведение объектов в многомерном пространстве траекторий; анализа сложных, нелинейных зависимостей с помощью алгоритмов кусочной аппроксимации. Реализация этой концепции подразумевает создание человеко-машинной системы с интеллектуальным интерфейсом для пользователя – предметника, позволяющая формировать пользователем схемы использования алгоритмов, текущего анализа промежуточных результатов, наглядного их отображения.

1. Постановка задачи классификационного анализа данных

Структура исходных данных в задачах классификационного анализа. Функционирование любой сложной системы описывается состоянием составляющих её элементов (объектов) и их взаимодействием. Соответственно, данные о системе представляют собой либо таблицу значений некоторых параметров, характеризующих состояние объектов, либо таблицу, отражающую взаимодействие между объектами, либо, наконец, таблицу связи между параметрами. При описании концепции системы, опирающейся на методологию классификационного анализа, ограничимся таблицами первого типа: «объекты-параметры». Заметим, что данные о системе обычно фиксируются ни в один момент времени, а многократно в течение некоторого периода (например, ежемесячно, ежеквартально или ежегодно в течение несколько лет работы системы). Характерной особенностью исследования реальных систем является невозможность получения значений всех параметров по всем объектам во все моменты времени, что приводит к пропускам в данных.

[1]Работа выполнена при частичной финансовой поддержке РФФИ, проекты 08-07-00349-а, 08-07-00347-а.

Итак, исходный материал о функционировании исследуемой системы представляет собой куб данных «объекты-параметры-время» $\left\| x_i^{(j)(t)};\ i=\overline{1,n};\ j=\overline{1,k};\ t=\overline{1,T} \right\|$, причем в данных возможны пропуски (здесь $x_i^{(j)(t)}$- значение j-го параметра на i-м объекте в момент времени t) .

Основная цель классификационного анализа заключается в выявлении наиболее общих закономерностей функционирования анализируемой системы: структуризация набора параметров для выявления групп тесно связанных параметров и построения небольшого числа интегральных показателей; структуризация множества объектов, для чего необходимо выделить в пространстве выбранных параметров области, отражающие типовые режимы деятельности отдельных объектов системы; построение кусочных моделей зависимости некоторых выходных параметров от входных; выявление динамических свойств системы – выделение характерных траекторий изменения параметров во времени, выявление зависимостей между параметрами с учетом временного сдвига и т.д.

2. Состав программно-алгоритмического комплекса (ПАК)

Для реализации вышеперечисленных целей был разработан человеко-машинный программно-алгоритмический комплекс (ПАК) с интеллектуальным интерфейсом для пользователя-предметника. В него входит база данных, в которой хранится исходный куб данных, наименования объектов, названия параметров, результаты всех этапов обработки. Для реализации алгоритмов структурно-классификационного анализа данных ПАК содержит 5 основных обработочных модулей**:** предварительной обработки и фильтрации исходных данных, классификационного анализа параметров, классификации объектов, анализа множества полученных классификаций, кусочной аппроксимации. ПАК позволяет постоянно обновлять куб данных, при этом все полученные ранее результаты классификационного анализа распространяются на новые данные. ПАК оснащён дружественным интерфейсом, который включает структуру меню для выбора режимов обработки, вводные формы для определения свободных параметров и т.д. ПАК позволяет отображать исходные данные и результаты анализа в наглядной форме, в том числе в виде географической карты, гистограмм, графиков и т.д.

Работа с комплексом организована в виде диалога. На каждом этапе пользователю предоставляется возможность выбрать один из основных модулей обработки. В то же время ему даётся рекомендация, - какую обработку целесообразно проводить на данном этапе. Результаты применения программ каждого модуля заносятся в базу данных и являются исходными данными для работы других модулей.

На рис. 1. представлена общая блок-схема ПАК, отражающая рекомендуемую последовательность применения основных модулей (блоков). Ниже описывается каждый из основных блоков отдельно.

2.1. **Предобработка.** До проведения структурно-классификационного анализа необходима предварительная обработка: статистический анализ, выявление грубых ошибок в данных, заполнение пропусков в данных и т.п.

Рис. 1. Блок-схема ПРОГРАММНО-АЛГОРИТМИЧЕСКОГО КОМПЛЕКСА

Как для статистического, так и для структурно-классификационного анализа стандартным видом данных являются таблицы типа «объекты-параметры».

Существует два способа развертки исходного куба данных в таблицу такого рода. Первый способ заключается в том, что в качестве объектов таблицы рассматриваются пары «объект - момент времени» исходного куба данных (например, данные об одном и том же объекте за t лет рассматриваются как данные о t разных объектах). Набор параметров при этом остается без изменений. Такую развертку куба будем обозначать через $\mathrm{T}^{\text{пар}}_{\text{об-вр}}$. При втором способе множество объектов таблицы совпадает с множеством объектов исходного куба данных, а в качестве параметров

рассматриваются пары «параметр - момент времени». Соответствующая развертка обозначается как $T_{об}^{пар\text{-}вр}$.(см. рис.2).

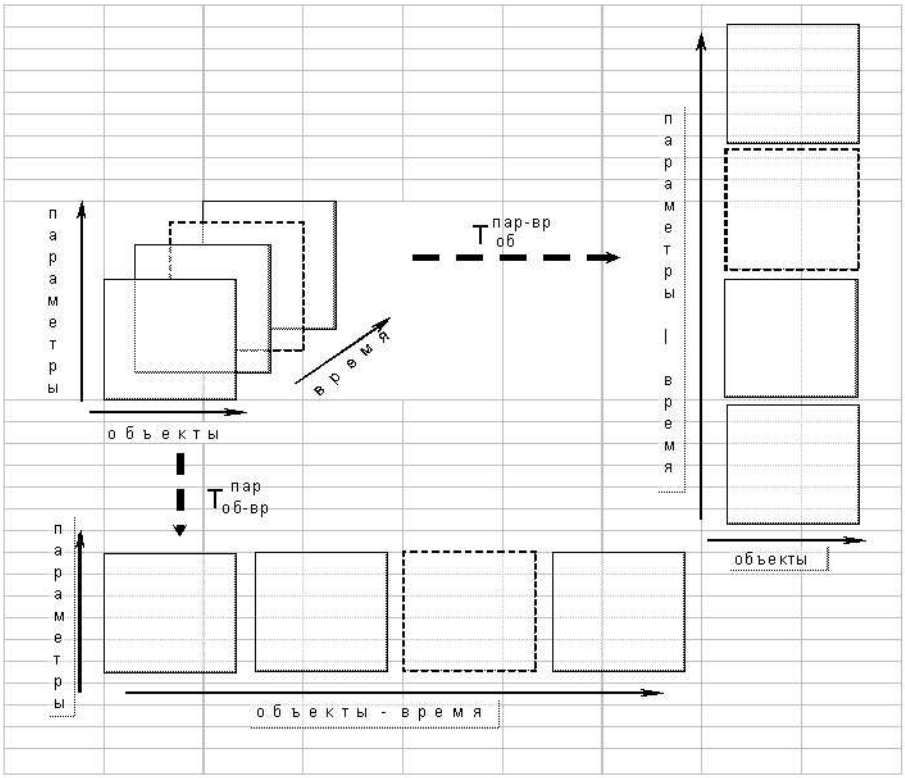


Рис. 2. Варианты развёртки куба данных

С помощью статистического анализа определяются средние значения параметров, их дисперсии и корреляции между ними. Такой анализ позволяет оценивать как независящие от времени статистические характеристики параметров, так и связи между параметрами, учитывая сдвиги во времени. Далее выявляются грубые ошибки в данных, т.е. значения параметров, сильно отличающиеся от средних значений.

Алгоритмы классификационного анализа модифицированы так, чтобы они работали и на данных с пропусками. С другой стороны, часто удобно заполнить пропуски до начала основного классификационного анализа.

Для заполнения пропусков предлагается следующая процедура:

Сначала производится группировка параметров, а затем по каждой группе параметров делается классификация объектов. Пропущенное значение $x_i^{(j)}$ таблицы заполняется следующим образом: находится группа, которой принадлежит j-й параметр; в соответствующей классификации находится класс, в котором лежит i-й объект, и в качестве $x_i^{(j)}$ берется среднее значение j-ого параметра по этому классу.

2.2. **Структуризация исходных параметров.** Опыт использования алгоритмов структурно-классификационного анализа показывает, что классификация по всем исходным параметрам не всегда приводит к желаемым результатам. Действительно,

при сравнительно небольших выборках экспериментальных наблюдений и наличии помех (ошибки в определении значений параметров, сознательное искажение информации и т.д.) использование для классификации большого числа входных параметров приводит к сильному «перемешиванию» классов, а сами классы при этом плохо поддаются интерпретации. По этой причине классификацию целесообразно проводить не в исходном пространстве, а в пространстве наиболее существенных (информативных) параметров, имеющем значительно меньшую размерность. Для выявления структуры исходного набора параметров вне зависимости от времени используется развертка исходного куба данных в таблицу типа $\mathrm{T}^{\text{пар}}_{\text{об-вр}}$. Структуризация проводится с помощью алгоритмов экстремальной группировки параметров [1],[5]. Определяется, нужна ли группировка с фоновой группой или без неё (т.е. отсекать или нет сильно шумящие параметры). Результатом группировки будут группы параметров и факторы – обобщённые характеристики групп. На основе результатов группировки строятся интегральные показатели исследуемой системы. В качестве таковых выбираются либо сами факторы, либо параметры в определённом смысле ближайшие к факторам. Основное условие – они должны быть легко интерпретируемы. Для удобства использования интегральных показателей по каждому из них делается одномерная классификация объектов. Благодаря этому интегральный показатель преобразуется в качественный, так как его значения можно качественно характеризовать в терминах типа «низкие», «средние», «высокие».

Другое применение метода экстремальной группировки – выбор информативных параметров для других методов анализа. Выбирается либо набор факторов, либо набор, в который входят по одному или небольшое число параметров из каждой группы. Обычно окончательное решение о выборе информативных параметров производится экспертом-пользователем.

Для определения зависимостей между параметрами с учётом времени, используется развертка куба данных, в которой за параметры принимаются пары «параметр-момент времени», т.е. рассматривается таблица вида $\mathrm{T}^{\text{пар-вр}}_{\text{об}}$.

2.3. **Классификация объектов.** Ключевым этапом структурного анализа исследуемой системы является структуризация множества элементов системы (объектов). В ПАК для решения этой задачи предназначены алгоритмы автоматической классификации. При этом используется целый ряд проблемно ориентированных алгоритмов – в детерминированной и размытой постановке, с фоновым классом, итерационных (на каждом шаге обрабатывается только один объект выборки), параллельных (когда все объекты выборки обрабатываются одновременно) и т.д. [1].

Вначале производится выбор пространства показателей, которое является исходным для классификации объектов.

Для нахождения независимых от времени режимов функционирования объектов исследуется таблица, у которой набор параметров совпадает с исходным, а роль объекта играет пара «объект-момент времени» (таблица $\mathrm{T}^{\text{пар}}_{\text{об-вр}}$). Классификация объектов из таблицы $\mathrm{T}^{\text{пар-вр}}_{\text{об}}$, у которой множество объектов совпадает с исходным,

а в качестве параметров рассматриваются пары «параметр-момент времени», приводит к нахождению типичных траекторий изменения параметров во времени. Для структуризации объектов изменяющихся во времени используются методы динамического классификационного анализа, позволяющие анализировать поведение объектов в многомерном пространстве траекторий. Здесь также представлен достаточно широкий спектр проблемно ориентированных алгоритмов – по типу обрабатываемых траекторий, по используемым мерам связи (близости) траекторий и т. д. [2].

До применения выбранного алгоритма классификации необходимо определить: вид функционала; строится ли классификация с фоновым классом или без (т. е. отбрасываются ли далекие объекты); тип размытости**:** четкая, размытая, с размытыми границами, четкая с размытым фоном, размытая с четким фоном и т. д. [1].

Результатом классификации являются функции принадлежности объектов к классам и описание самих классов.

2.4. **Структуризация результатов классификации.** Практически все алгоритмы структурно-классификационного анализа содержат свободные параметры, значения которых трудно выбрать заранее из теоретических соображений. Кроме того, эти алгоритмы находят лишь локальный экстремум соответствующего критерия, поэтому результаты их работы зависят от начальных условий (начального разбиения объектов на классы и параметров на группы). В связи с этим, при решении практических задач свободные параметры алгоритмов, начальные условия, а часто и состав переменных, образующих исходное пространство, варьируются в широких пределах. Это приводит к тому, что образуется целое множество различных вариантов классификации. Число классификаций часто оказывается столь большим, что для их анализа приходится применять машинные методы, вводя меру близости между классификациями и разбивая их на группы «похожих» классификаций. Одна из таких мер предложена в [6]. Применение компьютерных методов обработки результатов классификации существенно облегчает их дальнейший неформальный анализ.

2.5. **Кусочная аппроксимация.** После структуризации исходных данных как по параметрам, так и по объектам можно, в случае необходимости, приступить к построению моделей зависимости выходных показателей от входных. Важной частью ПАК является модуль анализа сложных, нелинейных зависимостей и формирования математических моделей различных процессов.

Для этой цели используются алгоритмы кусочной аппроксимации [1],[4]. Предусмотрен специальный режим одномерной кусочной аппроксимации, когда соответствующий алгоритм обеспечивает получение глобально-оптимального решения задачи [4]. Классы объектов, полученные автоматической классификацией, используются либо непосредственно как области действия отдельных локальных моделей (при двухступенчатом методе аппроксимации), либо как начальные условия для алгоритма нахождения таких областей (при одноступенчатом алгоритме аппроксимации). Обычно для кусочной аппроксимации используется первый вариант развертки исходного куба данных, т.е. в качестве объектов рассматриваются пары «объект-момент времени». Иногда бывает удобно для таких пар учитывать время в качестве входного

параметра при построении кусочной модели. Для построения кусочной аппроксимации необходимо определить: выходной параметр; пространство входных параметров; число классов; нужен ли фоновый класс; тип размытости классификации [4].

Заключение

Разработанный программно-алгоритмический комплекс предназначен для решения задач анализа сложноорганизованных многомерных данных, а также для поддержки принятия решений при анализе и реформировании крупномасштабных систем управления. Предусмотрена возможность эксплуатация системы пользователями двух уровней. На первом уровне пользователь-аналитик формирует модели исследуемой системы, в том числе: набор интегральных показателей, пространство, в котором проводится классификация, результирующая классификация, результирующие кусочные прогностические модели. На втором уровне пользователь-предметник использует полученные на первом уровне модели для решения задач оперативного управления.

Разработанный ПАК использовался для решения многих прикладных задач, в том числе в задачах управления региональным здравоохранением [7], региональными пассажирскими автоперевозками [8], а также при обработке сложноорганизованных данных (например, при обработке пульсовых сигналов лучевой артерии [9]).

Список литературы

1. *Бауман Е.В., Дорофеюк А.А.* Классификационный анализ данных // Избранные труды Международной конференции по проблемам управления. Том 1. / – М. : СИНТЕГ. 1999. – С. 62-67.
2. *Чернявский А.Л., Бауман Е.В., Дорофеюк А.А.* Методы динамического классификационного анализа данных // Искусственный интеллект, № 2. 2002. – С. 290-298.
3. *Бауман Е.В., Дорофеюк А.А., Чернявский А.Л.* Методы структурной обработки эмпирических данных // Измерения, контроль, автоматизация. 1985. № 3. - С. 64-69.
4. *Бауман Е.В., Дорофеюк А.А., Корнилов Г.В.* Алгоритмы оптимальной кусочно-линейной аппроксимации сложных зависимостей // Автоматика и телемеханика. 2004. № 10. – С. 163-171.
5. *Браверман Э.М., Мучник И.Б.* Структурные методы обработки эмпирических данных // – М.: Наука. 1983. – 464 с.
6. *Бауман Е.В.* Структуризация номинальных признаков в задачах экспертизы. – В кн.: Экспертные оценки в задачах управления. Сборник трудов. // – М.: Институт проблем управления. 1982. – С. 22-26.
7. *Бауман Е.В., Дорофеюк А.А., Чернявский А.Л., Медик В.А.* Классификационные методы в аналитических задачах регионального управления // Труды Института проблем управления РАН. Том X. // – М.: ИПУ РАН. 2000. – С. 38-40.
8. *Блудян Н.О., Чернявский А.Л.* Структурные методы совершенствования управления региональным пассажирским автотранспортом // М.: «Альфа-Мир». Серия Транспорт. 2002. – 127 с.
9. *А.А.Десова, А.А.Дорофеюк, В.В.Гучук, Ю.А.Дорофеюк, И.В.Покровская* Процедуры классификационного анализа в задаче формирования информативных признаков при исследовании ритмической структуры биосигнала. // Автоматика и телемеханика. 2008. № 6. – с. 143-152.

Статья поступила в редакцию 28.04.2008

УДК 62-50

МЕТОДЫ КУСОЧНО-ЛИНЕЙНОЙ АППРОКСИМАЦИИ И ИХ ИСПОЛЬЗОВАНИЕ В ЗАДАЧАХ УПРАВЛЕНИЯ[1]

© Бауман Е.В., Гольдовская М.Д., Дорофеюк Ю.А.

Институт проблем управления РАН

Abstract. This paper is dedicated to the methods of solution piecewise-linear approximation problem, which were developed on the base of general approach to range data analysis. The main idea of piecewise approximation of complex dependence is to divide the in-parameter space into areas, so that within the bounds of each of these areas it will be able to approximate the complex dependence in the whole space with linear function.

1. Постановка задачи кусочно-линейной аппроксимации

1.1. **Случай чёткой классификации.** Рассматривается задача построения модели зависимости выходного показателя y (объясняемой переменной) от вектора входных показателей $x = \left(x^{(1)}, ..., x^{(k)}\right) \in X = \mathbf{R}^k$ (объясняющих переменных). Модель строится по выборке из n объектов, каждый из которых описывается вектором $(y_t, x_t) = \left(y_t, x_t^{(1)}, \ldots, x_t^{(k)}\right) \in \tilde{X} = \mathbf{R}^{k+1}$.

Классическая схема кусочно-линейной аппроксимации состоит в следующем [1]:

Пространство с помощью одного из алгоритмов автоматической классификации [2] разбивается на r классов $(H_1, ..., H_r)$. Затем в каждом классе по методу наименьших квадратов строится линейная регрессия выходного показателя y от вектора входных показателей x. В i-м классе находятся такой вектор коэффициентов $\mathrm{c}_i = \left(c_i^{(1)}, ..., c_i^{(k)}\right)$ и константа d_i линейной функции$((c_i, x) + d_i) = d_i + \sum\limits_{j=1}^{k} c_i^j x^j$, которые минимизируют функционал $K_i = \sum\limits_{x_t \in H_i} (y_t - ((c_i, x_t) + d_i))^2$. Задача кусочно-линейной аппроксимации состоит в нахождении такого разбиения на классы, для которого сумма квадратов невязок по моделям всех классов была бы минимальна. Другими словами, необходимо найти такую классификацию $(H_1, ..., H_r)$ и такие векторы коэффициентов c_i и константы d_i, для которых функционал $I = \sum\limits_{i=1}^{r} \sum\limits_{x_t \in H_i} (y_t - ((c_i, x_t) + d_i))^2$ принимал бы минимальное значение или функционал $J_{\mathrm{KA}} = -I = -\sum\limits_{i=1}^{r} \sum\limits_{x_t \in H_i} (y_t - ((c_i, x_t) + d_i))^2$принимал бы максимальное значение. Последний функционал является частным случаем функционала классификационного анализа общего вида [2]: $J(H, A) = \sum\limits_{x \in X} \sum\limits_{i=1}^{r} K(x, \alpha_i)\, \varphi(h_i(x))$.

[1]Работа выполнена при частичной финансовой поддержке РФФИ, проекты 08-07-00349-а, 08-07-00347-а.

Классификацию $H = (H_1, ..., H_r)$ будем задавать через вектор-функцию принадлежностей $H(x) = (h_1(x), ..., h_r(x))$. Тогда функционал I можно переписать в следующих двух эквивалентных записях:

$$I_1 = \sum_{i=1}^{r} \sum_{t=1}^{n} [y_t - ((c_i, x_t) + d_i)]^2 h_i(x), \tag{1}$$

$$I_2 = \sum_{t=1}^{n} \left[y_t - \sum_{i=1}^{r} ((c_i, x_t) + d_i)\ h_i(x) \right]^2. \tag{2}$$

Когда классификация чёткая (т.е. каждый объект однозначно относится к одному из классов), функционалы (1) и (2) совпадают. Однако интерпретируются они поразному. При минимизации функционала (1) линейные модели зависимостей строятся в каждом классе в отдельности, а затем суммируются квадраты отклонения ошибок. В функционале (2) выражение $\sum\limits_{i=1}^{r} ((c_i, x_t) + d_i)\ h_i(x)$ можно считать кусочно-линейной моделью выходного показателя y.

Решению задачи построения кусочно-линейной зависимости в такой постановке посвящены, например, публикации [1, 3, 4].

Основной сложностью решения данной задачи является то, что при минимизации функционалов (1) и (2) по классификации и по коэффициентам модели решающие правила оптимальной классификации $H = (H_1, ..., H_r)$ записываются в терминах не только входных, но и выходного показателя. Это существенно уменьшает возможность использования построенной модели для прогноза. Приходится использовать различные варианты проекции классификации в расширенном пространстве $\tilde{X}$ на пространство X входных показателей. Подобное проектирование областей на пространство меньшей размерности приводит к появлению в пространстве входных показателей зон, в которых одновременно могут действовать модели как одного, так и другого класса. Именно исходя из этого, при построении кусочных моделей по существу возникает размытость между классами.

1.2. **Случай размытой классификации.** Рассмотрим возможность использования в задаче кусочной аппроксимации размытой классификации. Будем задавать классификацию через вектор-функцию принадлежностей $H(x) = (h_1(x), ..., h_r(x))$, удовлетворяющую ограничению:

$$\sum_{l=1}^{r} h_i(x) = 1; \quad h_i(x) \geq 0,\ x \in X,\ l = 1, ..., r \tag{3}$$

В данном случае функционалы (1) и (2) не совпадают, и их минимизация приводит к разным результатам. Так как функционал $I_1 = I_1(c_i, d_i, i = 1, ..., r; H(x))$ линеен по $H(x)$, то функционал $\hat{I}_1 = -\min\limits_{c_i, d_i} I_1(c_i, d_i, i = 1, ..., r; H(x))$ является выпуклым по $H(x)$. Отсюда следует, что оптимальная классификация $H(x)$ лежит на границе допустимой области, т.е. в ограничении все $h_i(x)$ равны либо 1, либо 0, что

соответствует четкой классификации. Следовательно, в оптимальной кусочной аппроксимации для функционала (1) при ограничениях (3) классификация - чёткая.

Рассмотрим функционал (2). Достаточно легко показать, что, если на $h_i(x)$ не накладывать дополнительные ограничения, то за счёт большого количества степеней свободы можно всегда точно аппроксимировать все значения y на данной выборке.

Для того чтобы использовать функционал (2) в размытом варианте, необходимо ограничить класс разрешённых функций $h_i(x)$. В соответствии с общей методикой обобщённого среднего [2] для получения кусочной аппроксимации с размытой классификацией функционал (1) необходимо модифицировать следующим образом:

$$I_3 = \sum_{i=1}^{r} \sum_{t=1}^{n} [y_t - ((c_i, x_t) + d_i)]^2 \varphi(h_i(x)). \tag{4}$$

Здесь $\varphi(h)$ – одна из ниже перечисленных функций: 1) $\varphi_1(h) = h$ приводит к чёткой кусочной аппроксимации (функционалы (1) и (2) совпадают); 2) $\varphi_2(h) = (h)^t, t > 1$, приводит к размытой кусочной аппроксимации с функцией принадлежности $h_i(x)$; 3) $\varphi_3(h) = t - \sqrt{t^2 - (2t-1)h}$, $t > 1$ приводит к кусочной аппроксимации с классификацией с размытыми границами (размываются лишь границы классов).

Поскольку функционал (3) является частным случаем функционала классификационного анализа общего вида, то для его оптимизации можно использовать общий алгоритм классификационного анализа [2] - последовательное применение двухэтапной процедуры: 1) фиксируется вектор-функция $H(x)$, для неё находятся оптимальные значения коэффициентов модели $c_i, d_i, i = 1, ..., r$; 2) фиксируются коэффициенты $c_i, d_i, i = 1, ..., r$ и для них находится оптимальная вектор-функция $H(x)$. Сходимость этого алгоритма следует из сходимости алгоритма [2].

Как и для чёткого случая, недостатком такого подхода является то, что в решающие правила аппроксимации входят не только входные показатели, но и выходной.

1.3. **Случай с ограничениями на класс решающих правил.** В ряде работ [4, 5] предлагается строить кусочную аппроксимацию так, чтобы классификация производилась по одному набору показателей, а аппроксимация в каждом классе – по другому. Будем считать, что кроме пространства X есть ещё пространство $Z = \mathbf{R}^s$, в котором производится классификация объектов. Дальнейшие рассуждения не изменятся, если часть показателей в пространствах X и Z будут одни и те же.

Считается, что каждый из n объектов исходной выборки, описывается $k + s + 1$ параметром, т.е. вектором $(y_t, x_t, z_t) = \left(y_t, x_t^{(1)}, ..., x_t^{(k)}, z_t^{(1)}, ..., z_t^{(s)}\right) \in \mathbf{R}^{k+s+1}$.

Самый используемый критерий качества классификации – критерий средневзвешенной дисперсии, который для пространства Z запишем как функционал $J(\alpha_1, ..., \alpha_r, H(z)) = \sum_{i=1}^{r} \sum_{t=1}^{n} (z_t - \alpha_i)^2 \varphi(h_i(z_t))$, зависящий от эталонов классов $\alpha_1, ..., \alpha_r$ и вектор-функции принадлежностей $H(z)$. Считается, что эталоны классов могут быть произвольными точками пространства Z, вектор-функция $H(z)$ удовлетворяет условиям (3), а функция φ равна либо φ_1, либо φ_2, либо φ_3.

Минимизация этого функционала производится как по классификации $H(z)$, так и по набору эталонов классов $A = (\alpha_1, ..., \alpha_r)$. В оптимальном случае эталон i-го класса записывается в виде $\alpha_i = \frac{\sum\limits_{t=1}^{n} z_t \varphi(h_i(z_t))}{\sum\limits_{t=1}^{n} \varphi(h_i(z_t))}$, т.е. совпадает с центром класса.

Если $\varphi(t) = \varphi_1(t)$, то итерационный алгоритм минимизации этого функционала совпадает с известным алгоритмом ISODATA [2]. Если $\varphi(t) = \varphi_2(t)$, то – с размытым вариантом ISODATA. Если $\varphi(t) = \varphi_3(t)$, то оптимальная классификация дает размытые границы.

Если фиксирован набор эталонов классов $A = (\alpha_1, ..., \alpha_r)$, то эталонная классификация $H^A(z) = \left(h_1^A(z), ..., h_r^A(z)\right)$ для каждого из трёх вариантов функции $\varphi_j(h)$ однозначно определяется следующим образом.

1. Для $\varphi_1(h)$ $h_i^A(x) = \begin{cases} 1, & i = \arg\min\limits_{j=1,...,r}(z - \alpha_j) \\ 0, & i \neq \arg\min\limits_{j=1,...,r}(z - \alpha_j) \end{cases}$.

2. Для $\varphi_2(h)$ $h_i^A(x) = \frac{(z-\alpha_i)^{\frac{2}{1-t}}}{\left[\sum\limits_{j=1}^{r}(z-\alpha_j)^{\frac{2}{1-t}}\right]}$.

3. Случай функции $\varphi_3(h)$.

Для каждого объекта и каждого центра класса подсчитываются числа

$$v_i^{(1)} = t - (z - \alpha_i)^2 \sqrt{\frac{(r-1)t^2 + (t-1)^2}{\sum\limits_{j=1}^{r}(z - \alpha_j)^4}}.$$

Обозначим через $sign(v_i^{(1)}) = \begin{cases} 1, & v_i^{(1)} > 0 \\ 0, & v_i^{(1)} \leq 0 \end{cases}$.

Пусть r_- – число классов, для которых $sign(v_i^{(1)}) = 1$.Определим числа

$$v_i^{(1)} = sign(v_i^{(1)}) \left[t - (z - \alpha_i)^2 \sqrt{\frac{(r_- - 1)t^2 + (t-1)^2}{\sum\limits_{j=1}^{r} sign(v_i^{(1)})(z - \alpha_j)^4}} \right].$$

По ним определяем функции принадлежностей

$$h_i^A(x) = t - \sqrt{t^2 - (2t-1)v_i^{(1)}}.$$

Таким образом, для любого набора из r векторов пространства Z можно определить эталонную классификацию$H^A(z) = \left(h_1^A(z), ..., h_r^A(z)\right)$. При решении задачи аппроксимации ограничимся множеством эталонных классификаций пространства Z.

Постановка задачи аппроксимации**:** минимизировать функционал кусочной аппроксимации (1), (2) или (3) при условии, что классификация $H(x)$ является эталонной в пространстве Z. Свободными параметрами, по которым необходимо оптимизировать функционал качества аппроксимации, являются: во-первых, набор эталонов

классов $A = (\alpha_1, ..., \alpha_r)$, задающих эталонную классификацию; а во-вторых, коэффициенты линейных моделей каждого из классов $c_i, d_i, i = 1, ..., r$. Всего $(l + k + 1)\, r$ параметров. Перепишем функционалы (1), (2) и (3) в виде:

$$I'_1\left(A; c_i, d_i, i = 1, ..., r\right) = \sum_{i=1}^{r} \sum_{t=1}^{n} \left[y_t - \left((c_i, x_t) + d_i\right)\right]^2 h_i^A\left(z_t\right), \tag{5}$$

$$I'_2\left(A; c_i, d_i, i = 1, ..., r\right) = \sum_{t=1}^{n} \left[y_t - \sum_{i=1}^{r} \left((c_i, x_t) + d_i\right)\, h_i^A\left(z_t\right)\right]^2, \tag{6}$$

$$I'_3\left(A; c_i, d_i, i = 1, ..., r\right) = \sum_{i=1}^{r} \sum_{t=1}^{n} \left[y_t - \left((c_i, x_t) + d_i\right)\right]^2 \varphi\left(h_i^A\left(z_t\right)\right). \tag{7}$$

Если в эталонной классификации $\varphi(t) = \varphi_2(t)$ или $\varphi(t) = \varphi_3(t)$, то функционалы (4)–(6) дифференцируемы по своим свободным параметрам и для нахождения их локальных экстремумов можно применять градиентные процедуры.

Недостатком алгоритмов локальной оптимизации является сильная зависимость результата от начальных условий работы алгоритма. Поэтому актуальным является разработка методов глобальной оптимизации.

2. Алгоритм построения кусочной аппроксимации для конечного множества эталонов

Заметим, что если зафиксировать набор эталонов классов $A = (\alpha_1, ..., \alpha_r)$, то однозначно по методу наименьших квадратов находятся коэффициенты линейных моделей классов $c_i, d_i,\ i = 1, ..., r$, минимизирующие один из функционалов (4) – (6). Таким образом, если можно перебрать все возможные наборы эталонов классов, то можно найти глобальный минимум выбранного функционала.

Пусть в пространстве Z выделено некоторое конечное множество точек $Z_p = \{\beta_1, ..., \beta_p\}$. Будем считать, что эталоны классов можно выбирать только из множества Z_p. Число вариантов выбора различных эталонов будет равно p^r. Так как в прикладных задачах число классов в кусочной аппроксимации редко бывает больше пяти-шести, а число точек в Z_p для подавляющего числа прикладных задач порядка 100, то такой перебор вполне можно делать на современных ПЭВМ.

В качестве множества Z_p можно взять, например, реализацию в пространстве Z исходной выборки объектов. В данном случае $Z_n = \{z_1, ..., z_n\}$. В качестве множества Z_p можно взять также достаточно разреженную решетку в пространстве Z. Такой вариант хорошо использовать в качестве начальных условий для градиентного алгоритма построения кусочной аппроксимации без ограничения на набор эталонов.

Следует отдельно выделить случай, когда классификационное пространство Zсоставляет один показатель.

2.1. Кусочная аппроксимация для одномерного классифицирующего пространства. Общая постановка задачи классификации не учитывает специфику одномерной классификации, а именно – упорядоченность точек числовой прямой. В результате в оптимальной классификации все точки (за исключением эталонов классов

$\alpha_1, ..., \alpha_r$) принадлежат с некоторым весом всем классам одновременно. Более того, функция принадлежности i-го класса $h_i(x)$ не унимодальная и имеет следующую структуру: на отрезке $[\alpha_{i-1}, \alpha_i]$ $h_i(x)$ возрастает от 0 до 1, на отрезке $[\alpha_i, \alpha_{i+1}]$ она убывает от 1 до 0, на отрезке $[\alpha_{j-1}, \alpha_j]$ $(j \neq i, i+1)$ она возрастает от 0 до некоторой величины $b < 1$ и затем опять убывает до 0, при $x \to \infty$ $h_i(x) \to 1/r$.

В случае одномерной классификации естественно предполагать, что перекрываться могут лишь соседние классы.

В итоге оказывается, что, α_i – не только эталон класса, но и граница между $(i$ -1)-м и $(i+ 1)$-м классами. В соответствии с этим на эталонную классификацию $H^A(z) = \left(h_1^A(z), ..., h_r^A(z)\right)$ наложим следующие дополнительные ограничения (предполагается, что $\alpha_1 \leq \alpha_2 \leq ... \leq \alpha_r$):

- если $z \leq \alpha_1$, то $h_1^A(z) = 1$, $h_j^A(z) = 0$, $j \neq 1$, т.е. на этом луче все точки однозначно относятся к первому классу;
- если $\alpha_{i-1} \leq z \leq \alpha_i$, то $h_{i-1}^A(z) \geq 0$, $h_i^A(z) \geq 0$, $h_j^A(z) = 0$, $j \neq i-1$, $j \neq i$, т.е. на этом отрезке ненулевые веса могут иметь лишь i-й класс и $(i$-1)-й;
- если $\alpha_r \leq z$, то $h_r^A(z) = 1$, $h_j^A(z) = 0$, $j \neq r$, на этом луче все точки однозначно относятся к последнему классу.

Заметим, что функционалы (4) и (6) отличаются лишь наличием функции φ. Другими словами (4) – это (6), в котором $\varphi(h) = h$. Далее рассматривается алгоритм для (6).

Эталоны классов разбивают числовую прямую на $(r+ 1)$ промежутков, в каждом из которых могут применяться не более двух локальных линейных моделей кусочной аппроксимации. В соответствии с этим перепишем функционал (6) в следующем виде:

$$I_3''(A; c_i, d_i, i = 1, ..., r) = \sum_{z_t \leq \alpha_1} (y_t - (c_1 x_t + d_1))^2 \quad + \sum_{\alpha_r \leq z_t} (y_t - (c_r x_t + d_r))^2 \quad +$$

$$+ \sum_{i=2}^{r} \sum_{\alpha_{i-1} \leq z_t \leq \alpha_i} \left[(y_t - (c_{i-1} x_t + d_{i-1}))^2 \varphi\left(h_{i-1}^A(z_t)\right) + (y_t - (c_i x_t + d_i))^2 \varphi\left(h_i^A(z_t)\right)\right].$$

Обозначим $\Delta(i, \alpha, \beta, c, d) = \sum_{\alpha \leq z_t \leq \beta} \left[(y_t - (c x_t + d))^2 \varphi\left(h_i^A(z_t)\right)\right]$., тогда

$$I_3''(A; c_i, d_i, i = 1, ..., r) = \Delta(1, \alpha_0, \alpha_1, c_1, d_1) + \sum_{i=2}^{r} \Delta(i-1, \alpha_{i-1}, \alpha_i, c_{i-1}, d_{i-1}) +$$
$$+ \sum_{i=2}^{r} \Delta(i, \alpha_{i-1}, \alpha_i, c_i, d_i) + \Delta(r, \alpha_r, \alpha_{r+1}, c, d_1);$$

$$I_3''(A; c_i, d_i, i = 1, ..., r) = \sum_{i=1}^{r} (\Delta(i, \alpha_{i-1}, \alpha_i, c_i, d_i) + \Delta(i, \alpha_i, \alpha_{i+1}, c_i, d_i)).$$

Здесь предполагается, что $\alpha_0 = -\infty$, а $\alpha_{r+1} = +\infty$.

Если известны α_{i-1}, α_i и α_{i+1}, то однозначно известна функция $h_i(x)$, следовательно, по ней однозначно рассчитываются коэффициенты c_i и d_i.

Пусть $S_i(\alpha_{i-1}, \alpha_i, \alpha_{i+1}) = (\Delta(i, \alpha_{i-1}, \alpha_i, c_i, d_i) + \Delta(i, \alpha_i, \alpha_{i+1}, c_i, d_i))$, тогда

$I_3''(A; c_i, d_i, i = 1, ..., r) = \sum_{i=1}^{r} S_i(\alpha_{i-1}, \alpha_i, \alpha_{i+1})$. Для $i = r-1, ..., 1$ строим функции

$$F_r(\alpha_{r-1,}\alpha_r) = S_r(\alpha_{r-1}, \alpha_r, \alpha_{r+1}), \quad F_i(\alpha_{i-1,}\alpha_i) = \min_{\alpha_{i+1}} [S_i(\alpha_{i-1}, \alpha_i, \alpha_{i+1}) + F_{i+1}(\alpha_{i,}\alpha_{i+1})].$$

Последнее выражение – рекуррентное уравнение динамического программирования Беллмана. Решая задачу $F_0(\alpha_0) = \min_{\alpha_1} [S_1(\alpha_0, \alpha_1, \alpha_2) + F_1(\alpha_{0,}\alpha_1)]$, получим оптимальное значение α_1, по нему – значение α_2 и т.д. до получения оптимального значения α_r.

2.2. **Кусочная аппроксимация с классификацией по выходному параметру.** В задаче кусочной аппроксимации часто бывает полезно результирующую классификацию проектировать не на пространство X, а на y [1]. Тогда классы можно интерпретировать как режимы работы, которые обеспечивают соответствующий уровень y. Для этого достаточно рассмотреть случай $z = y$ и глобальный экстремум функционалов (1) и (3) находится с помощью процедуры динамического программирования.

Заключение

В работе рассмотрены алгоритмы решения задачи кусочно-линейной аппроксимации сложной зависимости с использованием вариационного подхода к задачам классификационного анализа данных. Даются постановки этой задачи, как для чёткой, так и для размытой классификаций. Рассматривается два способа нахождения глобального экстремума функционала – для случая конечного множества эталонов и для одномерного классификационного пространства. Для последнего случая построена рекуррентная схема Беллмана (динамическое программирование), реализация которой и обеспечивает получение глобального экстремума. Рассмотрен важный для приложений случай, когда одномерное классификационное пространство – выходной параметр.

Список литературы

1. *Райбман Н.С., Дорофеюк А.А., Касавин А.Д.* Идентификация технологических объектов методами кусочной аппроксимации. М.: ИПУ, 1977. – 70 с.
2. *Бауман Е.В.,Дорофеюк А.А.* Классификационный анализ данных. В сб.: «Избранные труды Международной конференции по проблемам управления. Том 1». М.: СИНТЕГ, 1999.
3. *Bauman E.V.* Variational Methods of Piecewise Approximation. Proc. of 8-th Conference on Analysis and Optimization Systems. (Antibes, France, 1988), INRIA, 1988.
4. *Бауман Е.В.,Дорофеюк А.А., Чернявский А.Л.* Методы структурной обработки эмпирических данных. Измерение, контроль, автоматизация. 1985, № 3. 5.
5. *Бауман Е.В., Блудян Н.О.* Методы нахождения глобальных экстремумов функционалов в задаче классификационного анализа данных. Труды ИПУ РАН, т. XIII. М.: ИПУ, 2001, С. 129–136.

Статья поступила в редакцию 28.04.2008

УДК 519.6

ОЦЕНКИ РИСКА В БАЙЕСОВСКОЙ МОДЕЛИ РАСПОЗНАВАНИЯ ПОРЯДКОВОЙ ПЕРЕМЕННОЙ ПО КОНЕЧНОМУ МНОЖЕСТВУ СОБЫТИЙ [1]

© Бериков В.Б.

Институт математики им. С.Л.Соболева СО РАН
пр-т Академика Коптюга, 4, г. Новосибирск, 630090, Россия

e-mail: *berikov@math.nsc.ru*

Abstract. In the paper we consider the ordered regression problem with use of logical decision functions. We suggest the Bayesian model of rank variable recognition on a finite set of events, which is applied for finding the optimal complexity of the class. The evaluations of the risk, obtained by the Bayesian model, are given.

Введение

Одним из наиболее перспективных подходов к решению задач распознавания и прогнозирования является подход, основанный на логико-вероятностных моделях (ЛВМ) [1]. Такого рода модели используются, например, в широко известных методах построения решающих деревьев или логических решающих функций (ЛРФ). *Разработка указанных методов особенно важна для внедрения информационно-вычислительных технологий в различных трудноформализуемых областях исследований (генетика, медицина, экономика и т.д.).* В этих областях существуют некоторые особенности, учет которых в наибольшей степени возможен только при использовании ЛВМ. Этими особенностями являются: недостаточность знаний об изучаемых объектах, что затрудняет формулировку их математических моделей; большое число разнотипных (количественных или качественных) факторов при сравнительно малом объеме данных; нелинейность взаимосвязей; наличие пропусков, погрешностей измерения характеристик; требование представления результатов анализа в форме, понятной специалистам прикладной области.

Проблема построения моделей, обладающих минимальным риском ошибочного прогноза, является одной из важнейших при решении задач как методами, основанными на ЛВМ, так и другими методами анализа данных. Известно, что сложность модели (где под сложностью может пониматься размерность Вапника-Червоненкиса, число логических закономерностей или листьев решающего дерева и т.д.) является существенным фактором, влияющим на качество решений. Для наилучшего качества должен достигаться определенный компромисс между сложностью и точностью решений на обучающей выборке. В достаточно большом круге прикладных задач, наряду с обучающей выборкой, могут быть использованы различного рода экспертные знания, не связанные с жестким заданием модели распределения. При выборе оптимальной сложности ЛВМ возникает проблема совместного учета имеющихся эмпирических данных и экспертных знаний. Провести такой учет позволяет, в частности,

[1]При поддержке международного фонда "Научный потенциал" (грант №144), Российского фонда фундаментальных исследований (гранты №07-01-00331-а, 08-07-00136а)

байесовская теория обучения. *В рамках этого направления в работах* [1, 2, 3] *были разработаны байесовские модели распознавания по конечному множеству событий, созданы алгоритмы построения этих моделей.* Предложенный подход обладает тем преимуществом, что он не ориентирован на самый "неблагоприятный" вид распределения и на асимптотический случай.

В работе рассматривается распространение разработанного подхода на задачу распознавания порядковой переменной. Указанная задача (известная также как порядковая регрессия (ordered regression)) в некотором смысле является "промежуточной" между распознаванием образов и регрессионным анализом. Обзор имеющихся методов решения можно найти, например, в работе [4].

Требуется с использованием байесовской модели оценить оптимальную сложность класса ЛРФ, при которой ожидаемый риск минимален. Для этого необходимо в рамках модели найти риск, усредненный по множеству стратегий природы и по всевозможным обучающим выборкам заданного объема. Кроме того, расматривается задача нахождения апостериорной оценки риска и ее применения как критерия качества решающей функции.

1. Байесовская модель распознавания по конечному множеству событий

При распознавании образов требуется предсказать номер образа ω для произвольного объекта a генеральной совокупности Г, описываемого набором некоторых переменных $X_1, \ldots, X_n$. При этом предсказание осуществляется на основе анализа случайной обучающей выборки, в которой для каждого объекта указаны значения этих переменных вместе с номером соответствующего образа. Переменные могут быть разнотипными, т.е. часть из них может иметь количественную, а часть – качественную природу. Как правило, для решения задачи используется некоторый класс решающих функций Ф, в котором ищется оптимальная по заданному критерию функция. Класс логических решающих функций определяется на множестве разбиений пространства переменных на конечное число подобластей, описываемых конъюнкциями предикатов простого вида. Число подобластей M определяет степень сложности логической функции.

Байесовская модель распознавания образов по конечному множеству событий вводится путем формулирования некоторых положений, смысл которых состоит в абстрагировании от локальных метрических свойств пространства переменных - перехода от точек пространства к "событиям", где под событием понимается принятие исходными переменными значений из некоторой подобласти разбиения; рассмотрении задачи распознавания по значениям дискретной неупорядоченной переменной; использовании понятия метода обучения как отображения из множества всевозможных выборок во множество решающих функций; привлечении различных способов формализации экспертных знаний о задаче распознавания, не требующих жесткого задания модели распределения вероятностей.

Рассмотрим две дискретные случайные переменные: входную (объясняющую или предикторную) переменную X с множеством неупорядоченных значений $D_X = \{c_1, \ldots, c_j, \ldots, c_M\}$, где c_j – j-е значение (ячейка) и выходную (предсказываемую) переменную Y с множеством неупорядоченных значений $D_Y = \{\omega^{(1)}, \ldots, \omega^{(i)}, \ldots, \omega^{(K)}\}$, где $\omega^{(i)}$ – i-е значение, называемое i-м образом (классом), $K \geq 2$ – число классов. Предполагается, что значения переменной Y упорядочены. Закодируем значения переменной X через номера ячеек, а образы – через соответствующие им номера. Для решения задачи распознавания требуется найти решающую функцию $f : D_X \to D_Y$. Предполагается, что задана ограниченная неотрицательная функция потерь $L_{i,q}$, возникающих в случае принятия решения $Y = i$, когда истинный образ есть q. Будем рассматривать квадратичную функцию потерь такого вида: $L_{i,q} = (i - q)^2$, где $i, q = 1, 2, \ldots, K$.

Пусть на подмножествах множества $D_X \times D_Y$ определена вероятностная мера; $p_j^{(i)}$ - вероятность совместного события "$X = j, Y = i$", при этом выполняется $p_j^{(i)} \geq 0$, $\sum\limits_{i,j} p_j^{(i)} = 1$, $j = 1, 2, \ldots, M$, $i = 1, 2, \ldots, K$. Обозначим $\theta = (\theta_1, \ldots, \theta_j, \ldots, \theta_M)$, где $\theta_j = (p_j^{(1)}, p_j^{(2)}, \ldots, p_j^{(K)})$.

Пусть имеется некоторый класс Ф решающих функций распознавания. Под сложностью класса будем понимать величину M. Каждой решающей функции f из Ф можно сопоставить ожидаемые потери (риск) при распознавании произвольного наблюдения

$$R_f(\theta) = \sum_{j=1}^{M} \sum_{i=1}^{K} p_j^{(i)} (f(j) - i)^2.$$

Если бы вектор θ был известен, можно было бы построить оптимальную решающую функцию ("функцию регрессии") f_r, для которой риск минимален:

$$f_r(j) = l : \sum_{i=1}^{K} p_j^{(i)} (l - i)^2 = \min_{\rho} \sum_{i=1}^{K} p_j^{(i)} (\rho - i)^2,$$

где $\rho = 1, 2, \ldots, K$, $j = 1, 2, \ldots, M$. Легко показать, что значение $f_r(j)$ находится в ближайшей окрестности точки $f^* = \dfrac{1}{p_j} \sum\limits_i p_j^{(i)} i$, где $p_j = \sum\limits_i p_j^{(i)}$.

В прикладных задачах вектор θ обычно неизвестен. Решающая функция выбирается из Ф на основе случайной выборки наблюдений над X и Y (обучающей выборки) с помощью некоторого *заданного метода* μ. Будем полагать, что результаты наблюдений являются независимыми, одинаково распределенными случайными величинами. Пусть N– объем выборки, $n_j^{(i)}$- число наблюдений i-го образа, соответствующих j-й ячейке; $\sum\limits_{i,j} n_j^{(i)} = N$. Обозначим вектор частот через $s = (s_1, \ldots, s_j, \ldots, s_M)$, где $s_j = (n_j^{(1)}, n_j^{(2)}, \ldots, n_j^{(K)})$. Оценкой эмпирического риска (или просто эмпирическим

риском) для решающей функции f назовем величину

$$\widehat{R}_f(s) = \frac{1}{N}\sum_{i,j} n_j^{(i)}(f(j)-i)^2.$$

Метод μ можно рассматривать как функцию, задающую отображение конечного множества всевозможных векторов $\mathbf{S}=\{s\}$ во множество решающих функций Φ: $f=\mu(s)$. Метод μ^* минимизации эмпирического риска (аналог метода наменьших квадратов) состоит в следующем: поставить в соответствие j-й ячейке такой номер образа l, что

$$\sum_i n_j^{(i)}(l-i)^2 = \min_\rho \sum_i n_j^{(i)}(\rho-i)^2,$$

где $\rho = 1,2,\ldots,K$, $j=1,2,\ldots,M$. Обозначим через $\widehat{f_r}$ решающую функцию ("эмпирическую регрессию"), полученную с помощью этого метода. Нетрудно убедиться, что $\widehat{f_r}(j)$ равно округленному до ближайшего целого среднему значению переменной Y по j-й ячейке: $\widehat{f}^* = \dfrac{1}{n_j}\sum\limits_i n_j^{(i)} i$, где $n_j = \sum\limits_i n_j^{(i)}$. Условимся, для определенности, что в случае, когда округление не может быть проведено однозначно, выбирать наибольший номер класса. Если некоторая ячейка оказалась "пустой", т.е. не содержащей ни одного наблюдения, договоримся приписывать такой ячейке значение Y, равное K.

Пусть S – случайный вектор частот. Этот вектор подчиняется полиномиальному распределению $\mathbf{P}(S=s|\theta)=p(s|\theta)$, где

$$p(s|\theta) = \frac{N!}{\prod\limits_{i,j} n_j^{(i)}!}\prod_{i,j}\left(p_j^{(i)}\right)^{n_j^{(i)}}.$$

Рассмотрим семейство полиномиальных моделей распределения вектора частот с множеством параметров $\Lambda=\{\theta\}$. Будем использовать байесовский подход: предположим, что на множестве Λ определена случайная величина $\Theta = (P_1^{(1)},\ldots,P_M^{(K)})$ с некоторым известным априорным распределением $p(\theta)$ при $\theta\in\Lambda$. В этом случае риск является функцией $R_f(\Theta)$, зависящей от случайного вектора параметров модели. Значение этой функции можно рассматривать как оценку неизвестного риска, вычисленную в предположении о том, что истинный вектор параметров равен θ.

Будем полагать, что величина Θ подчиняется распределению Дирихле $\Theta \sim Dir(d_1^{(1)}, d_1^{(2)},\ldots,d_M^{(K)})$: $p(\theta) = \dfrac{1}{Z}\prod\limits_{i,j}(p_j^{(i)})^{d_j^{(i)}-1}$, где $d_j^{(i)} > 0$ – некоторые заданные вещественные числа, выражающие априорные знания о распределении Θ, $i=1,2,\ldots,K$, $j=1,2,\ldots M$, Z – нормализующая константа: $Z = \dfrac{\prod\limits_{i,j}\Gamma(d_j^{(i)})}{\Gamma(D)}$, где $\Gamma(\cdot)$ – гамма-функция, $D=\sum\limits_{i,j} d_j^{(i)}$. При $d_j^{(i)}\equiv 1$ получим случай равномерного априорного распределения, использование которого оправдано при отсутствии априорной информации о классе распределений Λ.

Выбор априорного распределения Дирихле объясняется тем, что оно удобно для выражения априорных знаний о распределении на множестве стратегий природы: величины $d_j^{(i)}$ аналогичны числу попаданий в ячейки наблюдений различных образов. Это распределение сопряжено с рассматриваемым полиномиальным распределением вектора частот. В работе [2] было показано, что при уменьшении параметра Дирихле понижается "степень пересечения" между образами, понимаемая как ожидаемое значение вероятности ошибки оптимальной байесовской решающей функции; было найдено выражение для указанной зависимости, согласно которому можно определять значение параметра по экспертной оценке степени пересечения.

В дальнейшем, при формулировке теоретических утверждений будем считать, что *выполняются все вышеозначенные предположения.*

2. Ожидаемый риск для метода обучения

Найдем математическое ожидание функции риска, где усреднение проводится по всевозможным значениям параметров из множества Λ и всевозможным обучающим выборкам заданного объема N. Заметим, что в методе обучения при принятии решения может учитываться не только эмпирическая ошибка. Например, если есть экспертные знания о предпочтении какого-либо образа, решение может приниматься в его пользу несмотря на увеличение ошибки.

Теорема 1. *Пусть $f = \mu(s)$ – решающая функция, полученная по выборке s с помощью некоторого детерминированного метода μ, такого, что решение для j-й ячейки принимается по набору частот s_j, $j = 1, \ldots, M$. Тогда математическое ожидание $\mathbf{E}R_{\mu(S)}(\Theta)$ функции риска равно*

$$R_\mu = \frac{\Gamma(D)N!}{\Gamma(D+N+1)} \sum_{j=1}^{M} \frac{1}{\Gamma(D-d_j)\prod\limits_{l=1}^{K}\Gamma(d_j^{(l)})} \times$$

$$\times \sum_{s_j} \frac{\Gamma(\overline{n}_j + D - d_j)}{\prod\limits_l n_j^{(l)}!\overline{n}_j!} \prod_l \Gamma(d_j^{(l)} + n_j^{(l)}) \sum_{q=1}^{K} (f(j)-q)^2 (d_j^{(q)} + n_j^{(q)}),$$

где $d_j = \sum\limits_l d_j^{(l)}$, $\overline{n}_j = N - \sum\limits_l n_j^{(l)}$, оператор $\sum\limits_{s_j}$ означает, что суммирование ведется по всем наборам $(n_j^{(1)}, n_j^{(2)}, \ldots, n_j^{(K)})$ таким, что $\sum\limits_l n_j^{(l)} \le N$.

Доказательство. Из определения функции риска следует, что

$$\mathbf{E}R_{\mu(S)}(\Theta) = \int_\Lambda \sum_s p(s|\theta)p(\theta) \sum_{j,q} (f(j)-q)^2 p_j^{(q)} d\theta =$$

$$= \frac{1}{Z} \sum_{j,q} \int_\Lambda \prod_{l,m} (p_m^{(l)})^{d_m^{(l)}-1} p_j^{(q)} \sum_{s_j} (f(j)-q)^2 \sum_{\substack{s: \\ s_j \text{ фиксир.}}} p(s|\theta) d\theta =$$

$$= \frac{1}{Z}\sum_{j,q}\int\limits_{\Lambda}\prod_{l,m}(p_m^{(l)})^{d_m^{(l)}-1}p_j^{(q)}\sum_{s_j}(f(j)-q)^2 p(s_j|\theta_j)d\theta$$

по свойству полиномиального распределения. Здесь $p(s_j|\theta_j)$ – функция вероятности распределения величины S_j: $p(s_j|\theta_j) = \dfrac{N!}{\prod\limits_l n_j^{(l)}!\overline{n}_j!}\prod\limits_l (p_j^{(l)})^{n_j^{(l)}}(\overline{p}_j)^{\overline{n}_j}$, $\overline{p}_j = 1 - \sum\limits_l p_j^{(l)}$. Далее, $\mathbf{E}R_{\mu(S)}(\Theta) =$

$$= \frac{1}{Z}\sum_{j,q}\sum_{s_j}\frac{N!}{\prod\limits_l n_j^{(l)}!\overline{n}_j!}\int\limits_{\substack{p_j^{(1)},\dots p_j^{(K)}:\\ \sum\limits_i p_j^{(i)}\le 1}}\prod_l (p_j^{(l)})^{d_j^{(l)}-1}p_j^{(q)}(f(j)-q)^2\prod_l (p_j^{(l)})^{n_j^{(l)}}(\overline{p}_j)^{\overline{n}_j}\times$$

$$\times\left\{\int\limits_{\substack{\{p_m^{(l)}\}:\\ \sum\limits_{\substack{l,m:\\ m\ne j}} p_m^{(l)} = 1-\overline{p}_j}}\prod_{\substack{l,m:\\ m\ne j}}(p_m^{(l)})^{d_m^{(l)}-1}d\theta_1\dots d\theta_{j-1}d\theta_{j+1}\dots d\theta_M\right\}d\theta_j.$$

Воспользуемся формулой, обобщающей формулу Дирихле [5]:

$$\int\limits_{\substack{\{x_1,\dots,x_{m-1}:\\ x_i\ge 0\\ \sum\limits_{i=1}^{m-1}x_i\le h\}}}\prod_{i=1}^{m-1}x_i^{d_i-1}(h-\sum_i x_i)^{d_m-1}dx_1\dots dx_{m-1} = \frac{\prod\limits_{i=1}^{m}\Gamma(d_i)}{\Gamma(\sum\limits_{i=1}^{m}d_i)}h^{\sum\limits_{i=1}^{m}d_i-1},$$

где $d_1,\dots,d_m$ – вещественные неотрицательные числа. Отсюда

$$\mathbf{E}R_{\mu(S)}(\Theta) = \frac{1}{Z}\sum_{j,q}\sum_{s_j}\frac{N!}{\prod\limits_l n_j^{(l)}!\overline{n}_j!}\int\limits_{\substack{p_j^{(1)},\dots,p_j^{(K)}:\\ \sum\limits_i p_j^{(i)}\le 1}}\prod_l (p_j^{(l)})^{n_j^{(l)}+d_j^{(l)}-1}p_j^{(q)}(f(j)-q)^2\times$$

$$\times\frac{\prod\limits_{\substack{l,m:\\ m\ne j}}\Gamma(d_m^{(l)})}{\Gamma(D-d_j)}(\overline{p}_j)^{\overline{n}_j}(\overline{p}_j)^{D-d_j-1}d\theta_j = \frac{1}{Z}\sum_{j,q}\sum_{s_j}\frac{N!}{\prod\limits_l n_j^{(l)}!\overline{n}_j!}(f(j)-q)^2\frac{\prod\limits_{\substack{l,m:\\ m\ne j}}\Gamma(d_m^{(l)})}{\Gamma(D-d_j)}\times$$

$$\times\frac{\prod\limits_{l:\,l\ne q}\Gamma(d_j^{(l)}+n_j^{(l)})\Gamma(d_j^{(q)}+n_j^{(q)}+1)\Gamma(\overline{n}_j+D-d_j)}{\Gamma(N+D+1)}$$

Воспользуемся свойством гамма-функции: $\Gamma(x+1) = \Gamma(x)x$, получим

$$\mathbf{E}R_{\mu(S)}(\Theta) = \frac{\Gamma(D)}{\prod\limits_{l,m}\Gamma(d_m^{(l)})}\sum_j\sum_{s_j}\frac{N!}{\prod\limits_l n_j^{(l)}!\overline{n}_j!}\frac{\prod\limits_{\substack{l,m:\\ m\neq j}}\Gamma(d_m^{(l)})}{\Gamma(N+D+1)\Gamma(D-d_j)}\times$$
$$\times\prod_l\Gamma(d_j^{(l)}+n_j^{(l)})\Gamma(\overline{n}_j+D-d_j)\sum_q(d_j^{(q)}+n_j^{(q)})(f(j)-q)^2.$$

Отсюда следует справедливость теоремы 1.

3. Нахождение оптимальной сложности класса логических решающих функций

Покажем, как в рамках байесовской модели можно находить оптимальную сложность класса ЛРФ. Образуем некоторое исходное разбиение пространства переменных $X_1,\ldots,X_n$ на подобласти $E_1,E_2,\ldots,E_{M_0}$. Зададим соответствующую байесовскую модель распознавания по множеству из M_0 событий. Сформируем также множество разбиений, получаемых путем объединения некоторых подобластей исходного разбиения. Пусть произошло объединение подобластей $E_1,\ldots,E_{j_1}$ в новую подобласть $\widetilde{E}_1$, подобластей $E_{j_1+1},\ldots,E_{j_1+j_2}$ в подобласть $\widetilde{E}_2$, ..., подобластей $E_{j_1+\cdots+j_{M-1}+1},\ldots,E_{j_1+\cdots+j_M}$ в подобласть $\widetilde{E}_M$. Обозначим $\widetilde{p}_t^{(i)} = \mathbf{P}(Y=i,\widetilde{E}_t)$, где $t=1,2,\ldots,M$, $i=1,2,\ldots,K$. Тогда $\widetilde{p}_1^{(i)} = p_1^{(i)}+\cdots+p_{j_1}^{(i)}$, $\widetilde{p}_2^{(i)} = p_{j_1+1}^{(i)}+\cdots+p_{j_1+j_2}^{(i)}$, ..., $\widetilde{p}_M^{(i)} = p_{j_1+\cdots+j_{M-1}+1}^{(i)}+\cdots+p_{j_1+\cdots+j_M}^{(i)}$. По свойству распределения Дирихле [6], вектор $\widetilde{\Theta} = (\widetilde{P}_1^{(1)},\ldots,\widetilde{P}_t^{(i)},\ldots,\widetilde{P}_M^{(K)}) \sim Dir(d_1^{(1)}+\cdots+d_{j_1}^{(1)},\ldots,d_{j_1+\cdots+j_{t-1}+1}^{(i)}+\cdots+d_{j_1+\cdots+j_t}^{(i)},\ldots,d_{j_1+\cdots+j_{M-1}+1}^{(K)}+\cdots+d_{j_1+\cdots+j_M}^{(K)})$.

Теорема 1 позволяет вычислить ожидаемый риск для каждого из вариантов разбиения (число разбиений определяет сложность класса ЛРФ и, одновременно, сложность соответствующей ЛВМ). Будем перебирать различные варианты и найдем такой из них, для которого ожидаемый риск минимален.

Рассмотрим следующий пример. Пусть все подобласти сгруппированы в пары; каждый последующий вариант разбиения образуется из предыдущего путем слияния пар. Сложность исходной модели $M_0 = 2^u$, где u - целое число. Все параметры Дирихле исходной модели совпадают (что говорит об отсутствии априорных предпочтений между подобластями) и равны некоторой величине d_0. Таким образом, имеем последовательность разбиений сложности $M = M_0,\ldots,4,2,1$, а также набор байесовских моделей с параметрами $d = d_0, 2d_0, 4d_0, \ldots, M_0 d_0$. Пусть метод μ^* минимизирует эмпирическую ошибку. На рис. 1 приведен пример полученного графика зависимости ожидаемого риска от сложности M=1; 2; 4; 8 для объема обучающей выборки N=11. Параметр d_0=0.1, $K=6$.

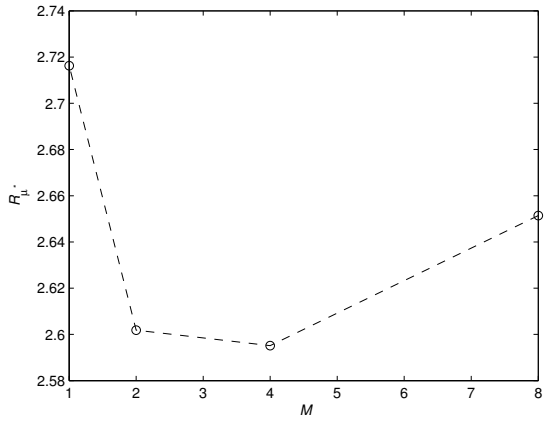


Рис. 1. Зависимость ожидаемого риска R_{μ^*} от сложности класса M.

4. Апостериорный риск

В данном параграфе исследуется оценка функции риска для решающей функции, полученной по заданной выборке. Согласно байесовскому подходу, функция риска $R_f(\Theta)$ при фиксированной выборке является случайной величиной, зависящей от случайного вектора Θ. Рассмотрим апостериорную плотность распределения $p(\theta|S = s) = p(\theta|s)$, где $s \in \mathbf{S}$. По свойству распределения Дирихле [6], $\Theta|s \sim Dir(d_1^{(1)} + n_1^{(1)}, \ldots, d_j^{(i)} + n_j^{(i)}, \ldots, d_M^{(K)} + n_M^{(K)})$.

Теорема 2. *Пусть f – произвольная решающая функция, выбранная из множества решающих функций Φ, и задана выборка $s \in \boldsymbol{S}$. Тогда апостериорное математическое ожидание функции риска у решающей функции f равно*

$$\mathbf{E}(R_f(\Theta)|s) = \frac{1}{N+D}\sum_{j,q}(f(j)-q)^2(n_j^{(q)} + d_j^{(q)}).$$

Доказательство. По определению функции риска,

$$\mathbf{E}(R_f(\Theta)|s) = \mathbf{E}_{\Theta|s}\sum_{j,q}(f(j)-q)^2 P_j^{(q)} = \sum_{j,q}(f(j)-q)^2(\mathbf{E}_{\Theta|s}P_j^{(q)}) =$$

$$= \sum_{j,q}(f(j)-q)^2 \int\limits_{\Lambda} p(\theta|s)p_j^{(q)}d\theta = \sum_{j,q}\frac{\Gamma(N+D)}{\prod\limits_{m,l}\Gamma(d_m^{(l)}+n_m^{(l)})}(f(j)-q)^2 \times$$

$$\times \int\limits_{\Lambda}\prod_{m,l}(p_m^{(l)})^{d_m^{(l)}+n_m^{(l)}-1}p_j^{(q)}d\theta =$$

$$= \sum_{j,q}\frac{\Gamma(N+D)}{\prod\limits_{m,l}\Gamma(d_m^{(l)}+n_m^{(l)})}(f(j)-q)^2\frac{\prod\limits_{\substack{m,l:\\(m,l)\neq(j,q)}}\Gamma(d_m^{(l)}+n_m^{(l)})\Gamma(d_j^{(q)}+n_j^{(q)}+1)}{\Gamma(N+D+1)} =$$

$$= \frac{1}{N+D}\sum_{j,q}(f(j)-q)^2(n_j^{(q)}+d_j^{(q)}),$$

что и требовалось доказать.

Назовем найденную выше величину $\mathbf{E}(R_f(\Theta)|s)$ байесовской оценкой риска, соответствующей решающей функции f. Полученная оценка может использоваться на этапе обучения как критерий оптимальности решающей функции.

Для поиска оптимальной ЛРФ может применяться следующий способ. Выборка разбивается на две части; по первой части строится дерево регрессии, возможно переобученное, т.е. имеющее излишнюю сложность в результате чрезмерной "настройки" на объекты обучающей выборки. По полученному дереву формируется множество событий, связанных с разбиениями пространства переменных в соответствии со структурой дерева. Затем вторая часть выборки используется для оценки качества распознавания с применением байесовской модели распознавания по конечному множеству событий. При этом рассматриваются различные варианты усечения дерева; ищется такой вариант, для которого ожидаемый апостериорный риск минимален.

Заключение

В работе впервые предлагается использовать байесовскую модель распознавания по конечному множеству событий в задаче порядковой регрессии. В рамках байесовской модели найдено выражение для риска, усредненного по множеству стратегий природы и по всевозможным обучающим выборкам заданного объема. Предложен способ определения оптимальной сложности класса логических решающих функций, при которой ожидаемый риск минимален. Найдена апостериорная оценка риска, предложен способ ее применения при построении логической решающей функции.

В перспективе предполагается *исследовать и другие виды априорных распределений в байесовской модели, а также применить полученные результаты для решения таких задач анализа данных, как группировка объектов и адаптивное построение логических решающих функций.*

Список литературы

1. *Лбов Г.С., Бериков В.Б.* Устойчивость решающих функций в задачах распознавания образов и анализа разнотипной информации. Новосибирск: Изд-во Ин-та математики, 2005.
2. *Berikov V.B.* Bayes estimates for recognition quality on a finite set of events // Pattern Recognition and Image Analysis. 2006. V. 16, N 3. P. 329–343.
3. *Бериков В.Б., Лбов Г.С.* Выбор оптимальной сложности класса логических решающих функций в задачах распознавания образов // Доклады Академии наук. 2007. Том 417, №1. С. 26-29.
4. *Kramer S., Widmer G., Pfahringer B., DeGroeve M.* Prediction of ordinal classes using regression trees // Fundamenta Informaticae. 2000. V. 34. P.1-15.
5. *Фихтенгольц Г.М.* Курс дифференциального и интегрального исчисления. Т. 3. М.: Физматлит, 1960.
6. *Уилкс С.* Математическая статистика. М.: Наука, 1967.

Статья поступила в редакцию 30.04.2008

УДК 004.9

ТЕКСТУРНЫЙ АНАЛИЗ УЛЬТРАЗВУКОВЫХ ИЗОБРАЖЕНИЙ ЩИТОВИДНОЙ ЖЕЛЕЗЫ

© Богуш А.Л., Ковалев В.А.

Объединённый институт проблем информатики НАН Беларуси
ул. Сурганова, 6, 220012, г. Минск, Беларусь

e-mail: *bogush@newman.bas-net.by, vassili.kovalev@gmail.com*

Abstract. In this paper we study effectiveness of different methods for the analysis of ultrasound image texture of the thyroid. It was found that the anisotropy properties of thyroid gland images contain the information most useful for the classification purposes. The best methods resulted in the accuracy of about 80% of pairwise classification of three different types of tumors.

Введение

Ультразвуковое обследование широко применяется в области медицинской диагностики. Это связано с несколькими особенностями визуальной диагностики данного типа. Во-первых, не существует достоверных данных о вредном воздействии ультразвукового излучения на пациента. Единственное обнаруженное изменение – это нагрев тканей на доли градуса, однако не получено доказательств негативного влияния такого нагрева на здоровье пациента. Во-вторых, установка для ультразвукового обследования очень компактна и может занимать место вплоть до размера дорожной сумки, в то время как, например, ЯМР-томограф занимает несколько комнат. Такая компактность позволяет использовать ультразвуковые аппараты в клиниках, в машинах скорой помощи, а также при выезде врача к пациенту на дом. В-третьих, стоимость такого аппарата и стоимость одного обследования существенно ниже, чем на других приборах медицинской диагностики (компьютерном томографе, томографе ядерного магнитного резонанса). В совокупности это позволяет использовать данный тип диагностики при широком спектре заболеваний. Кроме того, для отслеживания динамики исследуемого заболевания обследования можно повторять через короткие промежутки времени.

1. Цели

Проблемы эффективной ранней диагностики рака чрезвычайно важна для регионов, пострадавших от аварии на Чернобыльской АЭС. Ультразвуковое сканирование [1] – это нонинвазивная технология, подходящая как для диагонстики, так и для локализации узлов и навигации. Однако, кроме явных преимуществ, ультразвук обладает и своими недостатками, что затрудняет его повсеместное использование. Основное из них – качество получаемых изображений является низким, что связано с физическими особенностями прохождения ультразвуковых волн в тканях. Это накладывает дополнительные требования на квалификацию врача, так как иногда бывает очень сложно отличить разные типы тканей на таких изображениях. В итоге, диагностические способности ультразвука оказываются достаточно слабы для постановки точного диагноза без инвазивного исследования ткани при помощи биопсии

(FNA – fine needle aspiration). Цель этой работы заключается в оценке полезности текстурных свойств ультразвуковых изображений для компьютерного диагностирования опухолей щитовидной железы.

2. Методы

В период с 1998 по 2004 годы, 50 пациентов (18 мужчин и 32 женщин) в возрасте от 10 до 68 лет с узловой болезнью щитовидной железы были обследованы в клинике Научно-исследовательского Института Радиационной медицины и Эндокринологии и Республиканском центре Медицинской реабилитации и бальнеолечения в г. Минске. Заболевания были верифицированы либо как доброкачественный узловой (многоузловой) зоб, либо как аденома, либо карцинома (рак). Во время первоначальных исследований были проведены обычные двумерные (2D) и трехмерные (3D) ультразвуковые сканирования щитовидной железы при помощи метода свободной руки, а также оценка гормонального статуса щитовидной железы (анализ TSH, T3, FT3, T4, FT4, TG, TG-AB, TPO-AB) и пункционная биопсия (FNA). В соответствии с медицинскими показаниями (размер узлов щитовидной железы более $3cm$ в диаметре и продолжает увеличиваться, злокачественность или подозрение на злокачественность цитологических результатов после FNA), проводилось хирургическое вмешательство на щитовидной железе в Минском Государственном Онкологическом Диспансере на всех пациентах. Диагноз также был подтверждён морфологическим обследованием резецированного поражения после операции. Запись трехмерных ультразвуковых изображений [2] новообразования щитовидной железы проводилась системой "Freescan" (EchoTech, Germany) [3] и ультразвуковым сканером Hewlett Packard Image Point. Одно трехмерное обследование содержало набор из $300 - 400$ двумерных ультразвуковых изображений с соответствующими им данными о пространственном положении. Тот же ультразвуковой сканер использовался для получения обычных двумерных изображений. Узловые образования были разделены на классы в соответствии с характеристиками тока крови на ультразвуковых изображениях, полученных с эффектом Доплера. Так как у некоторых пациентов присутствовали опухоли в обеих долях щитовидной железы, в итоге в данной работе было проанализировано 64 изображения, представляющих 19 случаев зоба, 12 – аденомы, и 33 – раковых опухолей. Сегментация щитовидной железы проводилась при помощи полуавтоматического метода, основанного на ручных обводках области интереса (Рис. 1), проведённых врачом–экспертом в $6 - 10$ сечениях. Затем был использован метод реконструкции поверхности, основанный на математической морфологии [4] для интерполирования поверхности области интереса (Рис. 6). После реконструкции, поверхность использовалась для вырезания области интереса во всех слоях. Для этого трехмерная поверхность и сегментируемое изображение располагались в единой глобальной системе координат и затем вычислялось их пересечение. В результате процедуры сегментации для каждого узла было получено в среднем 50 сечений, а для щитовидной железы – $200 - 300$. Средний размер изображения щитовидной железы составил $2,5 \times 10^7$ вокселей, а изображения узла – $6,5 \times 10^6$ вокселей. Эти изображения были использованы для получения изображений различных видов

тканей и их границ, а также их сочетаний. Наиболее значимые результаты были получены при использовании изображений узлов с окружающей тканью в 3 пикселя от обводки эксперта (см. рис. 3).

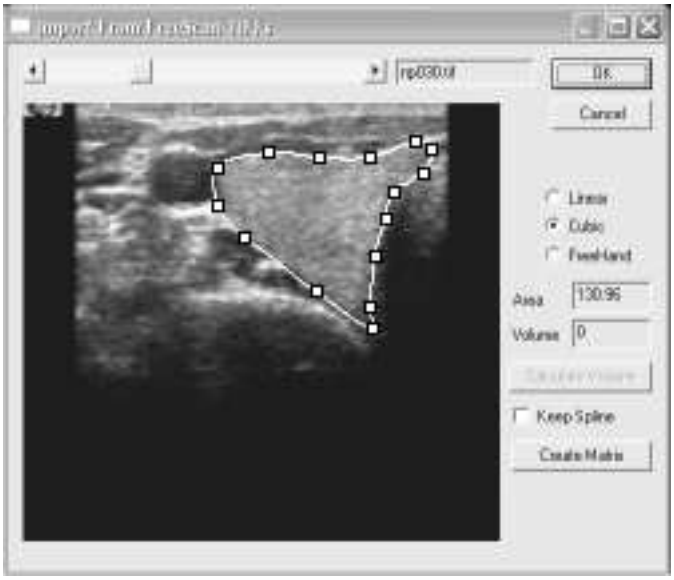


Рис. 1. Обводка щитовидной железы

Рис. 2. Реконструкция объёма

Следуя концепции обобщенных матриц совместной встречаемости [5], вычислялись признаки для набора двумерных изображений, которые составляют область интереса. Для формального определения матриц совместной встречаемости будем

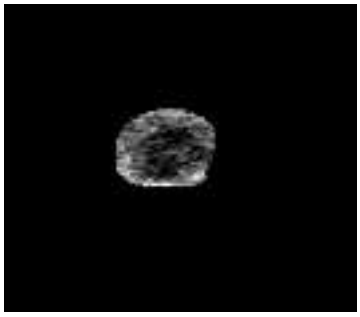


Рис. 3. Сегментированный узел

считать, что произвольная пара пикселей (i,k) , определена на дискретной пиксельной решетке индексами пикселей $i=(x_i,y_i)$, $k=(x_k,y_k)$ и евклидовым расстоянием $d(i,k)$. Обозначим яркость в этих пикселях как $I(i)$ и $I(k)$, локальные значения градиента как $G(i)$, $G(k)$ и угол между векторами градиента как $a(i,k)$. Тогда шестимерная (6-D) матрица совместной встречаемости общего вида определяется как

$$W=||w(I(i),I(k),G(i),G(k),a(i,k),d(i,k))|| \tag{1}$$

Для вычисления значений абсолютной величины градиента $G=\sqrt{Gx^2+Gy^2}$ и углов между векторами градиента мы использовали оператор Собеля с точными значениями $\pm 1/\sqrt{2}$ в масках свёртки (Рис. 4).

$\frac{-1}{\sqrt{2}}$	0	$\frac{1}{\sqrt{2}}$
-1	0	$+1$
$\frac{-1}{\sqrt{2}}$	0	$\frac{1}{\sqrt{2}}$

Gx

$\frac{1}{\sqrt{2}}$	0	$\frac{1}{\sqrt{2}}$
-1	0	$+1$
$\frac{-1}{\sqrt{2}}$	0	$\frac{-1}{\sqrt{2}}$

Gy

Рис. 4. Маска свёртки для оператора Собеля

Обозначая интервалы (кванты) значений яркости $I(i)$, $I(k)$ их индексами $b_I=1,\ldots,B_I$, интервалы величины градиента $G(i)$, $G(k)$ индексами $b_G=1,\ldots,B_G$, интервалы относительного угла градиента $a(i,k)$ индексами $b_a=1,\ldots,B_a$, а целые (округленные) расстояния на дискретной решетке изображения $d(i,k)$ индексами $b_d=1,\ldots,D_d$, элемент матрицы $w(I(i),I(k),G(i),G(k),a(i,k),d(i,k))$ формально определяется как

$$
\begin{aligned}
w(b_{Ii}, b_{Ik}, b_{Gi}, b_{Gk}, b_a, b_d) & \\
= card\{ \quad & (i,k) \in R^2 | i \neq k, \\
& b_{Ii} = I(i), b_{Ik} = I(k), b_{Gi} = G(i), b_{Gk} = G(k), \\
& b_a = a(i,k), b_d = round(d(i,k)), \\
& x_k = (x_i + \Delta x), y_k = (y_i + \Delta y), \\
& -D \leq \Delta x \leq D, \\
& 0 \leq \Delta y \leq D, \text{ если } \Delta x < 0, \\
& 1 \leq \Delta y \leq D, \text{ иначе.}
\end{aligned}
$$

В этом выражении Δx и Δy определяют смещение по осям X и Y, выраженные в единицах растра изображения. Последние три строчки определения формализуют требование выбора всех возможных пар пикселей без повтора. При вычислении матриц мы всегда следуем исходной растеризации изображения и округлённым до целых величин (индексам матрицы) евклидовым расстояниям $d(i,k)$ чтобы избежать рассмотрения несуществующих значений яркости, вызванных интерполяцией. Оператор округления был определён здесь в общем смысле, т.е. как округление к ближайшему целому. Согласно типам осей, такой вид матрицы совместной встречаемости для краткости мы назовём IIGGAD. Нам также необходимо делать проверку на существование каждого пикселя, так как мы удаляли некоторые из них во время сегментации, следовательно, мы не включаем пары пикселей в рассмотрение, если хотя бы один из них не принадлежит к отсегментированной части изображения. Матрицу для набора двумерных изображений (слоев) мы вычисляли как сумму матриц для каждого 2D изображения в последовательности предполагая, что текстура на всех изображениях имеет характерные признаки для каждого конкретного вида ткани. Такое суммирование приводит к приблизительно 7×10^5 парам пикселей для всей щитовидной железы и 2×10^5 парам для узлов, при $D = 4$. Мы также исследовали некоторые редуцированные версии матриц совместной встречаемости: интенсивности (IID), значения градиента (GGD) и угла градиента (gAD) , а также их комбинации. Редуцированные матрицы могут быть получены из основной матрицы IIGGAD путём суммирования вдоль соответствующих осей. Исходная матрица IIGGAD не может быть восстановлена из этих матриц обратно. Следовательно, любая комбинация частных матриц совместной встречаемости не может описать изображение также подробно, как матрица IIGGAD.

Анализировать всю матрицу совместной встречаемости не очень удобно из-за её большого размера. Например, наиболее репрезентативная матрица $I_8 I_8 G_8 G_8 A_6 D_4$ будет состоять из порядка 10^5 элементов, большинство из которых будут близки или равны нулю. Для выделения текстурных характеристик мы применили метод анализа главных компонент (the Principal Component Analysis, PCA) для набора исследуемых матриц совместной встречаемости. Было обнаружено, что всего лишь несколько главных компонент обычно представляют 95% вариации в анализируемых данных (Рис. 5). Поэтому указанный порог был использован в качестве критерия отбора главных компонент (текстурных признаков).

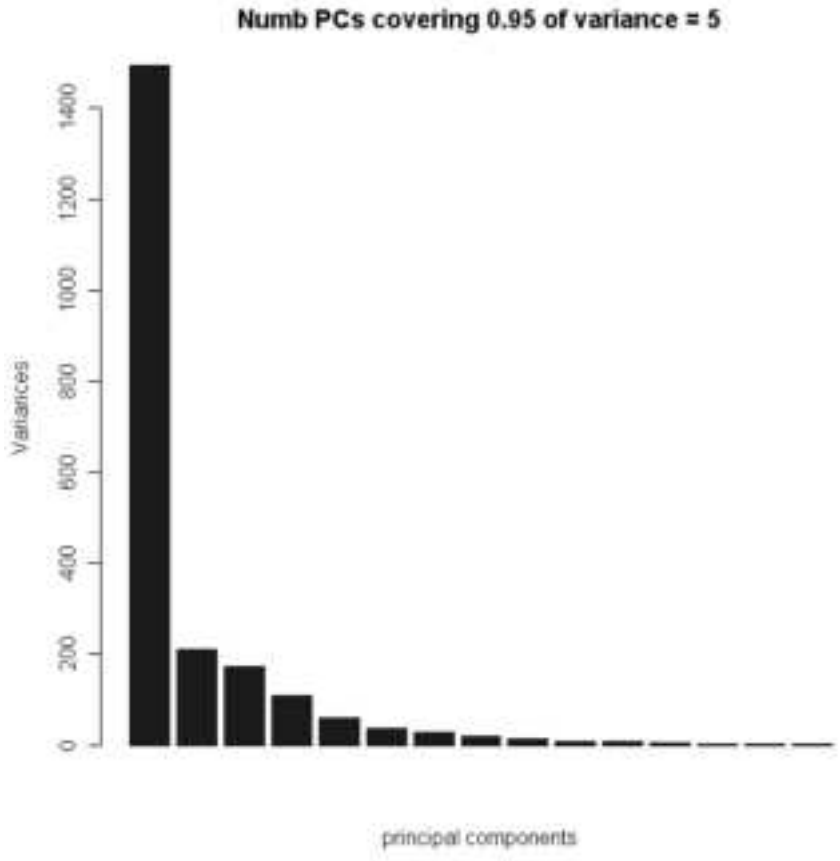


Рис. 5. График важности главных компонент (доля объясняемой вариации данных при использовании в качестве дескрипторов текстуры обобщенных матриц совместной встречаемости типа IIGGAD).

Значимость текстурных отличий и их потенциальная выгода для процесса диагностики были оценены путем выполнения классификации изображений. Для того чтобы избежать ложной интерпретации результатов классификации и их зависимости от специфических свойств одного конкретного метода классификации, классификация изображений опухолей была проведена при помощи трёх наиболее передовых методов, недавно предложенных в области распознавания образов. Они включают в себя: метод иерархической кластеризации, метод опорных векторов (SVM), и метод случайных лесов, реализованные в пакете R [6]. Во всех экспериментах изображения были разделены случайным образом на обучающую и тестовую последовательность без пересечения этих множеств. Для оценки зависимости результатов от размеров и конкретного содержания обучающей выборки, эксперименты проводились случайным разделением на 50% и 80% для изображений обучающей выборки и остальные 50% и 20% для тестирующей. Каждый эксперимент классификации повторялся 1000 раз для достижения достоверной статистики средней точности классификации.

3. Результаты

На первом этапе была изучена эффективность различных видов матриц совместной встречаемости, описанных в предыдущем разделе. В результате было обнаружено, что матрицы gAD , представляющие свойства анизотропии изображений щитовидной железы, содержат информацию, наиболее полезную для целей классификации. В качестве иллюстрации на рис. 6 приведен пример распределения углов между векторами градиентов для $D = 1$. Значения интенсивности и значения градиента были менее релевантны решаемой задаче (максимальное значение t для критерия Стьюдента $t = 3.34$ против максимум $t = 6.25$ для gAD) при определении зоба, аденомы и рака. Наиболее вероятно, что причиной этого является большая наблюдаемая вариабельность исходных изображений, а также то, что интенсивность сигнала неоткалибрована и сильно меняется от случая к случаю. Таким образом, всего 7 главных компонент (текстурных признаков) было отобрано для проведения классификации, 4 из которых были взяты из матриц gAD и 3 из обобщенных матриц IIGGAD.

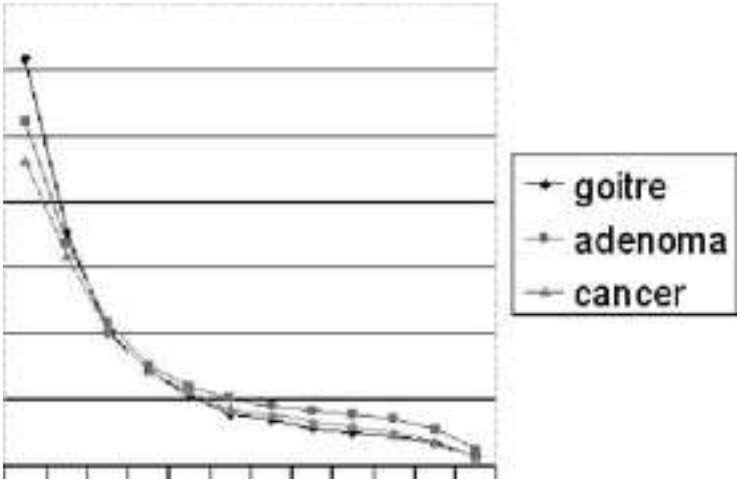


Рис. 6. Типичный пример распределения углов между векторами градиентов для зоба (goitre), аденомы (adenoma) и рака (cancer) щитовидной железы.

Результаты попарной классификации опухолей трех типов, а также классификация доброкачественных (зоб и аденома) относительно злокачественных опухолей подытожены в таблице.

Заключение

Результаты исследований, выполненных в рамках данной работы с использованием 64 трехмерных ультразвуковых изображений опухолей щитовидной железы, представляющих 19 случаев зоба, 12 случаев аденомы, и 33 случая рака позволяют сделать следующие выводы.

Метод кластеризации	Размер обучающей выборки, %	Средня точность классификации в % (1000 повторов)			
		зоб vs. рак	аденома vs. рак	зоб vs. аденома	(зоб+аденома) vs. рак
Иерархическая кластеризация	50	83.4	78.5	60.0	77.3
SVM		73.1	73.8	75.7	64.8
Случайные леса		78.2	72.7	80.8	63.5
Иерархическая кластеризация	80	56.4	52.2	66.7	54.2
SVM		78.7	79.7	82.4	67.3
Случайные леса		79.2	75.5	85.9	62.1

1. Свойства анизотропии текстуры ультразвукового изображения, в отличие от интенсивности и градиента, более важны для определения типа опухоли щитовидной железы.
2. Метод случайных лесов подходит для классификации опухолей с помощью текстурных признаков ультразвуковых изображений больше, чем метод опорных векторов и метод иерархической кластеризации.
3. Метод иерархической кластеризации превосходит другие методы в случаях, когда существует ограниченное количество изображений опухолей.
4. Текстурные свойства ультразвуковых изображений могут быть использованы в процессе компьютеризированной диагностики опухолей щитовидной железы для улучшения качества диагностики и повышения его объективности.

Благодарности. Данная работа выполнена при частичной финансовой поддержке Европейского Сообщества в рамках проекта INTAS 04-77-7036.

Список литературы

1. *K. Colquhoun, A. Alam.* Basic science: ultrasound, D. Wilson (Ed.), Radiology for the FRCS. Current Orthopedics, vol 19, pp. 27-33, 2005.
2. *R.N. Rohling And A.H. Gee.* Issues In 3-D Free-Hand Medical Ultrasound Imaging // CUED/F-INFENG/TR 246 January 1996 Cambridge University, Engineering Department
3. *S. Schlogl, E. Werner, M. Lassmann, J. Terekhova, S. Muffert, S.Seybold, and C. Reiners.* The use of three-dimensional ultrasound for thyroid volumetry // Thyroid, 11(6):569-574, 2001.
4. *A.L. Bogush, A.V. Tuzikov, S.A. Sheynin.* 3D Object Reconstruction from Non-parallel Cross-sections. 17th International Conference on Pattern Recognition ICPR'2004, 23-26 August 2004, Cambridge, UK, vol. 3, pp. 542-545, 2004
5. *V.Kovalev and M.Petrou,* Multidimensional Co-occurrence Matrices for Object Recognition and Matching, Graph. Models Image Processing, vol.58, no.3, pp.187-197, 1996.
6. *R Development Core Team.* R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2006, ISBN 3-900051-07-0.

Статья поступила в редакцию 21.04.2008

УДК 510.6

АДЕКВАТНОСТЬ ИНТЕЛЛЕКТА И ГЕРАКЛИТОВО СОСУЩЕСТВОВАНИЕ ПРОТИВОПОЛОЖНОСТЕЙ

© Брусенцов Н.П.

МГУ им. М.В.Ломоносова, факультет ВМиК
г. Москва, Россия

e-mail: *ramil@cs.msu.su*

Abstract. Foundation of adequate intellect, natural as well as artificial, consists in Geraclit's opposites coexisttence principle. Inobservance of this fundamental principle wrecks wisdom logic.

Введение

Указание Дерека Уолтерса [1, с.145] на то, что двухзначная Инь-Ян, предоставляя неограниченную возможность классификации, непригодна для умозаключений, подводит наконец черту под настойчивыми, но тщетными попытками придать двухзначной логике адекватность. Парадоксы, присущие ее материальной импликации, обусловленые принятием закона исключенного третьего, в условиях двухзначности неустранимы. И совершенно справедливо следующее заключение Уолтерса: [1, с.193]:

«Идея о том, что все мысли, вся материя и все законы вселенной могут быть объяснены при помощи «да» и «нет», очень плотно засела в сознании ученых. Вот почему теория Инь-Ян принимается с такой легкостью. Однако такое мышление не дает возможности развиваться новым идеям и является преградой на пути развития научной мысли».

1. Парадоксы импликации

В двухзначной логике отсутствует непременное средство умозаключений – отношение необходимого следования, без которого невозможны выводы, доказательства, исследование логических взаимосвязей, т.е. нет логики. Естественно, что проблема следования была и остается у логиков первостепенной. В условиях двухзначности отношение следования вырождено в материальную импликацию, которая не то чтобы следованием, но даже отношением не является, поскольку не обеспечивает переменности терминов. Тем, что термин может быть противоречивым либо общезначимым, порождаются парадоксы – «из противоречивого следует все, что угодно», «общезначимое следует из чего угодно».

Устранение парадоксов импликации стало актуальной проблемой уже в средневековье. Тогда Дунс Скот осознал, что «формальное следование имеет место..., когда оба термина изменчивы» [2, с.158]. Впрочем, именно законом Дунса Скота называют первый из приведенных выше парадоксов.

Уильям Оккам придал импликации трехзначность [2, с.143], что открывало возможность отождествления ее с адекватным следованием, осуществить которую ему не удалось.

В 20-м веке алгебраизация логики вновь актуализировала проблему парадоксов импликации. Были изобретены строгие, сильные, релевантные и другие импликации, но ни одна из них не оказалась полноценным следованием - парадоксы, модифицируясь, сохранялись. Парадоксальными оказались и формально сконструированные трехзначные импликации.

Вместе с тем естественное и простое решение проблемы обнаружилось у Аристотеля: «Непостижимое» отношение необходимого следования в его силлогистике представлено общеутвердительным суждением (посылкой) «Все x суть y». Неотобразимость силлогистики в современных логических исчислениях обусловлена тем, что положенный в основание их догматический закон исключенного третьего (ЗИТ) несовместим с ее первоосновой – диалектическим принципом сосуществования противоположностей (ПСП) Гераклита [3, 4].

2. Соблюденность ПСП в аристотелевской силлогистике

Аристотелем ПСП не сформулирован и не заимствован от Гераклита, подлинная суть диалектики которого ему не была известна, о чем свидетельствует следующий фрагмент из «Метафизики» [1005b]: «не может кто бы то ни было считать одно и то же существующим и не существующим, как это, по мнению некоторых, утверждал Гераклит». Но ведь у Гераклита утверждается не совместность («единство») противоположностей, а сосуществование их, благодаря которому, сопоставлением их, только и выявляется определяемое ими качество.

Тем не менее в силлогистике гераклитов ПСП безукоризненно соблюден, поскольку Аристотелем создана адекватная логика бытия, т.е. именно тот Логос, принцип которого установил Гераклит. Убедиться в этом позволяет аристотелево определение необходимого следования в «Первой аналитике» [57b]:

«...когда два [объекта] относятся друг к другу так, что если есть один, необходимо есть и второй, тогда, если нет второго, не будет и первого; однако если второй есть, то не необходимо, чтобы был первый. Но невозможно, чтобы одно и то же было необходимо и когда другое есть, и когда его нет».

Пусть «первым» будет x, а «вторым» – y, интерпретируемые как качества, совокупностями которых представлены сущности рассматриваемых объектов (вещей), в частности, и сущности самих качественных терминов. Фраза «если есть x, то необходимо есть и y» в силлогистике равнозначна суждениям: «в сущности x содержится сущность y», «качеству x присуще качество y», «невозможно x без y», «все x суть y». Отождествление ее в определении следования с «если нет y, то не будет и x» означает не что иное как соблюденность ПСП: не могут не сосуществовать «есть x» и «нет x», «есть y» и «нет y». Качество x мыслимо лишь как сосуществование вещей, которым оно (необходимо) присуще, и вещей, которым оно антиприсуще; символически: x-вещей и x'вещей. При этом связываемые отношением термины непременно будут переменными, ни один из них не может быть ни «пустым» (противоречивым), ни общезначимым. Потому и парадоксов нет.

Так устроена ПСП-реальность. Поскольку необходимо сосуществуют x-, x', y- и y'-вещи, несуществование, например, xy'-вещей не может быть несуществованием x-вещей или y'-вещей, а имеет место только тогда, когда x и y' сосуществуют, но несовместимы. Поэтому в подчиненной ПСП силлогистике существование/несуществование вещей обусловлено взаимосвязанностью терминов, совокупностями которых эти вещи определены. Несовместимость x с y', проявляющаяся несуществованием xy', означает, что сущность y содержится в сущности x, $x = xy$, чем и обусловлена несовместимость x с y': $xy' = (xy)y' = xyy'$, а вместе с тем и необходимое следование y из x, ибо когда есть x, то необходимо есть и содержащийся в нем y. Когда же нет y, т.е. когда есть y', то не может быть несовместимого с y' термина x и необходимо будет x'; из y' необходимо следует x', что равносильно следованию y из x (контрапозитивность следования).

Однако если есть y, то не необходимо, чтобы был x, и если нет x, то не необходимо, чтобы не было y. В этих случаях следование нельзя ни утверждать, ни отрицать, оно не невозможно, но и не необходимо.

Двухзначная логика в силу ЗИТ лишена способности отображать подобные взаимосвязи, составляющие исключенное из нее третье. В силлогистике не необходимость выражается частными посылками, например: «Некоторые y суть x», «Некоторые x' суть y'». В условиях же двухзначности частных посылок нет, как и нет подчиненности частного общему. Более того, понятие подчинения употребляется в противоестественном смысле: считается, что вид подчинен роду, тогда как в действительности виды включены в роды по объему.

Заключение

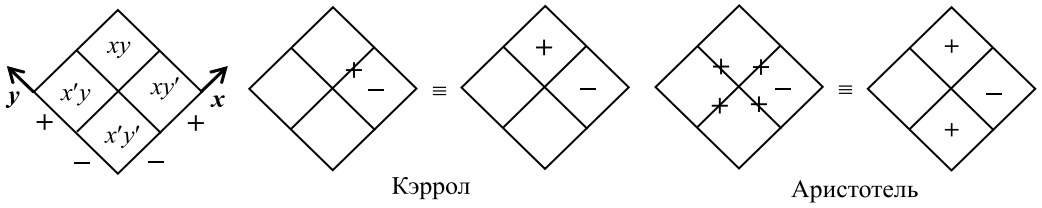
Как видно, принятие ЗИТ подобно Инь-Ян обусловило неполноценность логики, неполноту заложенного в основу ее набора состояний и базисных взаимосвязей. Из упомянутых в «Первой аналитике» посылок [24b] исключена диалектическая (не утверждающая и не отрицающая с необходимостью).

Уолтерс прав: двухзначность недостаточна для умозаключения. Однако только неисключенности третьего для реанимации адекватной логики недостаточно. Созданием трехзначных логик не было достигнуто даже устранение парадоксов имплликации.

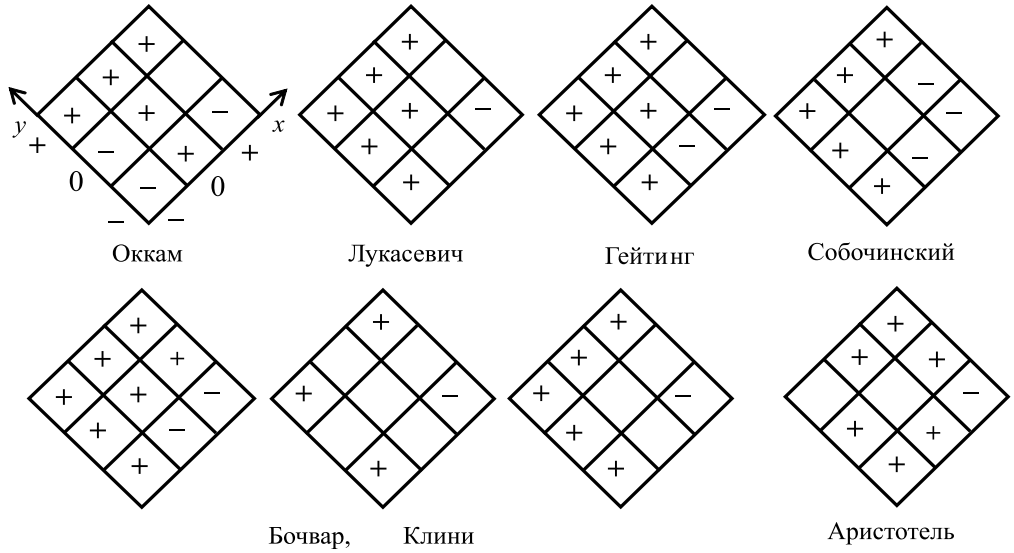
Трехзначная диаграмма Льюиса Кэррола [5] позволила коррекцией предложенного им истолкования общеутвердительной посылки «Все x суть y» выявить, наконец, безупречное представление аристотелева необходимого следования [6]. Но чтобы установить принцип, гарантирующий адекватность мышления, потребовалось убедиться в безупречности силлогистики Аристотеля и понять, почему в ней не возникают парадоксы.

Как оказалось, «ларчик просто открывался»: в основе силлогистики находится гераклитов диалектический принцип сосуществования противоположностей [7].

Трехзначные импликации двухзначных терминов:



Трехзначные импликации трехзначных терминов



Список литературы

1. *Уолтерс Д.* «Книга Великой Тайны» – забытое дополнение к «Книге Перемен». – К.: «София», М.: ИД «Гелиос», 2002.
2. *Стяжкин Н.И.* Формирование математической логики. - М.: Наука, 1967.
3. *Брусенцов Н.П.* Дедукция, диалектика и аристотелево содержательное следование // Искусственный интеллект, 2'2000, – Донецк, 2000. С. 285-291.
4. *Брусенцов Н.П.* Интеллект и диалектическая триада // Искусственный интеллект, 2'2002. – Донецк, 2002. С. 53-57.
5. *Кэррол Л.* Символическая логика // Л. Кэррол. История с узелками / Пер. с англ. М., 1973.
6. *Брусенцов Н.П.* Диаграммы Льюиса Кэррола и аристотелева силлогистика // Вычислительная техника и вопросы кибернетики. Вып. 13. – М.: Изд-во Моск. ун-та, 1977. С. 164-182.
7. *Брусенцов Н.П., Владимирова Ю.С.* Конструктная компьютеризация силлогистики // Математические методы распознавания образов: 13-я Всероссийская конференция. М.: МАКС-Пресс, 2007. С. 10-13.

Статья поступила в редакцию 16.04.2008

УДК 510.67-519.24

Меры опровержимости и расстояния на многозначных экспертных высказываниях в адаптивных методах построения логических решающих функций[1]

© Викентьев А.А., Викентьев Р.А.

Институт Математики им.академика С.Л. Соболева СО РАН
Новосибирский гос. университет, факультет механико-математический
пр-т академика Коптюга, 4, г. Новосибирск, Россия, 630090

e-mail: *vikent@math.nsc.ru*

Abstract. In the paper discusses logical statements of experts as logical formulas in n-valued logic. By makings use of the methods of mathematical logic and the model theory offer the techniques for introducing metric on these statements and measure of their refutability. Study the properties of entered metric and them measures of refutability.

Введение

В настоящее время появляется большой интерес к построению решающих функций на основе анализа экспертной информации, заданной в виде вероятностных логических высказываний нескольких экспертов, реализации адаптивных алгоритмов и согласования высказываний [1-6]. Важную роль при этом имеют развитие адаптивных методов построения логических решающих функций и согласования высказываний. В данной работе предложено записывать высказывания экспертов в виде формул n-значной (с их значениями истинности,$n > 2$) логики.На значения истинности таких формул можно так же смотреть и как на вероятности. В произвольном n-значном случае найдено правильное (с точки зрения главных экспертов) обобщение расстояния между такими формулами и меры опровержимости таких формул, что позволяет решать более утонченно (по сравнению с 2-значным случаем) прикладные задачи. В частности, значение истинности на модели (введенное аналогично случаю $n = 2$) может служить и субъективной вероятностью этой части реализации формулы в модели языка 1-го порядка.

Ясно, что различные такие высказывания экспертов (и соответствующие им формулы) несут в себе разное количество информации, а, значит, возникает вопрос о ранжировании высказываний экспертов и сравнении их по информативности (то есть мере опровержимости при не пустом высказывании). Для решения этих задач в работе будут введены и исследованы функция расстояния (см. [1]) между двумя такими формулами и мера опровержимости формул. Рассмотрен вопрос применимости данного подхода с учетом и обобщением случаев $n = 2$, $n = 3$. При организации поиска логических закономерностей и построении решающих функций требуются расстояния между высказываниями экспертов и формулами в моделях в произвольный (текущий) момент времени с фиксированными знаниями. Планируем обработку сообщений экспертов в произвольной (фиксированной) n-значной логике в различные

[1]Работа выполнена при поддержке РФФИ, проект № 07-01-00331а

моменты (срезы) времени в связи с возможностью того, что исходные общие гипотезы = предположения у экспертов, вообще говоря, могут изменяться. Значит, будет происходить адаптация во времени самой теории (по знаниям экспертов), и, соответственно этому, будем применять подходящие в это момент модели экспертов. Сигнал о смене класса моделей (а, значит, и теории) будет исходить либо от самих экспертов (по их изменяющимся общим знаниям), либо при получении неправильных результатов по расстояниям инженером-разработчиком Базы Знаний при использовании старых аксиом-знаний. Аппарат для обработки таких знаний в логических исчислениях подготовлен в работах Викентьева А.А., начатых со Лбовым Г. С. и Кореневой Л. Н. Результаты этих работ для n-значной логики обобщаются на n-значное исчисление предикатов при соответствующих аналогах для подмножеств предикатов фиксированной истинности в модели.

1. Основные определения и понятия

Определение 1. Множество элементарных высказываний $S^n(\phi)$, используемых при написании формулы многозначной логики ϕ, назовем *носителем формулы* ϕ.

Определение 2. Назовем *носителем совокупности знаний* $S^n(\Sigma)$ объединение носителей формул, входящих в Σ (множество формул), т. е. $S^n(\Sigma) = \bigcup\limits_{\phi\in\Sigma} S^n(\phi)$.

Определение 3. Назовем *множеством возможных значений носителя* совокупности формул (знаний) с указанием (всевозможных) их значений истинности (далее нас интересуют отличные от нуля) $Q_n(\Sigma) = \left\{\phi_{\frac{k}{n-1}} \;\middle|\; \phi \in S^n(\Sigma),\ k = 0, \ldots, n-1\right\}$.

Определение 4. Моделью M назовем любое подмножество $Q_n(\Sigma)$ такое, что M не содержит одновременно $\phi_{\frac{k}{n-1}}$ и $\phi_{\frac{l}{n-1}}$ при любых $k \neq l$ и $\phi \in Q(\Sigma)$.

Множество всех моделей будем обозначать $P^n(S(\Sigma))$.

Для упрощения записи верхний индекс n в формулах, означающий значность логики, будем опускать.

Теорема 1 (о мощности $P^n(S(\Sigma))$). $|P(S(\Sigma))| = n^{|S(\Sigma)|}$.

Доказательство. Доказательство проводится по индукции. □

Введем обозначение для множества моделей формулы A с фиксированным для нее значением истинности:

$$Mod_{S(\Sigma)}(A)_{\frac{k}{n-1}} = \left\{M \;\middle|\; M \in P(S(\Sigma)),\ M \models A_{\frac{k}{n-1}}\right\}.$$

Несложно доказываются всевозможные теоретико-модельные свойства связи между моделями формул и моделями их компонент с фиксированными вероятностями. Формулы назовем эквивалентными, если они имеют одно и тоже множество моделей в каждом фиксированном значении истинности. Это отношение является отношением эквивалентности.

Определение 5. Расстоянием между формулами ϕ и ψ, такими, что $S(\phi)\bigcup S(\psi) \subseteq S(\Sigma)$, на множестве $P(S(\Sigma))$ назовем (нормированную симметрическую разность в многозначном случае, что является естественным обобщением введения расстояния в классическом двузначном случае) величину

$$\rho_{S(\Sigma)}(\phi,\psi) = \frac{\left|\bigcup\limits_{k=1}^{n-1} Mod_{S(\Sigma)}\left(\phi_{\frac{k}{n-1}} \wedge \psi_0\right)\right| + \left|\bigcup\limits_{k=1}^{n-1} Mod_{S(\Sigma)}\left(\phi_0 \wedge \psi_{\frac{k}{n-1}}\right)\right|}{n^{|S(\Sigma)|}}.$$

2. Свойства расстояний и мер опровержимости на многозначных формулах

Теорема 2 (о свойствах расстояния $\rho_{S(\Sigma)}(\phi,\psi)$)**.** *Для любых формул ϕ, ψ, таких, что $S(\phi)\bigcup S(\psi) \subseteq S(\Sigma)$ верны следующие утверждения:*

1. $0 \le \rho_{S(\Sigma)}(\phi,\psi) \le 1$;
2. $\rho_{S(\Sigma)}(\phi,\psi) = \rho_{S(\Sigma)}(\psi,\phi)$;
3. $\rho_{S(\Sigma)}(\phi,\psi) = 0 \Leftrightarrow \phi \equiv \psi$;
4. $\rho_{S(\Sigma)}(\phi,\psi) = 1 \Leftrightarrow \bigcup\limits_{l=1}^{n-1}\bigcup\limits_{k=1}^{n-1}\left(Mod(\phi)_{\frac{k}{n-1}} \uplus Mod(\psi)_{\frac{l}{n-1}}\right) = P(S(\Sigma))$.
5. $\rho_{S(\Sigma)}(\phi,\psi) \le \rho_{S(\Sigma)}(\phi,\chi) + \rho_{S(\Sigma)}(\chi,\psi)$;
6. *Если* $\phi^1 \equiv \phi^2$, *то* $\rho_{S(\Sigma)}(\phi^1,\psi) = \rho_{S(\Sigma)}(\phi^2,\psi)$;

Доказательство. Доказательства всех пунктов следуют из определений, построения примеров, симметричности логических связок и непосредственных длинных и рутинных вычислений. □

Замечание. Поскольку в доказательствах не используют свойств множества $P(S(\Sigma))$, то расстояние можно рассматривать на любом его подмножестве, если это необходимо или вытекает из условий конкретной задачи. Если же введенное расстояние рассматривается на всем множестве $P(S(\Sigma))$, то возникает вопрос можно ли сделать вычисление более простым и удобным. Поскольку носители рассматриваемых формул составляют небольшое подмножество, то может ли хватить моделей, построенных из этих компонент формул, для которых надо найти расстояние. На все эти вопросы – пожелания получен положительный ответ. Оказывается выполняется свойство локальности для вычисления расстояния между двумя формулами: следующая теорема говорит о возможности такого упрощения вычисления.

Теорема 3 (О локальности нахождения расстояния)**.** *Для любого $S(\Sigma_0)$ такого, что $S(\phi)\bigcup S(\psi) \subseteq S(\Sigma_0)$ и любого $S(\Sigma_1)$ такого, что $S(\Sigma_0) \subseteq S(\Sigma_1)$, имеет место равенство:*

$$\rho_{S(\Sigma_0)}(\phi,\psi) = \rho_{S(\Sigma_1)}(\phi,\psi).$$

Доказательство. Доказательство аналогично классическому случаю с учетом многозначности значений истинности. □

Заметим, что эта теорема позволяет уменьшить количество основных множеств и ограничить сверху мощности носителей моделей при подсчете расстояний.

Подход к определению меры опровержимости основывается на естественном предположении, что чем больше моделей на которых высказывание принимает значение не равное 1, тем высказывание легче опровержимо. Поскольку в нашем случае значений не равных 1 у формулы несколько, то предлагается учитывать их с весами монотонно по этим значениям и для каждого такого значения истинности нормированными. Перейдем к формальному определению.

Определение 6. Мерой опровержимости $I_{S(\Sigma)}(\phi)$ для формул из $\Phi(\Sigma)$, где $\Phi(\Sigma) = \{\phi \mid S(\phi) \subset S(\Sigma)\}$, назовем величину

$$I_{S(\Sigma)}(\phi) = \sum_{i=0}^{n-2} \alpha_i \frac{\left|Mod_{S(\Sigma)}\left(\phi_{\frac{i}{n-1}}\right)\right|}{n^{|S(\Sigma)|}},$$

где α_i удовлетворяет условиям: $0 \le \alpha_i \le 1$, $\alpha_i + \alpha_{n-1-i} = 1$, $\alpha_k \ge \alpha_i$, для всех $i = 0, \ldots, \frac{n-1}{2}$ и всех $k = 0, \ldots, i$.

Теорема 4 (свойства меры опровержимости $I_{S(\Sigma)}$). *Для любых* $\phi, \psi \in \Phi(\Sigma)$ *верно*

1. $0 \le I_{S(\Sigma)}(\phi) \le 1$;
2. $I_{S(\Sigma)}(\phi) + I_{S(\Sigma)}(\neg\phi) = 1$;
3. $I_{S(\Sigma)}(\phi \wedge \psi) \ge \max\{I_{S(\Sigma)}(\phi), I_{S(\Sigma)}(\psi)\}$;
4. $I_{S(\Sigma)}(\phi \vee \psi) \le \min\{I_{S(\Sigma)}(\phi), I_{S(\Sigma)}(\psi)\}$;
5. $I_{S(\Sigma)}(\phi \vee \psi) + I_{S(\Sigma)}(\phi \wedge \psi) = I_{S(\Sigma)}(\phi) + I_{S(\Sigma)}(\psi)$;
6. $I^3_{S(\Sigma)}(\phi \wedge \psi) = \frac{1}{2}\left(I^3_{S(\Sigma)}(\phi) + I^3_{S(\Sigma)}(\psi) + \rho^3_{S(\Sigma)}(\neg\phi, \neg\psi)\right)$;
7. $I^3_{S(\Sigma)}(\phi \vee \psi) = \frac{1}{2}\left(I^3_{S(\Sigma)}(\phi) + I^3_{S(\Sigma)}(\psi) - \rho^3_{S(\Sigma)}(\neg\phi, \neg\psi)\right)$.

Доказательство. Доказательства пунктов теоремы либо очевидны, либо состоят в подробном расписывании левых частей выражений, с использованием принципа симметрии, подробных прямых вычислений и их применений. □

Доказанные теоремы указывают нам общие свойства меры опровержимости и расстояний, а при $n = 3$ говорят о справедливости гипотезы Г.С. Лбова, доказанную первым автором при $n = 2$ (см. [1]). При $n > 3$ такой связи с расстоянием нет, но есть более сложная и она найдена. В частном случае $n = 3$ доказаны также дополнительные свойства расстояний и меры опровержимости, похожие на случай $n = 2$ (см., например, [1]). Все полученные выше результаты использованы при написании программ и апробированы на прикладных задачах при различных (конкретных) значениях n. Подбор нужного значения n в конкретной прикладной задаче является частью процесса адаптации для введения расстояния и меры опровержимости для получения более тонких знаний. В общем случае $n > 2$ проведены дальнейшие теоретические исследования и рассмотрены различные прикладные аспекты. Проведена на примерах обработка сообщений экспертов в произвольной (фиксированной) n-значной логике в различные моменты (срезы) времени с изменением исходных общих гипотез-предположений экспертов. Результаты примеров показали адекватность предлагаемого подхода и качественное отличие результатов для различных n, и то, что с ростом n они все меньше и меньше отличаются.

Например, было рассмотрено дерево событий («отказа» работы заправочной станции), используемого для анализа причин возникновения аварийных ситуаций при автоматизированной заправке емкости. Структура дерева событий включает одно головное событие (авария, инцидент), которое соединяется с набором соответствующих нижестоящих событий (ошибок, отказов, неблагоприятных внешних воздействий), образующих причинные цепи (сценарии аварий). Проанализированы по дереву различные высказывания об отказах заправочной станции и найдены расстояния между различными формулами и их меры опровержимости при различных n. Логический знак «&» (или and) означает, что вышестоящее событие в дереве возникает при одновременном наступлении нижестоящих событий. Знак «V» означает, что вышестоящее событие может произойти вследствие возникновения одного из нижестоящих событий. Обозначим базовые события, которые на имеющемся у нас дереве записаны цифрами в кружках, через $A1, \ldots, A13$. События, появление которых приводит к аварии, можно записать, например, следующим образом: $A12$ V $A13$, $A5\&A6\&A7\ldots$ С другой стороны, существует набор событий, который гарантирует не возникновение главного (головного) события при условии, что если ни одно из событий, входящих в него, не произойдет. Так, например, авария не произойдет, если не будет событий ($A1, A2, A3, A4, A5, A12, A13$) или событий ($A7, A8, A9, A10, A11, A12, A13$).

Нами детально изучались следующие укрупненные реально возможные события, которые зададим неформально. $A(1)$ – это событие состоит в том, что произойдет обрыв цепей от датчиков объёма дозы или одновременно откажет расходомер и датчик уровня. Возникновение только этого события не ведет к головному событию, т.е. к аварии.

$A(2)$ – это событие, заключающееся в том, что оператор не знал о необходимости отключения насосов или отказал расходомер или произошло то, что одновременно отказали расходомер и датчик уровня, приведет к аварии. $A(3)$ - отказ расходомера и отсутствие реакции оператора на отказ САВД (датчиков) приведет к головному событию.

$A(4)$ – событие–отказ средств выдачи сигналов или отключение САВД не повлечет аварийной ситуации.

$A(5)$ – при отказе выключателя насоса сразу возникает авария.

$A(6)$ – при совокупности этих (А2 или А12 или одновременно А5 и А6) событий авария также наступает.

$A(7)$ – если авария возникнет вследствие неосуществления команды на отключение. Теперь, взяв формальные выражения для приведенных различных событий, можно посчитать расстояния между ними и меру их опровержимости. Вычислены таблицы в которых представлены посчитанные для каждой пары высказываний расстояния в многозначных логиках для различных n. Расстояния в таблицах получились различными, но с ростом n разность между ними уменьшается. Составлены так же таблицы в которых представлены посчитанные для каждого такого события- высказывания мера опровержимости в различных n-значных логиках и при различных

параметрах в общей формуле. Замечено, что с ростом параметра n происходит все меньшее отличие получаемых ответов.

Заключение

В работе получены следующие результаты: сформулированы определения расстояния и меры опровержимости на многозначных формулах, и доказаны их основные свойства. Полученные результаты переносятся на n-значный фрагмент логики 1-го порядка. Предложенные расстояния и мера опровержимости могут быть использованы при пополнениях конкретных баз знаний, кластеризации знаний, поиску противоречивости высказываний экспертов и при разработке адаптивных алгоритмов.

Список литературы

1. *Лбов Г.С., Старцева Н.Г.* Логические решающие функции и вопросы статистической устойчивости решений – Новосибирск: Изд-во Ин-та математики СО РАН, 1999.
2. *Кейслер Г., Чэн Ч. Ч.* Теория моделей – М.: Мир, 1977.
3. *Викентьев А.А, Лбов Г.С.* О метризации булевой алгебры предложений и информативности высказываний экспертов – Доклады РАН 1998.Т.361, №2 С.174-176.
4. *Викентьев А.А, Лбов Г.С.* Setting the metric and informativeness on statements of experts – Pattern Recognition and Image Analysis. 1997 V.7, N2, P. 175-183.
5. *Ершов Ю.Л., Палютин Е.А.* Математическая логика – Санкт-Петербург, 2004.
6. *Викентьев А.А., Коренева Л.Н.* К вопросу о расстояниях между формулами, описывающими структурированные объекты – Математические методы распознавания образов (ММРО-99). РАН ВЦ, Москва, 1999. С. 151-154.

Статья поступила в редакцию 30.04.2008

УДК 519.237.8+510.22

МЕТОД МЯГКОЙ ИНТЕРПРЕТАЦИИ РЕЗУЛЬТАТОВ НЕЧЕТКОЙ КЛАСТЕРИЗАЦИИ

© Вятченин Д.А.

Объединенный институт проблем информатики НАН Беларуси
ул. Сурганова, 6, г. Минск, Беларусь, 220012

e-mail: *viattchenin@mail.ru*

Abstract. The paper provides a new method of the interpretation of fuzzy clustering results which is allow put an automatic choosing of the threshold value that is a basis for detecting of sets of most informative elements of fuzzy classes. An effectiveness of the new method is illustrated on an example of fuzzy data fuzzy clustering results processing.

Введение

В последние годы значительно выросло количество публикаций, посвященных различным аспектам нечеткого подхода к решению задач автоматической классификации, или, иными словами, нечеткой кластеризации. Это обусловлено тем, что методы нечеткой кластеризации отличаются от традиционных «жестких» методов кластеризации, с одной стороны, высокой точностью, а с другой – содержательной осмысленностью результатов классификации. Как и в традиционных методах кластерного анализа, в рамках нечеткого подхода к решению задачи автоматической классификации выделяются эвристическое, оптимизационное и иерархическое направления, подробно рассматриваемые в работе [1]. Наиболее распространенным подходом к решению нечеткой модификации задачи автоматической классификации является оптимизационный подход [2], методы которого предусматривают нахождение оптимального, в смысле используемого критерия качества $Q(P(X))$, разбиения $P^*(X) = \{A^1, \ldots, A^c\}$ на заданное число c нечетких кластеров, описываемых функциями принадлежности μ_{li}, $l = 1, \ldots, c$, $i = 1, \ldots, n$, определенных на исследуемой совокупности объектов $X = \{x_1, ..., x_n\}$, так что задача нечеткой кластеризации заключается в нахождении экстремума целевой функции $Q(P(X))$, что в общем виде описывается формулой

$$Q(P(X)) \to \underset{P(X)\in\Pi}{extr}, \tag{1}$$

где Π – множество всех возможных нечетких разбиений $P(X)$ множества классифицируемых объектов X, при ограничениях, определяемых условием

$$\mu_{li} \geq 0, \quad \sum_{i=1}^{n} \mu_{li} = 1, \quad i = 1, \ldots, n, \quad l = 1, \ldots, c, \tag{2}$$

именуемым также условием нечеткого c-разбиения или нечеткого разбиения в смысле Распини [3], которое описывается матрицей $P_{c\times n} = [\,\mu_{li}]$, где $\mu_{li} = \mu_{A^l}(x_i)$ – значение принадлежности элемента $x_i \in X$ некоторому нечеткому кластеру $A^l \in \{A^1, \ldots A^c\}$.

Постановка проблемы, рассматриваемой в статье, заключается в отнесении объекта классифицируемой совокупности одному или нескольким классам в случае решения задачи классификации с помощью оптимизационных методов нечеткой кластеризации.

Анализ существующих результатов решения указанной проблемы, проведенный в рамках предпринятого исследования, демонстрирует, что в процессе интерпретации результатов нечеткой классификации в ряде случаев могут возникнуть трудности, связанные с проблемой однозначного отнесения объекта к тому или иному классу; кроме того, существующие подходы являются жесткими в том смысле, что при отнесении объекта к тому или иному классу, ассоциированному с соответствующим нечетким кластером полученного нечеткого c-разбиения, значения принадлежностей объектов нечетким кластерам элиминируются, так что применение нечеткой кластеризации с методологической точки зрения теряет смысл.

Целью исследования является обоснование метода «мягкой» интерпретации результатов нечеткой кластеризации, позволяющего, с одной стороны, отнести каждый объект исследуемой совокупности к наименьшему числу $\tilde{c}$, $1 \le \tilde{c} \le c$ нечетких кластеров нечеткого c-разбиения $P^*(X) = \{A^1, \ldots, A^c\}$, являющегося результатом классификации, а с другой – сохранить значения принадлежности μ_{li}, которые можно интерпретировать как степени обладания объектом $x_i \in X$ свойств класса, ассоциированного с нечетким кластером A^l, $l \in \{1, \ldots, c\}$ – элементом нечеткого c-разбиения $P^*(X)$, оптимального в смысле выбранного критерия качества $Q(P(X))$.

1. Основные методы интерпретации результатов нечеткой кластеризации

Наиболее простым и распространенным методом интерпретации результатов нечеткой классификации является дефаззификация матрицы $P_{c\times n} = [\,\mu_{li}]$ нечеткого c-разбиения по правилу максимального значения принадлежности [4]:

$$P_i^{MM} = \mathrm{e}_l \Leftrightarrow \mu_{li} > \mu_{ai},\ a = 1, 2, \ldots, c,\ a \neq l, \tag{3}$$

так что значениями принадлежности μ_{li}^{MM} матрицы $P_{c\times n}^{MM}$ являются числа 0 и 1, и принадлежность i-го объекта l-му классу определяется по формуле

$$\mu_{li}^{MM} = \begin{cases} 1, & \mu_{li} > \mu_{ai},\ a = 1, 2, \ldots, c, a \neq l \\ 0, & \text{в противном случае} \end{cases}. \tag{4}$$

Однако подобный подход является неприемлемым, если для некоторого объекта $x_i \in X$ его значения принадлежностей составляют $\mu_{li} = 1/c$, $l = 1, \ldots, c$.

С другой стороны, В. Педричем [5] для выявления структуры нечетких кластеров предлагается ввести два порога, один из которых определяется исследователем, а значение второго вычисляется, и которые позволяют выделять ядра нечетких кластеров. Так как значение μ_{li} выражает степень принадлежности i-го элемента l-му кластеру, то объект $x_i \in X$ может рассматриваться как элемент ядра нечеткого кластера A^l, если для некоторого порога φ имеет место $\mu_{li} > \varphi$. При рассмотрении для каждого $x_i \in X$ структурного отношения ко всем $A^l, l = 1, \ldots, c$, принимаются во внимание функции принадлежности точки x_i всем остальным нечетким кластерам,

за исключением рассматриваемого нечеткого кластера A^l. Поскольку число кластеров равно c, можно отметить, что любая точка $x_i \in X$, для которой $\mu_{li} = \frac{1}{c}, l = 1, \ldots, c$ не способствует выявлению сущности кластерной структуры. Мерой структурного свойства некоторой точки $x_i \in X$ служит показатель

$$\xi(x_i) = 1 - c^c \prod_{l=1}^{c} \mu_{li}, \tag{5}$$

то есть $\xi : X \to [0, 1]$, и из выражений (2) и (5) следует, что если для некоторого x_i и одного из кластеров A^l, $l = 1, \ldots, c$, будет $\mu_{li} = 1$, то значение показателя (5) в x_i максимально, то есть $\xi(x_i) = 1$, и точка x_i будет элементом ядра нечеткого кластера, если $\xi(x_i)$ превышает порог ϕ. Таким образом, понятие (ϕ, φ)-ядра нечеткого кластера A^l, $l = 1, \ldots, c$, может быть определено как подмножество $X^l_{\phi\varphi}$ универсума $X = \{x_1, ..., x_n\}$ при $\phi, \varphi \in [0, 1]$, содержащее множество $\left\{ x_i \in X \mid 1 - c^c \prod_{l=1}^{c} \mu_{li} \geq \phi,\ \mu_{li} \geq \varphi \right\}$, то есть

$$X^l_{\phi\varphi} = \{ x_i \in X \mid \xi(x_i) \geq \phi,\ \mu_{li} \geq \varphi \}, \tag{6}$$

так что множество ядер $X^l_{\phi\varphi}$, $l = 1, \ldots, c$, и резидуальное множество данных X^R, содержащее все оставшиеся элементы универсума X, связаны соотношением

$$X = \bigcup_{l=1}^{c} X^l_{\phi\varphi} \cup X^R, \tag{7}$$

где первая составляющая правой части является существенной для рассмотрения структурой, а вторая соответствует малосущественной структуре множества X.

Недостатком обоих подходов является утрата значений принадлежности объектов нечетким кластерам, позволяющая содержательно интерпретировать результаты кластеризации.

2. Понятие α-ядра нечеткого кластера

В свою очередь, концепция α-ядер нечетких кластеров, предложенная в работе [6] в рамках разработки методологии многостадийной нечеткой кластеризации, предполагает нахождение такого порога α, $\alpha \in (0, 1]$, чтобы выполнялось условие

$$X = \bigcup_{l=1}^{c} Supp(A^l(\alpha)), \tag{8}$$

где $X = \{x_1, \ldots, x_n\}$ – исследуемая совокупность объектов, α-ядра $A^l(\alpha)$, $l = 1, \ldots, c$, нечетких кластеров $A^l \in P$, $l = 1, \ldots, c$, для $\alpha \in (0, 1]$ представляют собой нечеткие множества уровня [7], определяемые как $A^l(\alpha) = \{(x_i, \mu^\alpha_{li}) \mid \mu^\alpha_{li} \geq \alpha\}$, $x_i \in X$, так что $A^l(\alpha) \subseteq A^l$, $\alpha \in (0, 1]$, $A^l \in \{A^1, \ldots, A^c\}$, а $Supp(A^l(\alpha))$ – носитель α-ядра $A^l(\alpha)$ нечеткого кластера $A^l \in P$, причем $Supp(A^l(\alpha)) = A^l_\alpha$, то есть носитель α-ядра нечеткого кластера $A^l \in P$, $l = 1, \ldots, c$ будет представлять собой α-срез

$A^l_\alpha = \{x_i \in X \mid \mu_{li} \geq \alpha\}$ [8] этого кластера при соответствующем значении α, а значения принадлежности объекта α-ядру нечеткого кластера определяется в соответствии с формулой

$$\mu^\alpha_{li} = \begin{cases} \mu_{li}, & x_i \in A^l_\alpha \\ 0, & x_i \notin A^l_\alpha \end{cases} . \tag{9}$$

Порог α должен выбираться так, чтобы каждый объект $x_i \in X$, $i = 1, \ldots, n$, принадлежал бы по меньшей мере одному α-ядру нечеткого кластера, и может вычисляться по формуле

$$\hat{\alpha} = \min_i \max_l \mu_{li}, \tag{10}$$

что, в свою очередь, позволяет сформулировать следующее утверждение.

Теорема. *Для нечеткого с-разбиения $P = \{A^1, \ldots, A^c\}$ исследуемой совокупности объектов $X = \{x_1, \ldots, x_n\}$ носители $\{A^1_\alpha, \ldots, A^c_\alpha\}$ α-ядер $\{A^1(\alpha), \ldots, A^c(\alpha)\}$ нечетких кластеров образуют покрытие исследуемой совокупности $X = \{x_1, \ldots, x_n\}$ в том, и только в том случае, когда $\alpha \leq \hat{\alpha}$, $\alpha \in (0, 1]$, где $\hat{\alpha} \in (0, 1]$ вычисляется в соответствии с формулой (10).*

Доказательство. Если для некоторого $\alpha \in (0, 1]$ семейство множеств $C = \{A^1_\alpha, \ldots, A^c_\alpha\}$, являющихся носителями α-ядер нечетких кластеров $A^1, \ldots, A^c$, образует покрытие исследуемой совокупности $X = \{x_1, \ldots, x_n\}$, то каждый объект $x_i \in X$ является элементом по меньшей мере одного подмножества $A^l_\alpha \in C$. Если каждый объект $x_i \in X$ является элементом только одного подмножества $A^l_\alpha \in C$, то $\bigcap\limits_{l=1}^{c} A^l_\alpha = \varnothing$, а если каждый объект $x_i \in X$ является элементом более чем одного подмножества $A^l_\alpha \in C$, то будет иметь место $\bigcap\limits_{l=1}^{c} A^l_\alpha \neq \varnothing$. Так как $\bigcup\limits_{l=1}^{c} A^l_\alpha$ – наименьшее множество, содержащее все множества $\{A^1_\alpha, \ldots, A^c_\alpha\}$, и так как $\bigcap\limits_{l=1}^{c} A^l_\alpha \subseteq \bigcup\limits_{l=1}^{c} A^l_\alpha$, то очевидно, что $x_i \in \bigcup\limits_{l=1}^{c} A^l_\alpha$, $\forall i \in \{1, \ldots, n\}$.

В свою очередь, условие (10) можно переписать в следующем виде:

$$\hat{\alpha} = \bigwedge_i (A^1 \vee A^2 \vee \ldots \vee A^c), \tag{11}$$

где символом $\vee$ в теории нечетких множеств традиционно обозначается операция взятия максимума, а символом $\wedge$ – операция взятия минимума [8]. В силу ассоциативности операции $\vee$ имеет место $A^1 \vee A^2 \vee \ldots \vee A^c = A$, и в силу конечности множества X существует по крайней мере один элемент $x_i \in X$, для которого выполняется условие

$$\hat{\alpha} = \bigwedge_i \mu_A(x_i), \tag{12}$$

так что для некоторого $\bar{\alpha} > \hat{\alpha}$, $\alpha \in (0,1]$ будет иметь место $\mu_{A(\bar{\alpha})}(x_i) = 0$, и, как следствие, $x_i \notin Supp(A(\bar{\alpha}))$. Соответственно, будет иметь место $x_i \notin \bigcup_{l=1}^{c} A^l_{\bar{\alpha}}$, что доказывает корректность утверждения теоремы. □

Из теоремы вытекает ряд утверждений, которые, в силу их очевидности и ограниченности изложения, приводятся без доказательства.

Следствие 1. *Если $\alpha = \hat{\alpha}$, $\alpha \in (0,1]$, где значение $\hat{\alpha}$ вычисляется по формуле (10), то покрытие, образуемое носителями $\{A^1_\alpha, \ldots, A^c_\alpha\}$ α-ядер $\{A^1(\alpha), \ldots, A^c(\alpha)\}$ кластеров нечеткого c-разбиения $P(X) = \{A^1, \ldots, A^c\}$, минимально.*

Следствие 2. *Если в условии (8) имеет место равенство, то носители $\{A^1_\alpha, \ldots, A^c_\alpha\}$ α-ядер кластеров нечеткого c-разбиения образуют разбиение исследуемой совокупности $X = \{x_1, \ldots, x_n\}$ на непересекающиеся множества.*

Следствие 3. *В случае нечеткого c-разбиения исследуемой совокупности $X = \{x_1, \ldots, x_n\}$ на два класса при $\hat{\alpha} = 0.5$ в области пересечения носителей α-ядер обоих кластеров будет находиться по меньшей мере один объект $x_i \in X$, $i \in \{1, \ldots, n\}$.*

3. Иллюстративный пример

Эффективность предложенного метода можно проиллюстрировать на примере обработки нечеткого c-разбиения при $c = 3$, полученного с помощью предложенного М.-Ш. Янгом и Ч.-Х. Ко [9] FCN-алгоритма, совокупности 30 треугольных нечетких чисел $V_i = (m, a, b)_T$, $i = 1, \ldots, 30$, где m – модальное значение, а a и b – левый и правый коэффициенты нечеткости соответственно, представленных, как и их значения принадлежностей, в таблице 1.

Значения принадлежностей μ_{li}, $l = \overline{1,3}$, $i = 1, \ldots, 30$, объектов $V_1, \ldots, V_{30}$ классифицируемой совокупности нечетким кластерам A^l, $l = \overline{1,3}$, могут быть изображены в виде так называемой линейной диаграммы [1], представленной на рис. 1, где символом ∘ обозначены значения принадлежностей объектов нечеткому кластеру A^1, символом ■ – нечеткому кластеру A^2, а символом ▲ – нечеткому кластеру A^3.

Подобное представление результатов нечеткой кластеризации значительно усложняет их содержательную интерпретацию, особенно в случаях большого числа объектов исследуемой совокупности или числа нечетких кластеров в искомом нечетком c-разбиении, а также в случаях достаточно высоких значений принадлежности некоторых объектов нескольким нечетким кластерам одновременно.

В результате применения к матрице значений принадлежности $P_{c\times n} = [\mu_{li}]$, $l = \overline{1,3}$, $i = 1, \ldots, 30$, предложенного метода интерпретации, значение порога, позволяющего выделить α-ядра кластеров нечеткого c-разбиения, вычисляемого в соответствии с формулой (10), составило $\hat{\alpha} = 0.5255$, и значения принадлежностей объектов $\mu^{\hat{\alpha}}_{li}$, $l = \overline{1,3}$, $i = 1, \ldots, 30$, полученные по формуле (9), изображены на рис. 2.

Значения принадлежностей объектов α-ядрам нечетких кластеров демонстрируют их хорошую разделимость, что выражается в единственности положительного значения μ^{α}_{li} для каждого объекта V_i, $i = 1, \ldots, 30$ исследуемой совокупности.

Таблица 1. Совокупность 30 нечетких чисел и их значения принадлежностей нечетким кластерам

i	Параметры нечеткого числа			Значения принадлежностей		
	m	a	b	μ_{1i}	μ_{2i}	μ_{3i}
1	3.34	1.46	1.30	0.7836	0.1625	0.0538
2	9.56	0.27	1.00	0.9460	0.0437	0.0102
3	10.56	1.95	1.93	0.9680	0.0261	0.0057
4	10.89	0.56	1.17	0.9779	0.0182	0.0038
5	13.89	0.89	0.88	0.9866	0.0115	0.0018
6	14.78	0.12	1.21	0.9397	0.0529	0.0073
7	14.90	1.19	0.41	0.9511	0.0427	0.0061
8	15.67	1.82	0.90	0.8978	0.0904	0.0116
9	16.87	1.90	1.85	0.7412	0.2345	0.0241
10	17.45	1.79	1.95	0.6468	0.3236	0.0294
11	19.78	1.47	0.42	0.2929	0.6724	0.0345
12	20.67	1.34	1.10	0.1597	0.8124	0.0277
13	21.45	0.92	1.60	0.0760	0.9056	0.0183
14	22.34	0.04	1.58	0.0225	0.9698	0.0076
15	23.47	0.81	0.51	0.0041	0.9940	0.0018
16	24.67	0.14	1.09	0.0034	0.9942	0.0023
17	25.78	0.39	1.51	0.0189	0.9628	0.0181
18	26.45	1.61	0.92	0.0254	0.9473	0.0271
19	28.34	1.95	0.12	0.0556	0.8451	0.0992
20	32.29	1.66	1.64	0.0709	0.4035	0.5255
21	32.77	0.63	0.47	0.0658	0.3517	0.5824
22	34.88	1.08	0.66	0.0360	0.1527	0.8112
23	35.45	1.48	1.26	0.0274	0.1101	0.8624
24	35.88	1.79	0.16	0.0248	0.0978	0.8773
25	38.88	0.66	0.64	0.0004	0.0014	0.9980
26	40.25	0.52	1.71	0.0011	0.0034	0.9953
27	40.47	1.95	0.15	0.0006	0.0017	0.9976
28	43.56	0.92	0.63	0.0164	0.0412	0.9423
29	43.98	1.74	1.69	0.0195	0.0482	0.9321
30	45.77	1.71	0.79	0.0315	0.0735	0.8949

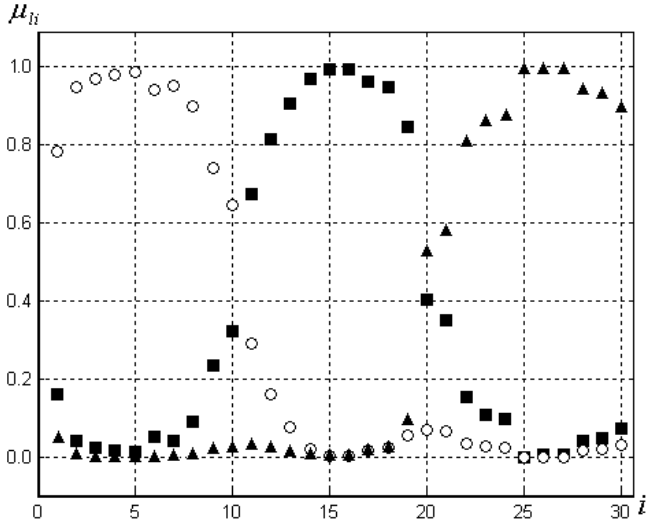


Рис. 1. Диаграмма значений принадлежности объектов нечетким кластерам

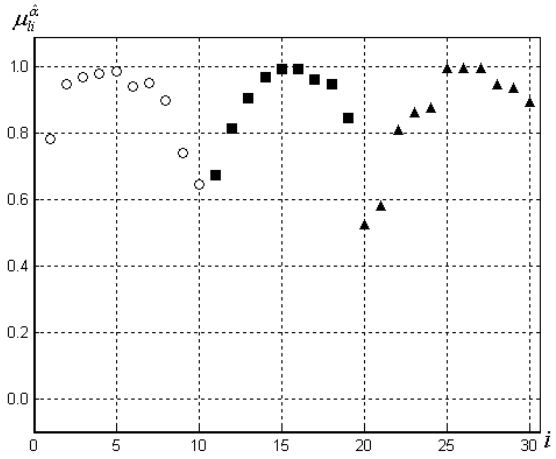


Рис. 2. Диаграмма значений принадлежности объектов α-ядрам нечетких кластеров

Заключение

Результатом проведенного исследования является метод выделения α-ядер нечетких кластеров, представляющих собой наиболее информативные, с точки зрения интерпретации результатов нечеткой классификации, нечеткие подмножества объектов исследуемой совокупности.

Предложенная концепция α-ядер нечетких кластеров обладает большей гибкостью, чем правило максимального значения принадлежности (3) и концепция (ϕ, φ)-ядер нечетких кластеров [5], так как сохраняет значения принадлежности μ_{li}^{α} объектов, что является немаловажным для содержательной интерпретации результатов кластеризации; кроме того, определение порога $\hat{\alpha}$ зависит только от значений принадлежности μ_{li}, $l = 1, \ldots, c$, $i = 1, \ldots, n$, в матрице $P_{c\times n} = [\mu_{li}]$ нечеткого c-разбиения, и не зависит от классифицируемых объектов и их признаков, формы и других характеристик нечетких кластеров.

Необходимо указать, что в случаях, когда объем исследуемой совокупности $X = \{x_1, \ldots, x_n\}$ достаточно велик, носители $\{A_{\alpha}^1, \ldots, A_{\alpha}^c\}$ α-ядер $\{A^1(\alpha), \ldots, A^c(\alpha)\}$ кластеров нечеткого c-разбиения $P(X) = \{A^1, \ldots, A^c\}$ могут рассматриваться как множества объектов, подлежащие дальнейшей классификации. Данное обстоятельство является основополагающим при последовательном применении методов нечеткой кластеризации к большим массивам данных [6], и выделение α-ядер нечетких кластеров позволяет обрабатывать данные в полностью автоматическом режиме.

Список литературы

1. *Вятченин Д.А.* Нечеткие методы автоматической классификации. – Мн.: УП «Технопринт», 2004. – 219 с.
2. Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition / Höppner F., Klawonn F., Kruse R., Runkler T. – Chichester: Wiley Intersciences, 1999. – 289 p.

3. *Заде Л.А.* Размытые множества и их применение в распознавании образов и кластер-анализе // Классификация и кластер / Под ред. Дж. Вэн Райзина; пер с англ.; под ред. Ю.И. Журавлева. – М: Мир, 1980. – С. 208-247.
4. *Bensaid A.M., Hall L.O., Bezdek J.C., Clarke L.P.* Partially supervised clustering for image segmentation // Pattern Recognition. – 1996. – Vol. 29. – P. 859-871.
5. *Pedrycz W.* Fuzzy sets in pattern recognition: methodology and methods // Pattern Recognition. – 1990. – Vol.23. – P. 121-146.
6. *Viattchenin D.A.* Methodological Aspects of Fuzzy Clustering Application to Data Analysis Problems // Proceedings of the Sixth ISTC Scientific Advisory Committee Seminar (Moscow, Russia, September 15-17, 2003), Vol. I. – Moscow: Russian Academy of Sciences, 2003. – P. 286-293.
7. *Radecki T.* Level fuzzy sets // Journal of Cybernetics. – 1977. – Vol. 7. – P. 189-198.
8. *Кофман А.* Введение в теорию нечетких множеств / Пер. с фр. В.Б. Кузьмина; под ред. С.И. Травкина. – М.: Радио и связь, 1982. – 432 с.
9. *Yang M.-S., Ko C.-H.* On a class of fuzzy c-numbers clustering procedures for fuzzy data // Fuzzy Sets and Systems. – 1996. – Vol. 84. – P. 49-60.

Статья поступила в редакцию 25.04.2008

УДК 004.932

Реализация иерархической модели данных в системе компьютерного планирования хирургических операций в ортопедии

© Гончаренко В.Г., Архипов В.И., Тузиков А.В.

Объединенный институт проблем информатики
НАН Беларуси
ул. Сурганова, 6, г. Минск, 220012, Беларусь

e-mail: *vasily@mpen.bas-net.by, arkhipau@gmail.com, tuzikov@newman.bas-net.by*

Abstract. Hierarchical model describing transformation results of large images is discussed. A procedure of binary image resection by a polygon ("scissor") selected by a user is described. All connected components, completely crossed by the polygon, are resected onto several parts.

Введение

В 1972 году в связи с появлением компьютерной томографии появилась возможность создавать трехмерные изображения внутренних органов и тканей. Эти изображения строятся в виде последовательности двухмерных слоев, составляющих трехмерное объемное изображение. Каждый слой томографического изображения независим от других, при этом ярость каждой точки на томографическом изображении характеризует плотность соответствующей анатомической области, что позволяет по яркости точки с определенной достоверностью классифицировать тип ткани (костная, мышечная, жировая и т.д.). Все это позволяет использовать трехмерные томографические изображения для построения объемных моделей внутренних органов и применять их для планирования хирургических операций при помощи компьютера [1, 2].

Планирование операций выполняется предварительно для вычисления параметров основных процедур хирургической операции в целях подготовки к реальной операции. Объектами операции являются органы и ткани пациента, имеющие выраженную патологию и требующие хирургического вмешательства. Обычно последовательность действий во время хирургической операции определяется хирургом самостоятельно на основании изучения состояния анатомических объектов по результатам предоперационного планирования [3].

Компьютерное планирование выполняется при помощи специальных программных средств путем воспроизведения на компьютере всей последовательности действий, составляющих хирургическую операцию. Компьютерное планирование операции можно назвать виртуальной операцией. Виртуальная операция не несет никакого риска для пациента и, в то же время, позволяет легко определить действия и их параметры, необходимые для проведения реальной операции. Для определения геометрических параметров костей и суставов анализируются цифровые изображения.

Процесс компьютерного планирования хирургических операций в ортопедии обычно состоит из 8 этапов:

1) сканирование пациента с помощью компьютерного томографа;
2) предварительная обработка томографического изображения и построение иерархической модели;
3) сегментация и классификация объектов планирования;
4) оценка визуальной информации;
5) измерение моделей костей;
6) оценка результатов измерений и принятие решений относительно дальнейших действий;
7) виртуальная остеотомия;
8) виртуальный остеосинтез.

Подробное описание этих этапов приводится в работе [3]. Здесь рассматривается понятие иерархической модели, а также задача разрезания объектов иерархической модели в процессе виртуальной остеотомии.

1. Концепция иерархической модели

В целях организации работы с различными типами преобразований отдельных частей изображений была спроектирована так называемая «иерархическая модель». В связи с тем, что различные части изображения могут потребовать различных методов их преобразования, предложено использование одновременно нескольких различных способов выделения объектов на изображении.

Иерархическая модель представляет собой дерево, вершинами которого являются объекты, получившиеся в результате преобразований различных участков исходного изображения. В корне дерева находится исходное изображение. В дереве выделяется один из объектов, который считается текущим – с ним пользователь работает в данный момент. Так, например, сегментация изображения всегда применяется к части текущего объекта, определяемой областью интереса в виде прямоугольного параллелепипеда. Объекты иерархической модели с точки зрения программной реализации представляют собой объекты класса C++ $KPOHierarchicalObject$. Это абстрактный класс, имеющий ряд чистых виртуальных функций, таких как $GetObjectVolume$ и $GetObjectSize$. Такая концепция позволяет реализовать два типа операций: хорошо параметризуемые и плохо параметризуемые. Объекты, являющиеся результатом хорошо параметризуемой операции, можно реконструировать за приемлемое время, используя в качестве параметров процедуры реконструкции некоторые атрибуты, требующие мало памяти по сравнению с самим объектом. Таким образом, объекты, получившиеся в результате хорошо параметризуемой операции, не хранят в явном виде значения вокселов соответствующего им участка. Вместо этого они хранят атрибуты, которые позволяют строить массив вокселов, используя данные их родителей и исходного изображения. Так, например, пороговую сегментацию можно трактовать как хорошо параметризуемую операцию, атрибутами которой являются пороговые уровни. Значения вокселов (бинарного изображения) можно построить из исходного

изображения, используя атрибуты (значения пороговых уровней). Такая схема позволяет освободить память, выполняя построение массива вокселов только тогда, когда он необходим, после чего он удаляется из оперативной памяти. Объекты, являющиеся результатами плохо параметризуемой операции, используются, когда построение массива вокселов объекта нельзя произвести за приемлемое время, или когда атрибуты занимают больше памяти, чем сам результат операции. В качестве примера плохо параметризуемой операции может выступать ручная сегментация.

Каждый объект иерархической модели имеет свои размеры, а также координаты и матрицу ориентации относительно своего объекта-родителя. Это позволяет быстро выполнять перенос и вращения всей ветви дерева иерархической модели.

Объектами иерархической модели могут быть не только результаты сегментации изображения, но и результаты промежуточных преобразований (например, фильтрации). Предложенная концепция хорошо и плохо параметризуемых операций позволяет учесть множество различных преобразований участков изображения.

После загрузки исходного изображения оно помещается в корень иерархической модели. Как только пользователь выполнит сегментацию какой-либо части исходного изображения, отсегментированное изображение помещается в иерархическую модель как потомок корневого элемента (т.е. исходного изображения). Обычно после сегментации выполняется классификация отсегментированных объектов – они помещаются на третий уровень иерархической модели (рисунок 1). Количество классифицированных объектов соответствует количеству уникальных объектов, имеющихся на классифицируемом участке текущего объекта. Пользователь в любое время может выбрать текущий объект и те объекты, которые необходимо отображать (текущий объект всегда является отображаемым объектом). Если отображается какой-либо объект, то в это же время его потомки и предки отображаться не могут. Все действия по выбору объектов пользователь выполняет при помощи менеджера объектов – специального диалогового окна, где отображена древовидная структура текущей иерархической модели.

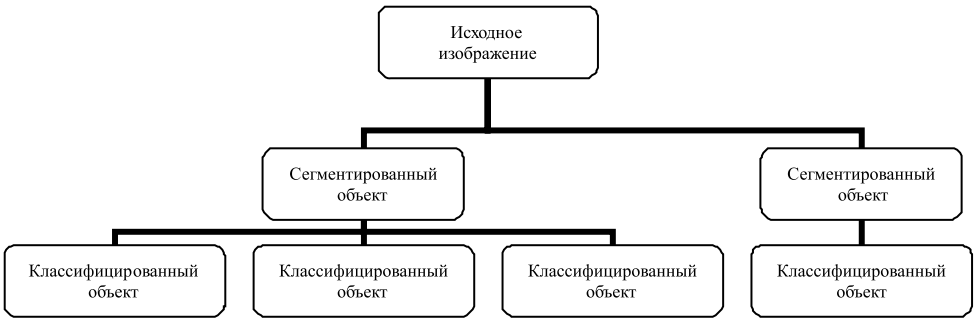
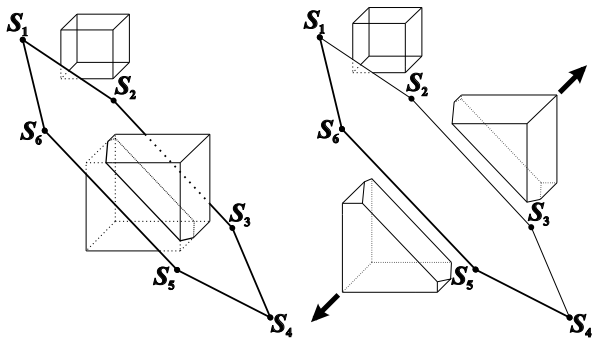


Рис. 1. Примерная структура обычной иерархической модели

2. Выделение связных компонент и разрезание объектов

Одной из важных плохо параметризуемых операций над бинарными объектами иерархической модели в системе компьютерного планирования выступает разрезание объектов. Эта операция имеет большое значение при планировании остеотомии [2].

В процессе выполнения виртуальной операции появляется необходимость разрезать текущий объект на две или несколько частей замкнутой ломаной линией $\rho = (\mathbf{S}_1, \ldots, \mathbf{S}_k)$, образующей k-угольник, все вершины которого лежат в одной плоскости $\Omega(\rho)$. Эта ломаная образует активную часть плоскости разрезания, называемую «резцом» (рисунок 2). Бинарный объект состоит из некоторого множества связных компонент. При использовании резца разрезаются только те связные компоненты, которые полностью пересекаются активной частью плоскости разреза. Сама процедура разрезания использует операцию выделения связных компонент и является, по сути, её модификацией.



(а) (б)

Рис. 2. Разрезание объекта инструментом «резец», в котором активная часть плоскости разрезания задаётся шестиугольником $(S_1, \ldots, S_6)$, и разрезаются только объекты, попадающие в активную часть плоскости разреза: а) вид объектов до разрезания; б) вид объектов после разрезания

Рассмотрим, как определяются связи между соседними вокселами трехмерного изображения. Если инструмент «резец» не задан, либо он не пересекается с текущим объектом, то все вокселы, принадлежащие объекту и являющиеся соседями в используемой метрике, имеют связи между собой (рисунок 3(а)). Как только задается «резец», пересекающий текущий объект, вычисляются точки пересечения активной части плоскости резца ρ и связей между вокселами, образующими объект (см. рисунок 3(в)). «Резец» разрезает только ребра решетки, пересекаемые активной частью плоскости разрезания ρ.

Опишем более подробно процесс построения связей элементов изображения при использовании инструмента «резец». Пусть задано бинарное изображение B, на котором объекты имеют значение 1, а фон – значение 0. Обозначим через $\mathbf{n}$ вектор

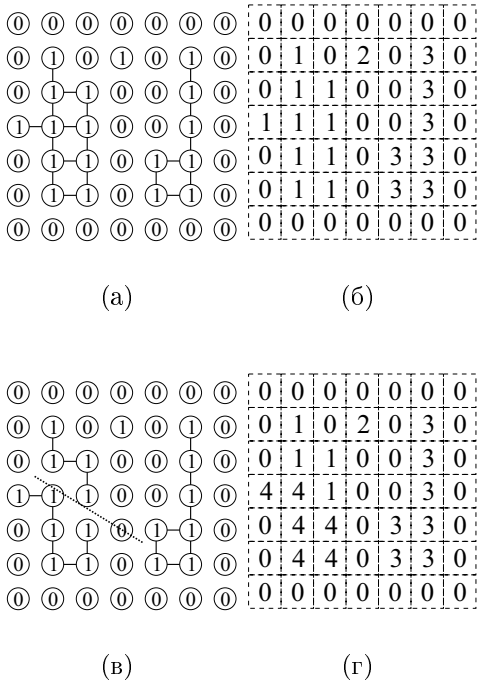


Рис. 3. Пример разрыва связей пикселов связной компоненты инструментом «резец» с использованием четырехсвязной метрики; а) построение связей между пикселами при выделении связных компонент; б) связные компоненты изображения, помеченные соответствующими номерами; в) разрыв связей между пикселами (пунктирная линия); г) связные компоненты, получившиеся при разрезании объекта.

нормали плоскости разрезания. Введем функцию $\delta_\Omega(\mathbf{p})$ положения воксела $\mathbf{p}$ относительно плоскости разрезания $\Omega(\rho)$:

$$\delta_\Omega(\mathbf{p}) = \mathbf{n} \cdot (\mathbf{S}_1 - \mathbf{p}), \tag{1}$$

где «$\cdot$» обозначает скалярное произведение двух векторов.

Если воксел $\mathbf{p}$ принадлежит объекту и у него имеется сосед $\mathbf{q}$, принадлежащий тому же объекту, то можно определить, пересекает ли плоскость $\Omega(\rho)$ связь между $\mathbf{p}$ и $\mathbf{q}$. Связь пересекается, если значение функции δ_Ω для вокселов $\mathbf{p}$ и $\mathbf{q}$ имеет разный знак:

$$\delta_\Omega(\mathbf{p})\delta_\Omega(\mathbf{q}) < 0. \tag{2}$$

Связь между вокселом $\mathbf{p}$ и его соседом $\mathbf{q}$ пересекается «резцом» ρ и разрывается, если она пересекается плоскостью $\Omega(\rho)$, и точка $\mathbf{I}_{\mathbf{p},\mathbf{q}}$ пересечения прямой, проходящей через $\mathbf{p}$ и $\mathbf{q}$, и плоскости разрезания $\Omega(\rho)$ находится внутри многоугольника ρ.

Координаты точки $\mathbf{I}_{\mathbf{p},\mathbf{q}}$ вычисляются по формуле:

$$\mathbf{I}_{\mathbf{p},\mathbf{q}} = \mathbf{p} + (\mathbf{q} - \mathbf{p})\frac{\delta_\Omega(\mathbf{p})}{\mathbf{n} \cdot (\mathbf{q} - \mathbf{p})}. \tag{3}$$

В случае, если знаменатель в (3) равен 0 (прямая, соединяющая $\mathbf{p}$ и $\mathbf{q}$, параллельна плоскости разрезания $\Omega(\rho)$), связь между $\mathbf{p}$ и $\mathbf{q}$ считается неразорванной.

Перейдем к двухмерным координатам и обозначим через $\rho' = (\mathbf{S'}_1, \ldots, \mathbf{S'}_k)$ двухмерный многоугольник, координаты вершин которого определяются следующим образом:

$$\begin{cases} \mathbf{S'}_i^x = \mathbf{S}_i \cdot \mathbf{i}, \\ \mathbf{S'}_i^y = \mathbf{S}_i \cdot \mathbf{j}, \end{cases} \tag{4}$$

где i изменяется от 1 до k, а $\mathbf{i}, \mathbf{j}$ – базисные векторы, определяющиеся следующим образом:

$$\begin{cases} \mathbf{i} = \frac{\mathbf{S}_2 - \mathbf{S}_1}{\|\mathbf{S}_2 - \mathbf{S}_1\|}, \\ \mathbf{j} = \mathbf{i} \times \mathbf{n}. \end{cases}$$

Обозначим через $\mathbf{I'}_{\mathbf{p},\mathbf{q}} = (\mathbf{I'}^x_{\mathbf{p},\mathbf{q}}, \mathbf{I'}^y_{\mathbf{p},\mathbf{q}})$ точку $\mathbf{I}_{\mathbf{p},\mathbf{q}}$ в базисе $(\mathbf{i}, \mathbf{j})$:

$$\begin{cases} \mathbf{I'}^x_{\mathbf{p},\mathbf{q}} = \mathbf{I}_{\mathbf{p},\mathbf{q}} \cdot \mathbf{i}, \\ \mathbf{I'}^y_{\mathbf{p},\mathbf{q}} = \mathbf{I}_{\mathbf{p},\mathbf{q}} \cdot \mathbf{j}. \end{cases}$$

Рассмотрим задачу определения принадлежности точки многоугольнику. Существует несколько различных способов решения этой задачи [4]. В данном случае, когда количество сторон многоугольника невелико, реализовано следующее решение. Вначале определяется некоторый базовый луч, исходящий из точки $\mathbf{I'}_{\mathbf{p},\mathbf{q}}$. Затем определяется количество ребер многоугольника ρ' пересекаемых этим лучом. Если количество пересечений нечетное, то точка $\mathbf{I'}_{\mathbf{p},\mathbf{q}}$ лежит внутри многоугольника ρ' и, следовательно, связь между $\mathbf{p}$ и $\mathbf{q}$ разрывается. Иначе связь между $\mathbf{p}$ и $\mathbf{q}$ сохраняется.

В случае, если многоугольник ρ' является выпуклым, имеется более простое решение данной задачи. Вообще говоря, в большинстве случаев резец вполне можно сделать выпуклым многоугольником, не теряя при этом универсальности его применения и выигрывая в удобстве его использования. Достаточно удобно пользоваться резцом, представляющим собой выпуклый 4-угольник. Точка $\mathbf{I'}_{\mathbf{p},\mathbf{q}}$ лежит внутри выпуклого многоугольника ρ', если точка $\mathbf{I'}_{\mathbf{p},\mathbf{q}}$ находится всегда с одной стороны по отношению ко всем сторонам многоугольника при обходе их в одном направлении. Другими словами, точка $\mathbf{I'}_{\mathbf{p},\mathbf{q}}$ лежит внутри выпуклого многоугольника ρ', если величина $\mu_{\mathbf{p},\mathbf{q}}(i, i+1)$ имеет одинаковый знак для всех $i = 1, 2, \ldots, k$:

$$\mu_{\mathbf{p},\mathbf{q}}(i, i+1) = (\mathbf{I'}^y_{\mathbf{p},\mathbf{q}} - \mathbf{S}_i^{\prime y})(\mathbf{S}_{i+1}^{\prime x} - \mathbf{S}_i^{\prime x}) - (\mathbf{I'}^x_{\mathbf{p},\mathbf{q}} - \mathbf{S}_i^{\prime x})(\mathbf{S}_{i+1}^{\prime y} - \mathbf{S}_i^{\prime y}), \tag{5}$$

здесь $\mu_{\mathbf{p},\mathbf{q}}(k, k+1) = \mu_{\mathbf{p},\mathbf{q}}(k, 1)$.

При реализации данной схемы вычислений необходимо учесть, что для каждого помещаемого в очередь воксела $\mathbf{p}$ целесообразно заранее вычислять и сохранять значение $\delta_\Omega(\mathbf{p})$. Его можно использовать позже для проверки, пересекает ли плоскость $\Omega(\rho)$ связь между выбранным из очереди вокселом $\mathbf{p}'$ и его соседом $\mathbf{q}$. Вычисленное значение $\delta_\Omega(\mathbf{q})$ также желательно сохранять вместе с помещаемым в очередь вокселом $\mathbf{q}$.

Для случая регулярной метрики необходимо принимать во внимание расположение соседних вокселов. Например, при использовании шестисвязной метрики для трехмерного изображения каждый воксел содержит по 6 соседей (кроме вокселов, расположенных на границе изображения). Расстояние между текущим вокселом и его соседями равно 1. Причем, положение каждого соседа отличается от положения текущего воксела только одной координатой. Соответственно, выражения для $\delta_\Omega(\mathbf{q})$ и $\mathbf{I}_{\mathbf{p},\mathbf{q}}$ можно вычислять для каждого соседа в зависимости от его положения (таблица 1). Здесь $\mathbf{p}^x, \mathbf{p}^y, \mathbf{p}^z$ – координаты вектора $\mathbf{p}$ в трехмерном пространстве. Аналогично обозначаются координаты вектора $\mathbf{n}$.

Таблица 1. Оптимизация вычислений в случае шестисвязной метрики на трехмерной регулярной решетке

Координаты соседа $\mathbf{q}$ воксела $\mathbf{p}$	Значение $\delta_\Omega(\mathbf{q})$	Значение $\mathbf{I}_{\mathbf{p},\mathbf{q}}$
$(\mathbf{p}^x - 1, \mathbf{p}^y, \mathbf{p}^z)$	$\delta_\Omega(\mathbf{p}) + \mathbf{n}^x$	$\left(\mathbf{p}^x + \delta_\Omega(\mathbf{p})/\mathbf{n}^x, \mathbf{p}^y, \mathbf{p}^z\right)$
$(\mathbf{p}^x + 1, \mathbf{p}^y, \mathbf{p}^z)$	$\delta_\Omega(\mathbf{p}) - \mathbf{n}^x$	$\left(\mathbf{p}^x + \delta_\Omega(\mathbf{p})/\mathbf{n}^x, \mathbf{p}^y, \mathbf{p}^z\right)$
$(\mathbf{p}^x, \mathbf{p}^y - 1, \mathbf{p}^z)$	$\delta_\Omega(\mathbf{p}) + \mathbf{n}^y$	$\left(\mathbf{p}^x, \mathbf{p}^y + \delta_\Omega(\mathbf{p})/\mathbf{n}^y, \mathbf{p}^z\right)$
$(\mathbf{p}^x, \mathbf{p}^y + 1, \mathbf{p}^z)$	$\delta_\Omega(\mathbf{p}) - \mathbf{n}^y$	$\left(\mathbf{p}^x, \mathbf{p}^y + \delta_\Omega(\mathbf{p})/\mathbf{n}^y, \mathbf{p}^z\right)$
$(\mathbf{p}^x, \mathbf{p}^y, \mathbf{p}^z - 1)$	$\delta_\Omega(\mathbf{p}) + \mathbf{n}^z$	$\left(\mathbf{p}^x, \mathbf{p}^y, \mathbf{p}^z + \delta_\Omega(\mathbf{p})/\mathbf{n}^z\right)$
$(\mathbf{p}^x, \mathbf{p}^y, \mathbf{p}^z + 1)$	$\delta_\Omega(\mathbf{p}) - \mathbf{n}^z$	$\left(\mathbf{p}^x, \mathbf{p}^y, \mathbf{p}^z + \delta_\Omega(\mathbf{p})/\mathbf{n}^z\right)$

Рассмотрим подробнее алгоритм выделения связных компонент и разрезания объектов изображения (алгоритм 1). Он работает для изображения с любой связностью. Пусть каждый воксел имеет флаг состояния просмотра ($TEMP$ – еще не просматривался, $DONE$ – уже просматривался и включен в состав связной компоненты). Вначале инициализируется номер текущей связной компоненты $c \Leftarrow 1$. Затем алгоритм переходит в стадию последовательного просмотра массива бинарных вокселов. Как только обнаружится непросмотренный (имеющий флаг $TEMP$) воксел $\mathbf{p}$, принадлежащий объекту ($val(\mathbf{p}) = 1$), он помещается в хвост очереди при помощи функции $Enqueue(\mathbf{p})$, присоединяется к текущей связной компоненте $ConnectedComponent(\mathbf{p}) \Leftarrow c$ и ему присваивается флаг $DONE$. После этого номер текущей связной компоненты увеличивается на 1 и алгоритм переходит в стадию наращивания связной компоненты, которой принадлежит только что просмотренный воксел $\mathbf{p}$. При этом, из головы очереди снимается один воксел $\mathbf{p}' \Leftarrow Dequeue()$. Затем рассматриваются все его соседи, с которыми он имеет связи. Если воксел $\mathbf{q}$ имеет

флаг $TEMP$, принадлежит объекту и находится с обратной стороны от плоскости разрезания $\Omega(\rho)$ относительно воксела $\mathbf{p}'$, то выполняется проверка на то, попадает ли точка пересечения связи вокселов $\mathbf{p}'$ и $\mathbf{q}$ с плоскостью $\Omega(\rho)$ внутрь многоугольника ρ. Если связь между вокселами $\mathbf{p}'$ и $\mathbf{q}$ пересекается «резцом», то она удаляется. Иначе воксел $\mathbf{q}$ добавляется в хвост очереди командой $Enqueue(\mathbf{q})$, присоединяется к связной компоненте $ConnectedComponent(\mathbf{p}')$ и ему присваивается флаг $DONE$. Такая проверка выполняется для всех соседей воксела $\mathbf{p}'$, с которыми у него есть связи. После того, как очередь опустеет, алгоритм опять переходит в стадию последовательного просмотра вокселов и переходит к следующему за $\mathbf{p}$ вокселу. Процесс прекращается как только будут просмотрены все вокселы объекта. В результате этой процедуры создется изображение, состоящее из вокселов со значениями индексов соответствующих им связных компонент (рисунок 3(б)). Каждая связная компонента с одинаковым значением вокселов будет образовывать отдельный объект.

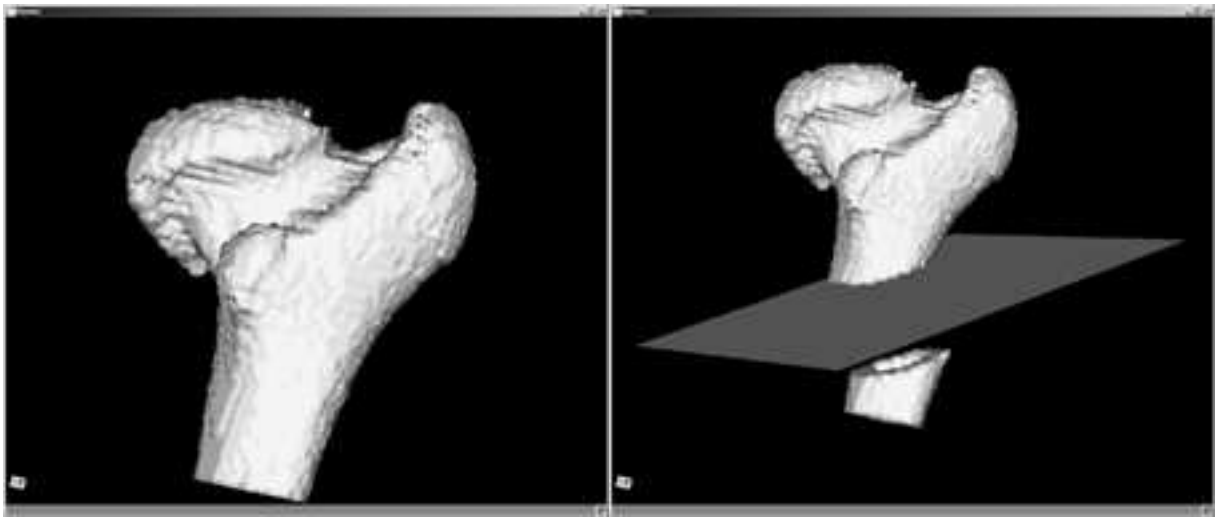
Алгоритм 1 Выделение связных компонент и разрезание бинарных объектов

На входе: Бинарное изображение B – массив вокселов $\mathbf{p} \in B$ с множеством значений $val(\mathbf{p}) \in \{0, 1\}$; множество R связей $rel(\mathbf{p}, \mathbf{q}) \in R$ между вокселами $\mathbf{p}, \mathbf{q} \in B$

На выходе: Бинарное изображение $ConnectedComponent$, состоящее из индексов связных компонент, получившихся после разрезания

1: **for all** $\mathbf{p} \in B$ **do**
2: $flag(\mathbf{p}) \Leftarrow TEMP$;
3: $c \Leftarrow 1$;
{Просмотр вокселов}
4: **for all** $\mathbf{p} \in B$ **do**
5: **if** $flag(\mathbf{p}) = TEMP$ **and** $val(\mathbf{p}) = 1$ **then**
6: $Enqueue(\mathbf{p})$;
7: $ConnectedComponent(\mathbf{p}) \Leftarrow c$;
8: $flag(\mathbf{p}) \Leftarrow DONE$;
9: $c \Leftarrow c + 1$;
{Наращивание связной компоненты}
10: **while** Очередь не пуста **do**
11: $\mathbf{p}' \Leftarrow Dequeue()$;
12: **for all** сосед $\mathbf{q}$ воксела $\mathbf{p}'$ **do**
13: **if** $flag(\mathbf{q}) = TEMP$ **and** $val(\mathbf{q}) = 1$ **then**
14: $connection \Leftarrow TRUE$;
{Проверка на пересечение плоскостью $\Omega(\rho)$ связи $\mathbf{p}'$ с $\mathbf{q}$}
15: **if** $\delta_\Omega(\mathbf{p}')\delta_\Omega(\mathbf{q}) < 0$ **then**
16: **if** знак $\mu_{\mathbf{p}',\mathbf{q}}(i, i+1)$ одинаков для всех $i = 1, \ldots, k$ **then**
17: $connection \Leftarrow FALSE$;
18: **if** $connection$ **then**
19: $Enqueue(\mathbf{q})$;
20: $ConnectedComponent(\mathbf{q}) \Leftarrow ConnectedComponent(\mathbf{p}')$;
21: $flag(\mathbf{q}) \Leftarrow DONE$;

На рисунке 4 показан пример разрезания бедренной кости.



(а) (б)

Рис. 4. Пример работы алгоритма разрезания: а) объект до разрезания; б) объект после разрезания.

Заключение

В статье разработана методика иерархического моделирования, позволяющая эффективно сохранять результаты различных преобразований частей изображения. Все преобразования делятся на два типа: хорошо и плохо параметризуемые. Результаты хорошо параметризуемых операций упаковываются таким образом, что в явном виде хранятся только их параметры, а результат строится «на ходу». Эффективность такой методики основывается на том, что параметры хорошо параметризуемой операции занимают значительно меньший объем памяти, чем ее результаты. Иерархическая модель представляет собой дерево, элементами которого выступают результаты преобразований частей одного изображения. В корне этого дерева находится исходное изображение. Каждое последующее преобразование части изображения создает ветвь дерева. Это позволяет связывать все операции, выполняемые над данными одного изображения.

Разработан алгоритм выделения связных компонент и последующего разрезания бинарных объектов специальным инструментом «резец». Этот инструмент представляет собой многоугольник, все точки которого лежат в одной плоскости, называемый активной частью плоскости разрезания. Разрезаются только те связные компоненты

объекта, которые пересекаются активной частью плоскости разрезания. Предложенный инструмент «резец» позволяет выполнять разрезание бинарных объектов любой формы, включая отдельные кости таза.

Список литературы

1. *Handels H., Ehrhardt J., Plötz W., Pöppl S.J.* Three-dimensional planning and simulation of hip operations and computer-assisted construction of endoprostheses in bone tumor surgery. Computer Aided Surgery, vol. 6, Nu. 2, p. 65–76, 2007.
2. *Krivonos O. Hesser J. Maenner R. Keppler P. Gebhard F. Kinzl L. Sakalouski A.A. Sakalouski A.M.* Precise computer aided correction of bone deformities. Proceedings of the World Congress on Medical Physics and Biomedical Engineering WC2003, Sydney, Australia, August 2003.
3. *Hancharenka V.G., Tuzikov A.V. Arkhipau V.I., Kryvanos A.* Preoperative planning of pelvic and lower limbs surgery by CT image processing. Proceedings of 8th International Conference on Pattern Recognition and Image Analysis PRIA'2007, 7-13 Oct., Yoshkar-Ola, Russia, vol. 1, p. 270–274, 2007.
4. *Haines E.* Point in Polygon Strategies. Graphics Gems IV, ed. Paul Heckbert, Academic Press, p. 24–46, 1994.

Статья поступила в редакцию 19.04.2008

УДК 004.622

ПИТАННЯ ОЧИСТКИ ДАНИХ ПРИ СТВОРЕННІ АВТОМАТИЗОВАНИХ СИСТЕМ НОРМАТИВНО-ДОВІДКОВОЇ ІНФОРМАЦІЇ

© Гула О.Ю.

ТОВ «Ер-Джі-Дейта»
Україна 03056, Київ, вул. Політехнічна, 33, к. 616
Тел.: +380 44 241 9131; факс: +380 44 236 3188

e-mail: *alexg@rgdata.com.ua*

The article describes methods of data identification in referenced data systems/ Algorithm of hierarchical classifiers data consolidation is suggested. Structure of data processing application for unified hierarchical classifier construction is proposed.

Вступ

Нормативно-довідкова інформація (НДІ) в автоматизованих системах призначена для групування та систематизації інформації про поняття, об'єкти, явища, тощо в стандартний формат, що допомагає визначити їх подібність. Систематизація інформації здійснюється шляхом її класифікації, а саме поділом множини об'єктів на підмножини, зі застосуванням прийнятих методів визначення подібності на підставі їх схожості чи несхожості [1]. При ієрархічній системі множина об'єктів ділиться на великі групи, потім кожна група ділиться на ряд множину підгруп, які в свою чергу також можуть ділитися, поступово конкретизуючи об'єкт.

В даній роботі розглядається задача побудови зведеного ієрархічного реєстру товарно-матеріальних цінностей (ТМЦ) корпорації, що об'єднує декілька окремих класифікаторів ТМЦ філій корпорації в єдину централізовану систему НДІ для подальшої обробки експертами.

При створенні такої централізованої системи НДІ значне місце займають питання консолідації, очистки даних та їх узгодження [2, 3], зокрема:

1. Усунення дублювання як окремих об'єктів, так і цілих груп об'єктів.
2. Усунення помилок та розбіжностей в значеннях атрибутів об'єктів (наприклад, помилки в написанні однакових пунктів, різні правила заповнення, різний порядок слів в назві, наявність чи відсутність додаткових кодів, марок та одиниць виміру ТМЦ).
3. Виправлення помилок та розбіжностей структуризації класифікатора, а саме наявності чи відсутності різних груп чи об'єктів.

1. Математична модель

Дамо деякі визначення.

$\overline{E} = [E_i]_{i=1}^{n_e}, n_e = \left\|\overline{E}\right\|$ — множина об'єктів E класифікатора, де n_e — кількість об'єктів.

$\overline{K} = \left[\overline{E}_i\right]_{i=1}^{n_k}, n_k = \left\|\overline{K}\right\|$ — ієрархічний класифікатор, заданий як множина множин об'єктів.

$\overline{E}^0 = \bigcup_{i=1}^{n_k} \overline{E}_i$ — множина нульового рівня класифікації, представляє собою множину всіх об'єктів класифікатора.

$\forall \overline{E}^l, l > 0\ \exists! \overline{E}^{l-1} : \overline{E}^l \subseteq \overline{E}^{l-1}$ — множина l рівня класифікації, вкладена рівно в одну множину $l-1$ рівня.

За визначенням класифікатора, об'єкт може бути вкладений тільки в одну множину на кожному з рівнів, причому множини повинні бути послідовно вкладеними. Тому можна побудувати множину всіх послідовно вкладених множин, що містять об'єкт:

$$\overline{B}\left(E^m\right) = \left[\overline{E}^l\right]_{l=1}^{m}, \forall l \in [1;m]\ \exists \overline{E}^{l-1} : \overline{E}^l \subseteq \overline{E}^{l-1}, E^m \in \overline{E}^l.$$

Тут m — рівень об'єкта, це рівень найменшої множини, що містить заданий об'єкт E^m. Якщо розглядати класифікатор як дерево, то $\overline{B}(E)$ — це гілка дерева, листовим елементом якої є заданий об'єкт E.

1.1. **Постановка задачі.** Нехай задана множина ієрархічних класифікаторів:

$$\overline{F} = \left[\overline{K}_i\right]_{i=1}^{n_F}, n_F = \left\|\overline{F}\right\|,$$

причому множини об'єктів класифікаторів можуть перетинатися

$$\overline{I} = \bigcup_{j=1,k=1}^{n_F,n_F} \overline{E}_j^0 \bigcap \overline{E}_k^0, \overline{E}_j^0 \in \overline{K}_j, \overline{E}_k^0 \in \overline{K}_k, \left\|\overline{I}\right\| \geq 0.$$

Задача полягає у побудові зведеного класифікатору $\overline{K}^F = \bigcup_{k=1}^{n_F} \overline{E}_k^0$, що містить елементи всіх заданих класифікаторів. Головною задачею, що необхідно вирішити, є зменшення дублювання об'єктів — $\left\|\overline{I}\right\| \to 0$.

1.2. **Нечітка відповідність між множинами.** Нехай задана попарна нечітка відповідність між об'єктами класифікаторів $s\left(E_i, E_j\right) \in [0;1]$.

Тоді відповідність між об'єктом та множиною є максимальна відповідність між об'єктом E_i та об'єктами множини $\overline{E}_j$:

$$d\left(E_i, \overline{E}_j\right) = \max_{k=1}^{n}\left(s\left(E_i, E_j\right)\right), n = \left\|\overline{E}_j\right\|.$$

Звідси можна визначити сумарну відповідність між множинами як суму відповідностей об'єктів множини $\overline{E}_i$ до множини $\overline{E}_j$:

$$\hat{d}\left(\overline{E}_i, \overline{E}_j\right) = \sum_{k=1}^{n_i} d\left(E_k, \overline{E}_j\right), E_k \in \overline{E}_i, n_i = \left\|\overline{E}_i\right\|.$$

Визначимо односторонню нечітку відповідність між множинами $\overline{E}_i$ та $\overline{E}_j$ як суму відповідностей об'єктів множини $\overline{E}_i$ до множини $\overline{E}_j$, поділена на кількість елементів множини $\overline{E}_i$:

$$l\left(\overline{E}_i, \overline{E}_j\right) = \frac{\hat{d}\left(\overline{E}_i, \overline{E}_j\right)}{\left\|\overline{E}_i\right\|}.$$

Нехай задана множина множин $M = \left[\overline{E}_i\right]_{i=1}^{m}, m \geq 2$. Дамо наступні визначення.

$\tilde{d}\left(\overline{E}_i, M\right) = \sum\limits_{j=1, j\neq i}^{m} \hat{d}\left(\overline{E}_i, \overline{E}_j\right)$ — сумарна відповідність множини $\overline{E}_i$ до всіх інших множин з M.

$\tilde{n}\left(\overline{E}_i, M\right) = \sum\limits_{j=1, j\neq i}^{m} \left\|\overline{E}_j\right\|$ — сумарна кількість об'єктів всіх множин з M, крім множини $\overline{E}_i$.

Тоді визначимо групову нечітку відповідність між множинами з множини M як суму парних відповідностей між об'єктами кожної з множин множини M, поділену на суму кількості елементів кожної з множин множини M :

$$g(M) = \frac{\sum\limits_{i=1}^{m} \tilde{d}\left(\overline{E}_i, M\right)}{\sum\limits_{i=1}^{m} \tilde{n}\left(\overline{E}_i, M\right)}, m = \left\|\overline{M}\right\|, \overline{E}_i \in M.$$

Нехай $p \in [0;1]$ - заданий поріг. Задамо наступні предикати попарної відповідності множин:

1. $P_I\left(p, \overline{E}_i, \overline{E}_j\right) : l\left(\overline{E}_i, \overline{E}_j\right) \geq p$ — вкладеність множини $\overline{E}_i$ в множину $\overline{E}_j$ зі ступенем $l\left(\overline{E}_i, \overline{E}_j\right)$.
2. $P_L\left(p, \overline{E}_i, \overline{E}_j\right) : P_I\left(p, \overline{E}_i, \overline{E}_j\right) \wedge \neg P_I\left(p, \overline{E}_j, \overline{E}_i\right)$ — точна вкладеність множини $\overline{E}_i$ в множину $\overline{E}_j$ зі ступенем $l\left(\overline{E}_i, \overline{E}_j\right)$.
3. $P_G(p, M) : g(M) \geq p$ — групова відповідність між множинами множини M зі ступенем $r(M)$.
4. $P_R(p, M) : \bigwedge\limits_{i=1}^{m} \bigwedge\limits_{j=1, j\neq i}^{m} P_I\left(p, \overline{E}_i, \overline{E}_j\right)$ — точна групова відповідність між множинами множини M.

Легко бачити, що $P_R(p, M) \Rightarrow P_G(p, M)$, але обернене невірно, звідси може виникнути ситуація, коли для пари множин одночасно виконуються предикати P_L та P_G. Тому для виявлення множин для об'єднання необхідно використовувати більш сильний предикат P_R.

1.3. **Об'єднання множин.** Розглянемо задачу об'єднання множин l рівня класифікації, де $l > 0$.

Визначимо оператор, що для заданої множини $\overline{E}^{l-1}$ знаходить всі множини l рівня класифікації, для яких виконується предикат точної групової відповідності множин та які вкладені в задану множину:

$$F_M^U\left(\overline{E}^{l-1}\right) = \left[\overline{M}_i^l\right]_{i=1}^{m}, \overline{M}_i^l = \left[\overline{E}_j^l\right], \overline{E}_j^l \in \overline{E}^{l-1}, P_R\left(p, \overline{M}_i^l\right),$$

причому він вибирає ці множини так, щоб вони не перетиналися:

$$\forall i \in [1;m] \, \forall j \in [1;m], j \neq i : \overline{M}_i^l \bigcap \overline{M}_j^l = \emptyset.$$

Також визначимо оператор об'єднання множин, який для множини $\overline{M}_i^l$ будує нову множину l рівня класифікації шляхом об'єднання множин, з яких вона складається:

$$F_U\left(\overline{M}_i^l\right) = \bigcup_{j=1}^{m_i} \overline{E}_j^l = \bigcup_{j=1}^{m_i} \left[\overline{E}_k^{l+1}\right]_{k=1}^{n_j}.$$

1.4. **Вкладення множин.** Розглянемо задачу вкладення множин l рівня класифікації, де $l > 0$.

Визначимо оператор, що для заданої множини $\overline{E}^{l-1}$ знаходить всі пари множин $\overline{E_a}^l$ та $\overline{E_b}^l$ l рівня класифікації, для яких виконується оператор точної вкладеності множини та які вкладені в задану множину:

$$F_M^I\left(\overline{E}^{l-1}\right) = \left[\left(\overline{E_{a_i}}^l, \overline{E_{b_i}}^l\right)\right]_{i=1}^{m}, \overline{E_{a_i}}^l \in \overline{E}^{l-1}, \overline{E_{b_i}}^l \in \overline{E}^{l-1}, P_L\left(p, \overline{E_{a_i}}^l, \overline{E_{b_i}}^l\right),$$

причому він вибирає ці пари множин так, щоб кожна множина зустрічалася в результуючій множині тільки один раз:

$$\bigcap_{i=1}^{m_l} \overline{E_{a_i}}^l \cap \bigcap_{i=1}^{m_l} \overline{E_{b_i}}^l = \emptyset.$$

Також задамо оператор об'єднання даних множин, що проводить обробку вкладення шляхом або об'єднання двох множин в одну або вкладення однієї множини в іншу, в залежності від предметної області і конкретної пари множин:

$$F_I\left(\overline{E_a}^l, \overline{E_b}^l\right) = \overline{E}^{l\prime} = \begin{cases} \left[\overline{E_{a_k}}^{l+1}\right] \bigcup \left[\overline{E_{b_k}}^{l+1}\right], \\ \left[\overline{E_a}^l\right] \bigcup \left[\overline{E_{b_k}}^{l+1}\right]. \end{cases}$$

1.5. **Алгоритм побудови зведеного класифікатора.** Таким чином, можна навести наступний алгоритм побудови зведеного класифікатора:

1. Побудувати об'єднану множину нульового рівня класифікації, що містить об'єднання множин першого рівня всіх класифікаторів:
$$\overline{E}^{0\prime} = \bigcup_{k=1}^{n_F} \left[\overline{E}^1\right]_k.$$
2. Провести об'єднання множин, для кожної множини $\overline{E}^{l-1}$, починаючи з $\overline{E}^0$:
   a) Застосувати оператор пошуку множин, що необхідно об'єднати, до множини $\overline{E}^{l-1}$:
   $$\left[\overline{M}^l\right] = F_M^U\left(\overline{E}^{l-1}\right).$$
   b) Застосувати оператор побудови об'єднаних множин F_U до кожної отриманої множини:
   $$\overline{E}^{l\prime} = F_U\left(\overline{M}^l\right).$$
   c) Провести об'єднання множин для кожної отриманої множини $\overline{E}^{l\prime}$.

3. Провести обробку вкладення множин, для кожної множини $\overline{E}^{l-1'}, l > 0$:
   a) Застосувати оператор пошуку множин, для яких необхідно провести обробку вкладення, до множини $\overline{E}^{l-1'}$:
   $$\left[\left(\overline{Ea}^{l'}, \overline{Eb}^{l'}\right)\right] = F_M^I\left(\overline{E}^{l-1'}\right).$$
   b) Застосувати оператор побудови об'єднаних множин F_I до кожної отриманої множини:
   $$\overline{E}^{l''} = F_I\left(\overline{Ea}^{l'}, \overline{Eb}^{l'}\right).$$
   c) Провести обробку вкладення кожної отриманої множини $\overline{E}^{l''}$.

2. Реалізація побудови зведеного ієрархічного реєстру

Пропонується консолідацію, очистку та узгодження даних при створенні централізованої системи НДІ здійснювати за наступною схемою:

1. Консолідація даних з окремих класифікаторів ТМЦ філій корпорації — уточнення структури та збір даних до єдиного місця зберігання; завантаження даних з баз даних класифікаторів підрозділів до об'єднаної БД.
2. Стандартизація даних — приведення значень атрибутів до узгодженого формату зберігання, форматування текстових даних: переведення в верхній регістр, видалення зайвих пробілів та символів - «паразитів».
3. Ідентифікація записів — знаходження нечіткої подібності між записами всередині одного класифікатору та між різними класифікаторами, використовуючи відстань Левенштейна та метод Q-грам [4, 5]. Головною характеристикою запису при знаходженні ступеню подібності до інших записів є його назва та набір текстових атрибутів.
4. Ідентифікація груп — знаходження нечіткої подібності між різними групами записів класифікаторів, використовуючи ступінь подібності між записами груп. Так як одні й ті ж групи в процесі побудови класифікаторів в різних філіях корпорації могли мати різні назви, то головною характеристикою групи при ступеню подібності до інших груп є не її назва, а множина всіх вкладених підгруп та записів.
5. Об'єднання груп з різних класифікаторів, що мають ступінь подібності більше заданого порогу. Якщо групи визначені, як подібні, то множини об'єктів цих груп об'єднуються в одну групу, якщо множина елементів однієї групи визначена як вкладена в множину елементів іншої групи, то перша група переноситься в другу групу як підлеглий елемент, разом зі всіма своїми елементами.
6. Оптимізація даних. Після об'єднання даних проводиться аналіз класифікатора на помилки та неоптимальну організацію даних в ієрархії. Даний етап проводиться за участю експерта.

2.1. **Консолідація даних.** Консолідація даних включає в себе уніфікацію структури даних — завантаження даних з різних джерел в БД єдиної структури.

Для завантаження даних з різних джерел необхідно визначити структуру збереження класифікаторів для БД кожної філії. Для цього вирішуються наступні задачі:

1. Визначення переліку всіх множин та об'єктів кожної з БД філій.
2. Визначення переліку атрибутів множини та об'єкту кожної з БД філій.
3. Визначення значень атрибутів кожної множини та атрибутів кожного об'єкту.
4. Однозначне віднесення кожної множини та кожного об'єкту до тієї чи іншої множини, в рамках даних БД однієї філії.
5. Визначення переліку атрибутів множини та об'єкту для об'єднаного класифікатора.
6. Визначення відображення атрибутів множини та об'єкту кожної з БД філій до атрибутів множини та об'єкту об'єднаного класифікатора.

Дані задачі можливо автоматизувати, застосовуючи систему правил для кожної БД філії. У випадку, коли неможливо провести автоматичну обробку, необхідна участь експерта у вирішенні задачі.

З точки зору математичної моделі даний етап є реалізацією першого етапу алгоритму — побудови об'єднаної множини нульового рівня класифікації.

2.2. **Стандартизація даних.** Стандартизація даних включає в себе виділення елементів атрибутів та уніфікацію — приведення представлення атрибутів об'єктів до єдиного формату.

Елементи атрибутів об'єктів можуть містити помилки, але за можливістю нечіткої обробки можна виділити наступні класи елементів:

1. Нечутливі до змін елементи — елементи, до яких можна застосувати нечітку ідентифікацію. До них відносяться терміни, що описують об'єкт, наприклад, «Металлорукав», «Полоса медная», «Лист свинц.», «діаметр».
2. Чутливі до змін елементи — елементи, до яких не можна застосовувати нечітку ідентифікацію внаслідок того, що зміна будь-якого символу призводить до повної зміни значення. До них відносяться:
   a) Спеціальні ідентифікатори – скорочення, абревіатури, коди, одиниці виміру, наприклад, «мм», «РП-Ц-А-75», «ГОСТ».
   b) Числові ідентифікатори – розміри, діаметри тощо.

Для виділення елементів атрибутів об'єктів застосовуються системи правил. Так як дані в різних БД філій можуть сильно відрізнятися за структурою та складом атрибутів, то застосовуються різні системи правил для даних, що були завантажені з різних БД філій.

Після виділення типів елементів проводиться уніфікація даних, що полягає у перекодуванні символів рядків (наприклад, переведення в верхній регістр, перекодування символів) та елементів атрибутів (наприклад, з використанням таблиць відповідності).

Результатом даного етапу є множина об'єктів класифікатора, приведена до єдиного стандарту.

2.3. **Ідентифікація записів.** Ідентифікація записів полягає в знаходженні відповідності між записами про об'єкти класифікатора в БД. Для цього для кожного запису в БД знаходяться записи, що точно співпадають з даним записом та ті, що мають ступінь подібності більше заданого порогу.

Якщо розглядати об'єкти як множини елементів атрибутів, то ступінь відповідності між об'єктами класифікатора E_i та E_j можна визначити як групову нечітку відповідність множини множин $[E_i, E_j]$:

$$s\left(E_i, E_j\right) = g\left(\left[E_i, E_j\right]\right).$$

Ідентифікація об'єктів прямим перебором має велику складність, тому використовується підготовка елементів — нечітка ідентифікація елементів та збереження інформації про результати ідентифікації в БД. Для нечутливих до змін елементів ступінь відповідності між елементами може бути побудована на основі відстані Левенштейна $d_l\left(A_i^M, A_j^M\right)$ чи на основі різниці Q-грам, зокрема, на основі біграм $d_q\left(A_i^M, A_j^M\right), q = 2$.

Нехай задані пороги відповідності $p^A \in [0;1]$ та $p^E \in [0;1]$.

Для підготовки елементи зберігаються в БД та прив'язується до об'єкту, до якого вони відносяться. Якщо елемент вже існує в БД, він повторно не додається, а тільки прив'язується до об'єкту. Крім того, між елементами, для яких ступінь відповідності більше заданого порогу p^A, зберігаються парні зв'язки із зазначенням ступеню відповідності.

Після підготовки проводиться ідентифікація об'єктів, що полягає в знаходженні та збереженні об'єктів, що подібні до заданого. Для кожного заданого об'єкту E_i виконуються наступні операції:

1. Пошук об'єктів — проводиться пошук записів, що можуть бути подібними до заданого. Проводиться з використанням підготованих елементів:
   a) Вибираються всі елементи, прив'язані до заданого об'єкту.
   b) Вибираються всі об'єкти, до яких прив'язані відібрані елементи.
2. Ідентифікація об'єктів — проводиться розрахунок групової нечіткої відповідності для кожної пари заданого об'єкту і відібраного об'єкту E_j.
3. Збереження результатів ідентифікації — для кожного відібраного об'єкту, для якого виконується предикат точної групової відповідності із заданим об'єктом $P_R\left(p^E, [E_i, E_j]\right)$, зберігається зв'язок між цим об'єктом та заданим об'єктом, із зазначенням ступеню відповідності.

Таким чином, результатом даного етапу є збережена інформація про парну нечітку відповідність між об'єктами класифікатора.

2.4. **Ідентифікація груп.** Ідентифікація груп полягає у знаходженні та збереженні односторонніх нечітких відповідностей між множинами класифікатору із заданим порогом $p^M \in [0;1]$.

Для кожної множини $\overline{E}_i^l$ виконуються наступні операції:

1. Пошук множин — для зменшення алгоритмічної складності проводиться попередній пошук множин, що можуть бути подібними до заданої:

a) Вибираються всі об'єкти заданої множини.
b) Вибираються всі об'єкти, прив'язані до вибраних на попередньому кроці
c) З множини $\overline{E}^{l-1}$, в яку вкладена множина $\overline{E}_i^l$, вибираються множини $\left[\overline{E}_j^l\right]_{j=1}^n$, до яких прив'язані об'єкти, вибрані на попередньому кроці:

$$\overline{E}_i^l \in \overline{E}^{l-1} \wedge \forall j \in [1;n] : \overline{E}_j^l \in \overline{E}^{l-1}.$$

2. Ідентифікація множин — для кожної відібраної множини $\overline{E}_j$ обраховується одностороння нечітка відповідність $l\left(\overline{E}_j, \overline{E}_i\right)$.
3. Збереження множин — для кожної відібраної множини, для якої виконується предикат вкладеності множин $P_I\left(p^M, \overline{E}_i, \overline{E}_j\right)$, зберігається зв'язок між цією множиною та заданою множиною, із зазначенням ступеню відповідності.

Таким чином, результатом даного етапу є збережена інформація про вкладеність множин класифікатора.

2.5. **Об'єднання груп.** Об'єднання груп полягає в побудові множин, для яких виконується предикат точної групової відповідності та об'єднання цих множин.

Для цього задається рівень $l = 1$ та вибирається множина верхнього рівня $\overline{E}^{l-1} = \overline{E}^0$, для якої виконуються наступні операції:

1. Побудова множини $M = \left[\overline{E}_i^l\right]_{i=1}^m$:
   a) Вибрана множина $\overline{E}_k^l$ утворює множину M.
   b) До множини M додається множина $\overline{E}_m^l$, що належить до множини $\overline{E}^{l-1}$ та має прямі та зворотні зв'язки зі всіма множинами з M, тобто виконується предикат точної групової відповідності $P_R\left(p^M, M \bigcup \left[\overline{E}_m^l\right]\right)$. Якщо таких множин декілька, то вибирається та, у якої сума ступенів відповідності всіх зв'язків максимальна.
   c) Операція 1b повторюється до тих пір, поки знаходяться множини $\overline{E}_m^l$.

2. Об'єднання множин $\overline{E}_i^l$ до множини $\overline{E}_r^l = \bigcup_{i=1}^m \overline{E}_i^l$:
   a) Створюється множина $\overline{E}_r^l$. Атрибути множини $\overline{E}_r^l$ формуються на основі атрибутів множин $\overline{E}_i^l$. Зокрема, проводиться конкатенація назв множин, якщо вони відрізняються в різних множин.
   b) Всі множини та об'єкти, що позначені як вкладені в $\overline{E}_i^l$, позначаються як вкладені в $\overline{E}_r^l$.
   c) Всі множини $\overline{E}_i^l$ видаляються.

3. Повторюються операції 1 та 2 до тих пір, поки на заданому рівні є множини, для яких є пов'язані множини.

4. Рекурсивно виконуються всі операції для всіх підмножин множини $\overline{E}^{l-1}$.

З точки зору математичної моделі даний етап є реалізацією другого етапу алгоритму – об'єднання множин, починаючи з верхнього рівня.

2.6. **Оптимізація даних.** Після проведення зведення даних в єдину БД необхідно провести аналіз дерева на повтори та неоптимальну організацію даних за участю експертів.

Оптимізація даних включає наступні операції:

1. Ідентифікація записів про об'єкти класифікатора. Полягає в об'єднанні записів про об'єкти класифікатора, що визнані записами про один об'єкт. Для цього експерту надається для обробки перелік пар пов'язаних об'єктів та надається можливість об'єднати об'єкти або відмінити об'єднання.
2. Обробка атрибутів множин. Після об'єднання декількох множин в одну атрибути об'єднаної множини формуються автоматично, але отримані атрибути потребують ручної обробки. Для цього експерту надається перелік об'єднаних множин та надається можливість ручного редагування атрибутів обраної множини.
3. Обробка вкладення множин. Для пар множин, для яких виконується предикат вкладеності множини, проводиться вкладення або об'єднання множин залежить від множин та предметної області. В разі об'єднання множин для об'єднаної множини проводиться ручна обробка її атрибутів.

Таким чином, система надає експерту перелік варіантів для прийняття рішення щодо кожного з пунктів та забезпечує виконання прийнятих рішень. Після проведення даних операцій можна продовжувати ведення елементів класифікатора, що включає редагування атрибутів обраного об'єкту чи групи, та переміщення об'єкту чи групи до іншої групи.

Висновки

В статті запропоновано алгоритм побудови зведеного ієрархічного реєстру ТМЦ. Особливістю даного алгоритму є те, що кожна група класифікатора розглядається як множина об'єктів, які відносяться до групи. Це дозволяє відкинути розбіжності в значеннях атрибутів групи, таких, як назва групи, при завантаженні даних із різних класифікаторів.

Даний алгоритм було реалізовано для побудови єдиного класифікатора ТМЦ корпорації, сформованого на основі класифікаторів чотирьох філій. Середній об'єм класифікаторів філій – 100000 об'єктів. Класифікатори ТМЦ філій мали схожу структуру класифікації, але в процесі незалежного використання в кожній з філій в структурі класифікаторів з'явилися розбіжності. В результаті проведеного аналізу були вибрані наступні пороги $p^A = 0.75$, $p^E = 0.75$, $p^M = 0.45$, що дозволило об'єднати близько 68% груп класифікаторів.

Одним з напрямків подальшого розвитку роботи є розробка рекомендацій щодо підбору порогів нечіткої відповідності для елементів атрибутів об'єктів, об'єктів та множин.

Список литературы

1. *ДСТУ 1.10:2005* Національна стандартизація. Правила розроблення, побудови, викладання, оформлення, ведення національних класифікаторів.
2. *Rahm E, Do H.H.* Data Cleaning: Problems and Current Approaches // IEEE Techn. Bulletin on Data Engineering, Dec. – 2000.
3. *Kimball R., Caserta J.* The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming and Delivering Data. Wiley 2004.
4. *Орлов Д.* Подсистема сопоставления записей в хранилище данных. http://www.olap.ru/basic/CompareLog_dw.asp.
5. *Гула А.Ю., Игнатенко А.П., Перечинский И.А.* Применение методов интеллектуальной обработки в задачах очистки хранилища данных // Сб. трудов конф. Системы поддержки принятия решений. Теория и практика. – Киев, 2007, С. 145 – 148.

Статья поступила в редакцию 30.04.2008

УДК 519.68

УСТОЙЧИВОСТЬ СКЕЛЕТНОЙ СЕГМЕНТАЦИИ[1]

© Домахина Л.Г.

Московский государственный университет им. М.В. Ломоносова,
факультет вычислительной математики и кибернетики
Воробьевы горы, 1, г. Москва, 119192, Россия

e-mail: *Ludmila.domakhina@gmail.com*

Abstract. There is enough of works in which methods of figures segmentation are described, however frequently they do not contain a serious substantiation in favour of a choice of this or that method of segmentation. In the present work the formalized definition of quality of segmentation through concept stability and a substantiation of stability of skeletal segmentation method is resulted.

Введение

Проблема сегментации (декомпозиции) фигуры в общем виде — это проблема разбиения фигуры на конечное множество компонент. Возможность осуществления различного рода разбиений открывает большие перспективы для решения *важных научных и практических задач*. К ним относятся задачи распознавания, машинного зрения, морфинга (преобразования формы), обработка видео, сжатие изображений и видео и другие. Понятно, что для каждого приложения может быть свой критерий качества сегментации. Посмотрим, как оценивается качество декомпозиций в работах различных авторов.

Работы, посвященные разбиению внутренней области фигуры (сегментации), можно разделить на два класса.

1. так или иначе связанные с различными видами скелета [1, 2, 3, 5, 6]
2. не использующие скелет и его свойства при построении декомпозиции фигуры [7, 8, 9, 10, 11]

В работе [1] описан метод сегментации многоугольной фигуры на «значимые части» на основе прямолинейного скелета. При этом «значимость частей» отмечена чисто визуально, а значит, для двух похожих фигур необязательно выделятся одни и те же «значимые части». Кроме того в работе [1] не показано приложений, в которых может использоваться данное разбиение.

В [2] описано представление формы, основанное на «деревьях симметричных осей». Следующие свойства считаются критерием качества разбиения:

1. *полнота* – сегментация содержит всю информацию об объекте;
2. *компактность* – сегментация содержит, немного частей, но информативных;
3. *робастность* – устойчивость к шумам и незначительным изменениям границы, а также а «артикуляции» – движению частей объекта (человек имеет одинаковую сегментацию в разных положениях движения);
4. *высокая алгоритмическая скорость вычислимости* (полиномиальная).

[1]Работа осуществлена при поддержке РФФИ, проект 08-01-00670

Если полнота метода [2] и высокая вычислительная скорость обоснована, то компактность и робастность показаны на примерах.

Работа [3] представляет иерархическое разбиение, основанное на осевом графе фигуры (дискретном скелете). Обоснованием выбора метода является тот факт, что осевой граф отражает геометрические свойства фигуры, а иерархическое представление фигуры может быть использовано для задач определения коллизий [4].

Интересный метод декомпозиции фигуры предложен в [5]. Разбиение строится так же как и в [3] иерархически, но во время параллельного итерационного процесса построения «приблизительного скелета», отражающего топологию фигуры. На каждой итерации с использованием некоторого критерия оценивается точность построенного скелета и качество декомпозиции. Устойчивость к шумам и деформациям показана на экспериментах.

Авторами настоящей статьи был предложен метод скелетной сегментации растровых объектов [6]. В основе его лежит построение непрерывного скелета фигуры, аппроксимирующей растровое изображение и выделение непересекающихся областей, относящихся к ребрам скелета. Их объединение – скелетная сегментация.

В качестве методов сегментации, не связанных с построением скелета, можно выделить морфологические подходы [7, 8], в которых происходит разбиение фигуры с использованием заданного структурного элемента путем морфологических операций (эрозия, открытие, закрытие). В работе [8] – пример «эффективной и точной» сегментации формы. Основной целью предложенного авторами [8] разбиения фигур было предотвращение «перекрытий» при морфологическом разбиении фигуры. Также авторы [8] указывают на то, что их метод дает значительно меньше компонент фигуры в разбиении по сравнению с некоторыми другими методами. В работе [9] основной задачей является разбиение фигуры на так называемую «основную форму» и «дополнительные отклонения» от нее. Для решения этой задачи используются дескрипторы Фурье, которые дают базовые характеристики формы такие как вытянутость, эллиптические и циркулярные характеристики. Внутренние свойства фигуры не рассматриваются. В [10] приведен пример разбиения на основе анализа выпуклости фигуры. Единственным его достоинством является простота. О стабильности таких подходов нет речи. Аналогичный подход предложен в [11]. В этой работе авторы говорят о «визуальном качестве» их подхода, так как выпуклые части фигуры визуально выделяются и должны быть отнесены к различным областям сегментации. Шумы на границе предлагается устранить с помощью выбора подходящей аппроксимирующей фигуры.

Можно сделать вывод о том, что большинство из приведенных работ не содержит серьезного обоснования в пользу выбора того или иного метода сегментации. Авторы ограничиваются общими словами о визуальном качестве предложенных сегментаций или проводят эксперименты на конкретных приложениях. Много работ опираются на эффективность и маленькую вычислительную сложность алгоритма построения декомпозиции. О стабильности или устойчивости предложенных сегментаций говорят немногие и, даже если говорят, то обычно подразумевают под этим устойчивость к шумам на границе фигуры. Таким образом, одной из *нерешенных*

задач общей проблемы сегментации остается вопрос формализации ее качества на основе устойчивости, стабильности. И этот вопрос является общим для всех приложений, в которых используется декомпозиция фигуры. Поэтому основной *целью настоящей статьи* является формальное определение устойчивости или стабильности сегментации фигуры. Также в работе приводится обоснование устойчивости метода скелетной сегментации, предложенного в работе [6].

1. Основные понятия

Определение 1. *Нормальная область* [12] – ограниченная замкнутая область, граница которой представляет собой объединение конечного числа замкнутых контуров, каждый из которых в свою очередь состоит из конечного числа участков аналитических кривых.

Определение 2. *Фигурой* назовем непрерывную аппроксимацию растрового изображения некоторой нормальной областью.

Определение 3. *Скелетом нормальной области* [12] называется множество центров максимальных вписанных в него окружностей.

Скелет нормальной области можно рассматривать как планарный граф [12], так называемый *скелетный граф*. Его вершинами являются центры окружностей, касающихся границы в трёх и более точках, а также терминальные точки скелета, а ребрами – серединные оси фигуры, линии, состоящие из центров окружностей, касающихся границы в двух и более точках.

Определение 4. Под *сегментацией (декомпозицией)* фигуры будем понимать ее разбиение на конечное множество областей.

Обозначим U – множество всех фигур. Обозначим Z – множество всевозможных сегментаций всех фигур из U. Оператором сегментации назовем функцию $R : U \to Z$, которая ставит фигуре $u \in U$ в соответствие ее сегментацию $z \in Z$.

Определение 5. *Скелетной сегментацией* [6] фигуры будем называть её специальное разбиение (рис 1в,1г,1ж,1з) на собственные области ребер скелета. *Собственная область* [13] ребра скелета — это минимальное подмножество точек фигуры, ограниченное ребром скелетного графа и соответствующими *радиальными отрезками* (перпендикулярами, опущенными из вершин скелетного графа степени 3 и более).

Скелетной сегментацией будем также считать разбиение основанное на любом подграфе скелетного графа. Тогда параметром скелетной сегментации можно считать тот подграф, на основе которого сегментация строится.

Зададим оператор скелетной сегментации, который ставит фигуре из U ее скелетную сегментацию из Z:

$$R_{sk} : U \to Z \tag{1}$$

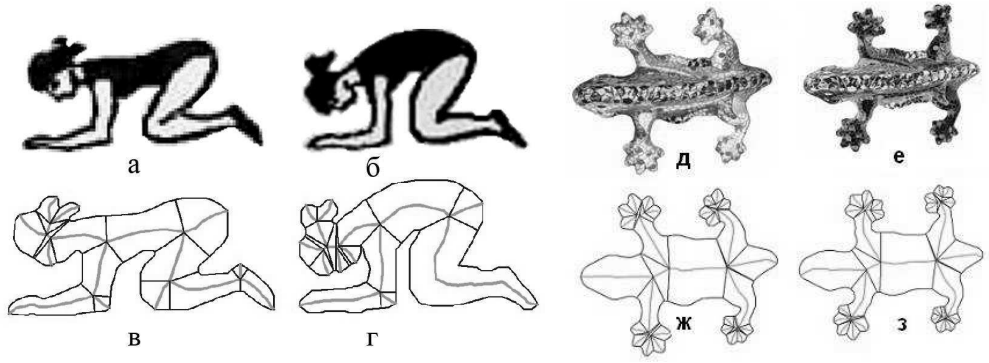


Рис. 1. Скелетная сегментация сходных изображений

2. Определение устойчивости сегментаций

Интуитивно понятно, что для задач распознавания и преобразования формы необходимо, чтобы похожие по форме фигуры имели похожую сегментацию (рис. 1).

Определение 6. *Устойчивость* применительно к геометрическим или иным объектам, зависящим от параметров, — это непрерывная зависимость этих объектов от параметров [14].

Определение 7. *Метод сегментации* фигуры с оператором $R : U \to Z$ будем называть *устойчивым* на паре метрических пространств (Z, U) с расстояниями $\rho_z(.,.)$ и $\rho_u(.,.)$, если для всякого $\xi > 0$ существует такое $\delta(\xi) > 0$, что для любых $u_1, u_2 \in U$ из неравенства $\rho_u(u_1, u_2) < \delta(\xi)$ следует неравенство $\rho_z(z_1, z_2) < \xi$, где $z_1 = R(u_1)$, $z_2 = R(u_2)$.

Для завершения формального определения устойчивости сегментации необходимо задать метрики $\rho_u(.,.)$ и $\rho_z(.,.)$ на пространствах U и Z соответственно.

3. Метрики на пространствах фигур и сегментаций

3.1. **Метрика** $\rho_u(.,.)$**.** Пусть есть две фигуры u_1 и u_2. Обозначим ma_1 и ma_2 — скелеты и соответственно. Метрику на пространстве фигур $\rho_u(.,.)$ зададим таким образом, чтобы она отвечала за сходство и различие форм. Будем считать, что форму фигур определяют их базовые скелеты [15].

Определение 8. Базовым скелетом sk_{base} фигуры u с точностью ε называется такой минимальный подграф скелета ma, что расстояние Хаусдорфа $\rho_{Hausdorff}(u, sil(sk_{base})) \leq \varepsilon$, где $sil(sk_{base})$ — силуэт базового скелета — объединение всех вписанных в u кругов, центры которых лежат на sk_{base} .

Обозначим sk_1^1 и sk_2^1 — базовые скелеты u_1 и u_2 с точностями ε_1^1 и ε_2^1 , а sk_1^2 и sk_2^2 — базовые скелеты u_1 и u_2 с точностями ε_1^2 и ε_2^2 соответственно. Будем считать, что для каждой фигуры u_i однозначно задаются точности ε_i^1 и ε_i^2.

Лемма 1. *Свойство базовых скелетов:*
$\varepsilon_1^2 \geq \varepsilon_1^1 \Rightarrow sk_1^2 \subseteq sk_1^1 \subseteq ma_1$ *и* $\varepsilon_2^2 \geq \varepsilon_2^1 \Rightarrow sk_2^2 \subseteq sk_2^1 \subseteq ma_2$.

Определение 9. Скелеты называются *изоморфными* ($sk_1 \cong sk_2$), если изоморфны их скелетные графы и при изоморфизме сохраняется направление обхода терминальных вершин.

Обозначим SK^*=$\{sk^* : sk_1^1 \supseteq sk_1^* \supseteq sk_1^2 | sk_2^1 \supseteq sk_2^* \supseteq sk_2^2 | sk_1^* \cong sk^* \cong sk_2^*\}$. То есть SK^* — множество скелетов, изоморфных одновременно базовым скелетам sk_1^* и sk_2^*. Элементы множества SK^* — ограничены снизу sk_1^2 и sk_2^2 , а сверху - sk_1^1 и sk_2^1 . Обозначим $\#sk$ — число ребер скелетного графа sk.

Определение 10 (Метрика на пространстве фигур ρ_u)**.**

$$\rho_u(u_1, u_2) = \begin{cases} \min\limits_{sk^* \in SK^*} \{(\#sk_1^1 - \#sk^*) + (\#sk_2^1 - \#sk^*)\} & \text{если} SK^* \neq \emptyset \\ \infty & \text{иначе} \end{cases} \quad (2)$$

Теорема 1. $\forall u \; \exists \; \varepsilon^1, \varepsilon^2 : \rho_u(.,.)$ *является полуметрикой на пространстве* U.

Доказательство. Докажем, что аксиомы метрики верны, причем аксиомы 1-3 верны для $\forall \varepsilon^1, \varepsilon^2$, для аксиомы 4 $\exists \; \varepsilon^1, \varepsilon^2$ что она также верна.

1. $\rho_u(sk_1, sk_2) \geq 0$
   $sk^* \subseteq sk_1^1 \;\Rightarrow\; \#sk^* \leq \#sk_1^1$
   $sk^* \subseteq sk_2^1 \;\Rightarrow\; \#sk^* \leq \#sk_2^1$
   $$+ \begin{cases} \#sk_1^1 - \#sk^* \geq 0 \\ \#sk_2^1 - \#sk^* \geq 0 \end{cases} \Rightarrow \#sk_1^1 - \#sk^* + \#sk_2^1 - \#sk^* \geq 0 \Rightarrow \rho_u(sk_1, sk_2) \geq 0$$
2. $\rho_u(sk_1, sk_1)$=0
   $\rho_u(sk_1, sk_1)$=$\#sk_1^1 + \#sk_1^1 - 2\#sk^*$=$2(\#sk_1^1 - \#sk^*)$
   sk^* - максимальный подграф $sk_1^1 \Rightarrow sk^* \equiv sk_1^1 \Rightarrow \rho_u(sk_1,sk_1)$=$2(\#sk^* - \#sk^*)$=0
3. $\rho_u(sk_1, sk_2)$=$\rho_u(sk_2, sk_1)$
   Очевидно, следует из определения $\rho_u(.,.)$.
4. $\rho_u(sk_1, sk_2) + \rho_u(sk_2, sk_3) \geq \rho_u(sk_1, sk_3)$ — неравенство треугольника
   Рассмотрим разность $\rho_u(sk_1, sk_2) + \rho_u(sk_2, sk_3) - \rho_u(sk_1, sk_3)$=$A$
   Докажем, что $A \geq 0$.
   A=$\#sk_1^1 + \#sk_2^1 - 2\#sk_{12}^* + \#sk_2^1 + \#sk_3^1 - 2\#sk_{23}^* - \#sk_1^1 - \#sk_3^1 + 2\#sk_{13}^*$=
   =$2\#sk_2^1 - 2\#sk_{12}^* - 2\#sk_{23}^* + 2\#sk_{13}^*$=$2(\#sk_2^1 - \#sk_{12}^*) + 2(\#sk_{13}^* - \#sk_{23}^*)$
   Нетрудно убедиться, что $A \geq 0$, например, при ε^1=ε^2.

□

Таким образом, расстоянием между фигурами u_1 и u_2 будет сумма отличий от максимального «общего» скелетного подграфа фигур в некоторых рамках, определяемых точностями аппроксимации ε_1^1, ε_1^2, ε_2^1 и ε_2^2. Верхние рамки ε_1^1 и ε_1^2 необходимы для того, чтобы «бахрома» на краях не влияла на сходство и различие форм. Их можно положить равными размеру одного пикселя или выбрать исходя из зашумленности

границы. Нижние рамки ε_2^1 и ε_2^2 определяют предел точности аппроксимации фигуры, начиная с которой фигуры считаются далекими. Так как sk_1^2 и sk_2^2 зависят от ε_1^2 и ε_2^2 соответственно, можно, например, выбирать ε_1^2 и ε_2^2 так, чтобы было выполнено: $\#sk_1^2 \geq M$ и $\#sk_2^2 \geq M$, где M - либо фиксированное, либо выбранное в зависимости от $\#ma_1$ и $\#ma_2$ натуральное число. Отметим также, что в формуле (2), можно вместо ∞ положить некоторое большое число $A \gg \#ma_1 + \#ma_2$.

3.2. **Метрика $\rho_z(.,.)$.** На пространстве сегментаций зададим метрику $\rho_z(.,.)$ с помощью графов смежности сегментаций.

Определение 11. *Граф смежности сегментации* — плоский граф, вершины которого соответствуют сегментам, две вершины соединены ребром, когда соответствующие им сегменты являются смежными в разбиении (то есть имеют общую границу).

Пусть $z_1 = R(u_1)$ и $z_2 = R(u_2)$. — сегментации фигур u_1 и u_2 соответственно, а g_1^* и g_2^* — графы смежности сегментаций z_1 и z_2 соответственно.

Определение 12 (Метрика на пространстве сегментаций ρ_z)**.**

$$\rho_z(z_1, z_2) = \begin{cases} 0 & \text{если } g_1^* \cong g_2^* \\ 1 & \text{иначе} \end{cases} \tag{3}$$

Теорема 2. *$\rho_z(.,.)$ — полуметрика на пространстве Z.*

Доказательство.
1. $\rho_z(z_1, z_2) \geq 0$ по определению.
2. $\rho_z(z_1, z_1) = 0$, так как $g_1^* \cong g_1^*$.
3. $\rho_z(z_1, z_2) = \rho_z(z_2, z_1)$, так как отношение изоморфизма графа смежности симметрично: $g_1^* \cong g_2^* \Rightarrow g_1^* \cong g_2^*$ и $g_1^* \ncong g_2^* \Rightarrow g_1^* \ncong g_2^*$.
4. $\rho_z(z_1, z_2) + \rho_z(z_2, z_2) \geq \rho_z(z_1, z_3)$ — неравенство треугольника
Если $\rho_z(z_1, z_3) = 0 \Rightarrow$ неравенство треугольника выполнено $\forall z_2$.
Рассмотрим случай $\rho_z(z_1, z_3) = 1$. В этом случае неравенство треугольника может быть не выполнено только в одном случае: $\rho_z(z_1, z_2) = 0$ и $\rho_z(z_2, z_3) = 0$. Покажем, что это невозможно. Допустим, что $\rho_z(z_1, z_2) = 0$ и $\rho_z(z_2, z_3) = 0$. Тогда $g_1^* \cong g_2^*$ и $g_2^* \cong g_3^*$. Но тогда $g_1^* \cong g_3^* \Rightarrow \rho_z(z_1, z_3) = 0$, что противоречит тому что $\rho_z(z_1, z_3) = 1$.

□

Отметим, что ρ_z — полуметрика достаточно «бедная» (сегментации могут быть только близкими или далекими). Однако, такой метрики достаточно для задачи расчета гомеоморфизма [13]. Для использования скелетной сегментации в задачах распознавания, возможно, необходима более «богатая» метрика.

4. Устойчивость скелетной сегментации

Рассмотрим оператор скелетной сегментации $R_{sk\varepsilon^1}$ с параметром базовым скелетом с точностью ε^1. Напомним, что для каждого u точность ε^1 задается однозначно. Можно, например, фиксировать ε^1, положить $\varepsilon^1 = 1$ или оставить ε^1 задающимся однозначно от u, например, как оценку зашумленности границы u.

Теорема 3 (Критерий близости скелетных сегментаций). *Для того чтобы две скелетные сегментации фигур u_1 и u_2, построенные с помощью оператора $R_{sk\varepsilon^1}$ были близки в смысле метрики ρ_z необходимо и достаточно, чтобы базовые скелеты u_1 и u_2 были изоморфны.*

Доказательство. Будем обозначать базовый скелет фигуры u с точностью аппроксимации ξ так: $sk(u,\varepsilon)$. $V(g)$ и $E(g)$ — множества вершин и ребер графа g.

Достаточность. Пусть $sk(u_1,\varepsilon_1^1) \cong sk(u_1,\varepsilon_2^1)$.

Рассмотрим вспомогательный граф $g{=}g(u,\varepsilon^1)$, определяемый по фигуре u и ее базовому скелету $sk{=}sk(u,\varepsilon^1)$ единственным образом:

1. $g \supseteq sk$.

2. $V(g){=}V(sk)\bigcup B$, где B – множество всех точек касания перпендикуляров (радиальных отрезков), опущенных из $V(sk)$ на границу u.

3. $E(g){=}E(sk)\bigcup C\bigcup D$, где C - множество всех радиальных отрезков; D - множество ребер вида: (v_1,v_2), где $v_1,v_2 \in B$, причем никакая вершина $v_3 \in B$ не находится при обходе границы u между v_1 и v_2;

То есть граф g — это «расширенный» с помощью радиальных отрезков и разбитой на части границы фигуры u граф sk. Полученный граф g задает скелетную сегментацию фигуры u, построенную при помощи $R_{sk\varepsilon^1}$. Циклы графа g, внутри которых не лежит ни одного ребра g, являются границами областей сегментации $A(g)$.

Определение 13. Назовем построенный граф g — графом скелетной сегментации.

Граф g обладает следующими свойствами:

1. любая область $\forall a \in A(g)$ имеет границу вида: ребро скелета $e \in sk$, два радиальных отрезка $c_1, c_2 \in C$, часть границы u (может и не быть)

2. любая пара смежных областей a_1 и a_2 имеет только одно общее ребро: либо ребро скелета $e \in sk$, либо радиальный отрезок $e \in$ C.

Докажем вспомогательную лемму:

Лемма 2. *Если скелетные графы двух фигур изоморфны $sk_1 \cong sk_2$, то изоморфны и соответственные их графы скелетной сегментации $g_1 \cong g_2$.*

Доказательство. Обозначим $Q{=}(v,r) : sk_1 \to sk_2$ — изоморфизм на скелетах. Так как $g \supseteq sk \Rightarrow$, для построения $\overline{Q}{=}(\overline{v},\overline{r}) : g_1 \to g_2$, надо доопределить изоморфное отображение вершин $\{V(g)\}\backslash V\{sk\}{=}B$ и ребер $\{E(g)\}\backslash E\{sk\}{=}C\bigcup D$.
Так как $\#V(u_1){=}\#V(u_2) \Rightarrow \#B_1{=}\#B_2$. Поставим эти множества вершин в соответствие друг другу последовательно при правильном обходе границ u_1 и u_2. $\#\mathrm{C}_1{=}\#\mathrm{C}_2$ и каждый радиальный отрезок имеет ровно одну инцидентную вершину $b \in B \Rightarrow$ зададим соответствие между радиальными отрезками так: $c_1 \in C_1 \leftrightarrow c_2 \in C_2 \Leftrightarrow b_1 \in B_1 \leftrightarrow b_2 \in B_2$, где c_i инцидентно $b_i; i{=}1,2$.
Наконец, $\#D_1{=}\#D_2$ — это ребра графов скелетных сегментаций g_1 и g_2, каждое из которых имеет ровно две инцидентные вершины из множеств B_1 и B_2 соответственно. Зададим соответствие $D_1 \leftrightarrow D_2$ так: $d_1 \in D_1 \leftrightarrow d_2 \in D_2 \Leftrightarrow b_1^1 \in B_1 \leftrightarrow b_2^1 \in B_2$ и $b_1^2 \in B_1 \leftrightarrow b_2^2 \in B_2$, где $d_1{=}(b_1^1,b_1^2)$ и $d_1{=}(b_2^1,b_2^2)$.

Нетрудно убедиться, что построенное соответствие $\overline{Q}$ — изоморфизм графов g_1 и g_2. □

По лемме имеем $g_1 \cong g_2$, а значит, и графы смежности g_1 и g_2 тоже изоморфны. □

Необходимость. Пусть $\rho_z(z_1{=}R_{sk\varepsilon^1}(u_1), z_2{=}R_{sk\varepsilon^1}(u_2))=0 \Rightarrow$ {по определению ρ_z} $\Rightarrow$ графы смежности сегментаций z_1 и z_2 изоморфны $\Rightarrow$ изоморфны и графы скелетных сегментаций, определенные в доказательстве достаточности $g_1 \cong g_2$, но $g_1 \supseteq sk_1$ и $g_2 \supseteq sk_2$. Рассмотрим процесс получения sk_1 и sk_2 последовательным удалением соответствующих в изоморфизме ребер g_1 и g_2, не являющихся ребрами sk_1 и sk_2. Такой процесс приведет нас к искомому изоморфизму $sk_1 \cong sk_2$. □

Теорема 3 доказана. □

Теорема 4 (Устойчивость скелетной сегментации). *Метод скелетной сегментации с оператором $R_{sk\varepsilon^1} : U \to Z$ и метриками ρ_u (2) и ρ_z (3) является устойчивым на пространствах фигур U и сегментаций Z.*

Доказательство. Доказательство основано на определении устойчивости метода сегментации (7). Разобьем его на две части.

1. $0 < \xi < 1$

Пусть $\delta(\xi){=}0.5 > 0$. Тогда $\rho_u(u_1, u_2){=}(\#sk_1^1 - \#sk^*) + (\#sk_2^1 - \#sk^*) \le 0.5$
Так как число ребер - целая величина $\Rightarrow$ $(\#sk_1^1 - \#sk^*) + (\#sk_2^1 - \#sk^*) \equiv 0$. При этом из свойств базового скелета $\#sk_1^1 \ge \#sk^*$ и $\#sk_2^1 \ge \#sk^*$. Поэтому $\#sk_1^1 \equiv \#sk^*$ и $\#sk_2^1 \equiv \#sk^*$. Отсюда $sk_1^1 \cong sk^*$ и $sk_2^1 \cong sk^* \Rightarrow sk_1^1 \cong sk_2^1$. То есть базовые скелеты u_1 и u_2 с точностями ε_1^1 и ε_2^1 изоморфны. А значит, по критерию близости скелетных сегментаций (теорема 3) $\rho_z(z_1, z_2){=}0 \Rightarrow \rho_z(z_1, z_2) \le \xi$.

2. $\xi \ge 1$

В этом случае для любого $\delta(\xi) > 0$ верно следующее: $\forall u_1, u_2 \in U : \rho_u(u_1, u_2) < \delta(\xi) \Rightarrow \rho_z(z_1, z_2) \le \xi$. Это верно всегда, так как $\rho_z(z_1, z_2) \le 1$, а $\xi \ge 1$. Таким образом, для любого ξ мы можем подобрать $\delta(\xi) > 0$ $\forall u_1, u_2 \in U : \rho_u(u_1, u_2) < \delta(\xi) \Rightarrow \rho_z(z_1, z_2) \le \xi$. Теорема доказана. □

Заключение

Данная работа посвящена исследованию качества сегментации фигур. Анализ последних работ в данном направлении показывает, что данный вопрос изучен недостаточно. Вопросы стабильности, устойчивости к шумам и деформациям поднимались во многих работах, но формального определения данных понятий не обнаружено. В настоящей работе *получены следующие результаты*:

- Введено понятие устойчивости метода сегментации на метрических пространствах фигур и сегментаций.

- Предложены полуметрики на пространствах фигур и сегментаций. ρ_u — полуметрика на пространстве фигур, в ее основе лежит сравнение формы на основе базового скелета. ρ_z — полуметрика на пространстве сегментаций, в ее основе — граф смежности сегментации.
- Доказана устойчивость метода скелетной сегментации с определенными параметрами на с полуметриками ρ_u и ρ_z.

Работа имеет *перспективы* приложения к задачам морфинга и распознавания, так как устойчивые сегментации успешно могут быть в данных задачах использованы. В качестве развития данной работы можно попытаться определить более «богатые» метрики ρ_u и ρ_z и доказать устойчивость метода скелетной сегментации с другими операторами, помимо $R_{sk\varepsilon^1}$.

Список литературы

1. *Mirela Tanase* Shape Decomposition and Retrieval //PhD Thesis, Utrecht University 2005 г.
2. *Davi Geiger, Tyng-Luh Liu, and Robert V.Kohn* Representation and Self-Similarity of Shape // IEEE Transactions on Pattern Recognition Analysis and Machine Intelligence, vol.25, no. 1, 2003, p.86-99.
3. *Maryann Simmons and Carlo H.Sequin* 2D Shape Decomposition and the Automatic Generation of Hierarchial Representations //International Journal of Shape Modelling, 1998.
4. *P.M.Hubbard* Collision Detection for Interactive Graphics Applications //IEEE Transactions on Vizualization and Computer Graphics, p.218-230, 1995.
5. *Jyh-Ming Lien and Nancy M. Amatoy* Simultaneous Shape Decomposition and Skeletonization //Technical Report TR05-015 Parasol Lab. Department of Computer Science Texas AM University, 2005
6. *Домахина Л.Г.* Об одном методе сегментации растровых объектов для задач преобразования формы // Труды 13 Всероссийской конф. Математические Методы Распознавания Образов (ММРО-13),Москва 2007, с. 311-314.
7. *C. Vasanthanayaki, S. Annadurai* Flexible Search-Based Approach for Morphological Shape Decomposition //1995.
8. *C. Vasanthanayaki, S. Annadurai* Optimal Morphological Shape Decomposition Scheme //ICGST-GVIP Journal, Volume (5), Issue (7), July 2005.
9. *M.A.Abidi and R.C.Gonzalez* Shape Decomposition Using Elliptic Fourier Descriptors //Proc. 18th IEEE Southeast Sympo. Sys. Theory, pp. 53-61, Knoxville, TN, April 1986.
10. *Paul L. Rosin* Shape Partitioning by Convexity //Proc. of British Machine Vision Conference, 2000.
11. *Longin Jan Latecki and Rolf Lakaamper* Convexity Rule for Shape Decomposition Based on Discrete Contour Evolution //Computer Vision and Image Understanding Vol. 73, No. 3, March, pp. 441–454, 1999
12. *Choi H.I., Choi S.W., Moon H.P.*Mathematical theory of medial axis transform // Pacific.J. of Math. 1997. Vol. 181, No. 1. P. 57-88.
13. *Петрова Л.Г., Местецкий Л.М.* Расчет гомеоморфизма односвязных многоугольных областей с изоморфными базовыми скелетами // Сборник «Искусственный интеллект», Таврический национальный университет им. В.И. Вернадского, г. Симферополь, Украина, 2006, с. 192-197.
14. *Виноградов И.М.* Устойчивости теория. // Математическая энциклопедия. Том 5, 1977, с. 551-553.

15. *Местецкий Л.М., Рейер И.А.* Непрерывное скелетное представление изображения с контролируемой точностью. // Труды 15 международной конференции ГРАФИКОН-2003, 2003 г., с. 246–249

16. *Edward M. Reingold, Jurg Nievergelt, Narsingh Deo*Combinatorial Algorithms, Theory and Practice // Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632, 1977, с. 396-402

17. *Thomas B. Sebastian, Benjamin B. Kimia*Curves vs skeletons in object recognition // Proceedings of 2001 International Conference of Image Processing(ICIP-2001), Thessaloniki, Grece,vol.3, с. 22-25

Статья поступила в редакцию 27.04.2008

УДК 5:519.876

УЧЁТ ЧЕЛОВЕЧЕСКОГО ФАКТОРА В ЗАДАЧАХ ПРИНЯТИЯ РЕШЕНИЙ ДЛЯ ОРГАНИЗАЦИОННЫХ СИСТЕМ УПРАВЛЕНИЯ[1]

© Дорофеюк А.А., Гольдовская М.Д., Чернявский А.Л.

Институт проблем управления РАН

Abstract. The decision-making methods, that allow to reduce human factor influence, and sometimes – to use it for receiving additional information, are discussed. For solving similar problems the use of collective multivariate expert examination procedures is proposed.

Введение

В процессе функционирования организационных и социально-экономических систем часто возникают ситуации, разрешение которых требует нестандартных решений. В таких случаях лицо принимающее решение (ЛПР) вынуждено при выработке решения ориентироваться не столько на стандартные схемы, сколько на информацию от специалистов-экспертов. Для этой цели используются разнообразные методы экспертизы, включающие процедуры сбора, обработки и интерпретации экспертной информации.

В классических методах экспертизы, как правило, предлагается набор готовых вариантов решений, заданы чёткие критерии оценки этих решений, а при обработке экспертной информации используются простейшие процедуры математической статистически. Однако, возникающие в организационных системах проблемы не имеют стандартных вариантов решения, поэтому для их подготовки классические методы непригодны. При анализе экспертных оценок аргументация экспертов часто бывает важнее самих оценок, поскольку позволяет получить дополнительную информацию о степени их обоснованности. Более того, так как в подавляющем числе случаев к принятию решения привлекаются эксперты, работающие в той же системе, их оценки не могут быть беспристрастными, поскольку невозможно абстрагироваться от последствий принятого решения, которые будут иметь место для него лично или для подразделения, в котором он работает. В этом проявляется сильное влияние человеческого фактора на экспертные процедуры принятия решений. Влияние человеческого фактора ещё больше усиливается из-за особенностей личных взаимоотношений экспертов – конфликтность, психологическая несовместимость, взаимоотношения типа «начальник-подчиненный» и т.п. Именно из-за неконтролируемого влияния человеческого фактора классические методы экспертизы и обработки экспертных оценок в таких случаях неприменимы.

В работе рассматриваются методы принятия решений с участием экспертов, позволяющие уменьшить отрицательное влияние человеческого фактора, а иногда – использовать его для получения дополнительной информации. Для решения подобных задач наиболее адекватным является использование методологии коллективной многовариантной экспертизы [1].

[1]Работа выполнена при частичной финансовой поддержке РФФИ, проекты 08-07-00349-а, 08-07-00427-а.

1. Методология коллективной многовариантной экспертизы

Любая коллективная экспертиза предполагает, что наиболее эффективным способом сопоставления, оценки и согласования мнений экспертов является коллективное обсуждение в экспертной комиссии [1]. При обсуждении важных для организации и работающих в ней людей вопросов практически всегда имеется несколько различных, а иногда и прямо противоположных точек зрения. При этом каждая из таких точек зрения объективно имеет свои достоинства и недостатки, которые можно было бы выявить в процессе обсуждения. Однако опыт показывает, что совместная работа в одной комиссии экспертов, придерживающихся разных точек зрения, как правило, оказывается малопродуктивной [1].

Кроме того, даже эксперты, имеющие сходные точки зрения, иногда не могут работать в одной комиссии из-за особенностей личных взаимоотношений. Поэтому вместо того, чтобы сталкивать между собой людей, придерживающихся взаимоисключающих точек зрения, не имеющих возможности обсуждать спорные вопросы на равных и т.д., целесообразно детально проработать каждую точку зрения в комиссии, состоящей из экспертов, придерживающихся приблизительно одинаковой точки зрения и не имеющих конфликтных взаимоотношений.

Методы коллективной многовариантной экспертизы предполагают, что экспертизу проводит специальная консалтинговая группа, приглашенная, для большей объективности, со стороны. Роль входящих в эту группу консультантов состоит в том, что они проводят всю организационную, подготовительную и расчетную работу; участвуют в работе экспертных комиссий; проводят компьютерную и содержательную обработку экспертных мнений и представляют результаты обработки и свое собственное мнение заказчику – лицу, принимающему решение (ЛПР). Такой способ организации коллективного обсуждения обеспечивает каждой из сторон равные возможности.

1.1. **Концепция коллективной многовариантной экспертизы.** Концепция коллективной многовариантной экспертизы базируется на следующих основных принципах:

- экспертиза проводится в экспертных комиссиях, число которых не меньше числа различных точек зрения на исследуемую проблему;
- в одну и ту же комиссию включаются эксперты, имеющие близкие точки зрения на проблему экспертизы;
- в каждой комиссии работают эксперты, не имеющие конфликтных взаимоотношений;
- для экспертизы отбираются условно компетентные эксперты;
- организация и проведение экспертизы, обработка экспертных оценок, формирование результатов экспертизы должны проводиться специальной консалтинговой группой, независимой и не заинтересованной в результатах экспертизы.

1.2. **Методика формирования экспертных комиссий.** Эти принципы были реализованы в рамках специальной методики формирования экспертных комиссий. Методика включает пять основных разделов (этапов): выявление кандидатов для работы в экспертных комиссиях; выявление существенно различных точек зрения;

определение групп неконфликтующих экспертов; оценка условной компетентности экспертов; формирование экспертных комиссий.

Этап 1. Формирование списка кандидатов в эксперты.

Для формирования списка кандидатов в эксперты предлагается использовать стандартные схемы типа «снежный ком» [2].

Этап 2. Выявление существенно различных точек зрения и классификация экспертов.

Информация, характеризующая точки зрения экспертов на проблемы организации, собирается в ходе бесед (интервью) членов консалтинговой группы с каждым экспертом, а также с помощью специально разрабатываемых анкет. Основой таких анкет и интервью является составленный в процессе предварительного «пилотного» обследования перечень возможных вариантов реформирования разных аспектов деятельности организации. Тогда «точка зрения» эксперта в первом приближении описывается набором его ответов на вопросы анкеты. Для выявления типовых точек зрения и классификации экспертов в работе используются методы многомерной автоматической классификации [3]. Каждый эксперт представляется точкой в k-мерном пространстве характеристик X (пространстве точек зрения). Тогда задача структуризации мнений n экспертов сводится к задаче структуризации в X n точек $x_1, \ldots, x_n$ на «компактные» группы (кластеры) $A_1, \ldots, A_R$. Считается, что эксперты, попавшие в одну группу, имеют сходные точки зрения.

Если количество вопросов в анкете достаточно велико, то пространстве точек зрения X имеет большую размерность. В таких случаях предлагается использовать процедуры выделения информативных параметров, например, алгоритмы экстремальной группировки «квадрат» в составе комплексного алгоритма классификации [3].

Этап 3. Выявление групп не конфликтующих экспертов.

Исходная информация на этом этапе – заполняемые каждым экспертом две анкеты, отражающие его оценки взаимоотношения экспертов.

В первой анкете эксперт отвечает на вопросы, характеризующие его взаимоотношения с другими экспертами. Опыт показывает, что задавать здесь прямые вопросы опасно, так как велика вероятность получения дезинформации. Дело в том, что на взаимоотношения экспертов накладывается целый ряд ограничений, связанных с влиятельностью, соподчиненностью, родственными и дружескими связями и т.д. По этой причине многие эксперты уклоняются от разговоров даже о наличии конфликтности, не говоря уже об оценке её уровня. В такой ситуации хорошие результаты дают так называемые косвенные вопросы, ответы на которые в совокупности дают существенно более надежную информацию, чем ответ на прямой вопрос.

Вопросы второй анкеты касаются оценки опрашиваемым экспертом взаимоотношений между другими экспертами. Ввиду специфики вопроса эта анкета заполняется обычно членом консалтинговой группы после интервью с экспертом. Обработка этих анкет производится также с помощью комплексного алгоритма классификации [3].

Вся информация, касающаяся взаимоотношений экспертов, представляется в виде n матриц отношений $B_j = \left\| b_{is}^j \right\|, \quad i = 1, \ldots, j-1, j+1, \ldots, n; \quad s = 1, \ldots, k_2,$

каждая из которых отражает отношение j-го эксперта к остальным. Здесь k_2 – число вопросов анкеты, b_{is}^j – ответ j-го эксперта на s-й вопрос анкеты относительно i-го эксперта. Каждая из этих матриц обрабатывается независимо, в силу чего опишем процедуру обработки матрицы B_j, соответствующей j-му эксперту.

Введем в рассмотрение k_2-мерное пространство X_{2j}, характеризующее отношение j-го эксперта к остальным. Тогда каждая строка $x_i^j = \left\{b_{i1}^j, \ldots, b_{ik_2}^j\right\}$ матрицы B_j, соответствующая i-му эксперту, может быть представлена в виде точки в этом пространстве. Если j-й эксперт приблизительно одинаково относится к некоторым двум экспертам, то соответствующие этим экспертам точки должны быть достаточно близки друг другу. Для выявления структуры взаиморасположения точек (их всего n-1) также используется алгоритм автоматической классификации [3]. С помощью этого алгоритма производится разбиение точек на d групп $D_1, \ldots, D_d$. Поскольку в данном случае необходимо разделить всех экспертов на «конфликтующих» и «неконфликтующих», то число групп выбирается небольшим – порядка $r = 2 - 4$. Заметим, что величина b_{is}^j, оценивающая «неконфликтность» экспертов, положительна, а оценивающая «конфликтность» – отрицательна.

Затем полученные группы упорядочиваются по степени «неконфликтности» с j-м экспертом. Для этой цели используется специальная процедура упорядочения групп D_i, $i = 1, \ldots, d$ по отношению к точке x^*, соответствующей «идеально неконфликтному эксперту». Затем производится их ранжировка по степени неконфликтности. Обычно для этой цели используют некоторую непрерывную шкалу неконфликтности, и каждой группе D_i^* присваивается определенное значение показателя неконфликтности в этой шкале. При решении прикладных задач часто используется другой, более простой способ, когда все группы делятся на две части – конфликтные и неконфликтные, для чего вводится порог неконфликтности a.

Описанная процедура обработки матрицы B_j выполняется для всех $9j = 1, \ldots, n$. Информация, полученная в результате такой обработки, сводится в матрицу отношений $V = \|v_{ij}\|$, каждый элемент которой v_{ij} равен значению показателя неконфликтности j-го эксперта по отношению к i-му, причем v_{ij} равно либо 1, либо 0.

Затем производится обработка матрицы V с целью выделения групп наиболее неконфликтующих экспертов. Для этой цели бинарная матрица V рассматривается как матрица смежности некоторого ориентированного графа (орграфа) с n вершинами, каждая из которых соответствует определенному эксперту. Наличие дуги из i-й вершины в j-ю (т.е. в случае v_{ij}=1) свидетельствует о том, что i-й эксперт неконфликтен с j-м экспертом (отсюда, вообще говоря, не следует, что j-й эксперт неконфликтен с i-м, т.к. матрица V может быть и несимметрична). Полученный орграф преобразуется в простой граф Г с n вершинами по следующему правилу: i-я и j-я вершины соединяются дугой в том и только в том случае, когда $v_{ij} = v_{ji} = 1$ (взаимно неконфликтные эксперты).

Тогда группе экспертов, каждая пара из которой взаимно неконфликтна, будет соответствовать некоторый полный подграф графа Г. Таким образом задача нахождения искомого разбиения $V_1, \ldots, V_{r_2}$ сводится к выделению полных подграфов

графа Г (их число обозначено через r_2) [4]. Для не бинарной матрицы V разбиения $V_1, \ldots, V_{K_2}$ получаются с помощью алгоритмов диагонализации матрицы связи [5].

Этап 4. Оценка условной компетентности экспертов.

Необходимость этого этапа связана с тем, что оценка компетентности существенно зависит от состава группы, в которой эта оценка получается. Поэтому компетентность каждого эксперта должна оцениваться с точки зрения только тех экспертов, которые будут с ним работать в одной комиссии («условная компетентность»).

Исходной информацией для этого этапа являются специальные анкеты, которые заполняет сам эксперт. В анкетах оценки проставляются для всех экспертов, однако, в итоге будут использованы оценки только для тех экспертов, которые войдут в ту же самую экспертную комиссию.

В первой анкете каждый эксперт дает оценку компетентности остальных экспертов по различным разделам (направлениям), имеющим отношение к предметной области экспертизы. Здесь необходимо отдавать предпочтение косвенным вопросам. Вторая анкета содержит вопросы по ключевым направлениям (проблемам) предметной области экспертизы. Ответы на нее позволяют «объективно» оценить профессионализм и кругозор эксперта. Формирование вопросов этой анкеты представляет собой самостоятельную задачу, которая также решается методами экспертизы. Информация, получаемая с помощью этой анкеты, используется для коррекции ответов экспертов по первой анкете, уточнения окончательных результатов ее обработки, получения абсолютных (реперных) значений на шкале условной компетентности.

Ответы экспертов на первую анкету с учетом информации, полученной из второй анкеты, представляются в виде $n \times n$ матрицы компетентности $K = \|k_{ij}\|$, каждый элемент которой k_{ij} является интегральной оценкой компетентности j-го эксперта, полученной с помощью i-го эксперта.

Если задано некоторое разбиение экспертов на группы $C_1, \ldots, C_q$, а j-й эксперт находится в s-й группе C_s, то в качестве меры «условной» компетентности u_j предлагается использовать среднее по s-й группе величин k_{ij} для j-го эксперта, т.е. величину $u_j = {}^1\!/_{n_s} \sum\limits_{i/x_i \in C_s} k_{ij}$, где n_s – число экспертов в группе C_s. Затем для u_j подсчитывается нижняя граница доверительного интервала $\Delta_j^{(H)}$. Если оказывается, что $u_j < a$, $\Delta_j^{(H)} < b$, где a и b – заданные пороговые значения, то j-й эксперт считается некомпетентным в группе C_s. Если выполняются неравенства $u_j \geq a \Delta_j^{(H)} < b$, то необходимо провести уточнение данных по j-му эксперту в группе C_s. И, наконец, если $u_j \geq a$, $\Delta_j^{(H)} > b$, то эксперт считается компетентным в группе C_s.

Этап 5. Формирование экспертных комиссий.

На заключительном этапе производится формирование экспертных комиссий.

Выше были сформулированы специальные требования к числу и составу экспертных комиссий. Формирование комиссий, удовлетворяющих всем этим требованиям, производится с помощью следующей процедуры.

Рассматриваются разбиения по точкам зрения $A = \{A_1, \ldots, A_{r_1}\}$, по взаимоотношениям (неконфликтности) $V = \{V_1, \ldots, V_{r_2}\}$. Строится пересечение разбиений A и V, т.е. формируются группы экспертов $A_i \cap V_j$, $i = 1, \ldots, r_1$; $j = 1, \ldots, r_2$.

Рассмотрим группы такого вида, содержащие более одного эксперта, и обозначим их через $E = \{E_1, \ldots, E_{r_3}\}$, где r_3 – общее число таких групп. Для каждой группы E_i определяется матрица компетентности K и из неё исключаются условно некомпетентные эксперты. Оставшиеся группы, содержащие более двух экспертов, обозначим через $E' = \{E'_1, \ldots, E'_{r_4}\}$, где r_4 – общее число таких групп.

Для формирования j-ой экспертной комиссии из множества E' выбирается группа $E'_{j\,\max}$, в которую входит максимальное число экспертов, имеющих j-ую точку зрения. Эксперты, входящие в эту группу, и составляют j-ю экспертную комиссию. Так формируются все экспертные комиссии. На практике иногда приходится формировать дополнительные экспертные комиссии, в которые входят либо высококомпетентные эксперты, не включенные ни в одну из сформированных комиссий по соображениям конфликтности, либо высокопоставленные чиновники исследуемой организации из соображений престижа.

1.3. **Методы работы экспертных комиссий.** На ***этапе анализа*** работа сформированных комиссий проходит по специальному сценарию. На заседаниях комиссий в основном обсуждаются разногласия и спорные вопросы, выявляемые по результатам анализа заранее заполняемых экспертами анкет.

Результаты анализа представляются каждой комиссией в виде развернутого заключения, в котором отмечаются недостатки и причины, их вызывающие.

На ***этапе разработки предложений*** по совершенствованию исследуемой системы изменяется цель работы экспертных комиссий – переход от суммирования существенных недостатков к выбору варианта предложений, в максимальной степени их устраняющих. Это влечет за собой изменения методики работы экспертных комиссий, поскольку на этапе разработки предложений проявляется различие точек зрения экспертов из разных комиссий. В этих условиях эксперты стремятся подчеркнуть преимущества и сгладить недостатки «своих» предложений. Задачей же экспертизы в целом является получение объективных характеристик (как положительных, так и отрицательных) каждого из предложений.

Для получения таких характеристик используется специальная ***процедура перекрестной экспертизы.***

Суть этой процедуры состоит в следующем. После обсуждения каждого из узловых вопросов в экспертной комиссии подготавливается предпроект № 1. Каждый такой предпроект передается для обсуждения в другие комиссии. Замечания по каждому предпроекту №1, высказанные остальными комиссиями, передаются в комиссию, подготовившую этот предпроект. Комиссия обсуждает сделанные замечания, вносит коррективы – как в свои предложения, так и в список положительных и отрицательных сторон этих предложений. В итоге появляется предпроект №2 и т.д. Итерация таких перекрестных экспертиз заканчивается, когда дополнительное обсуждение не дает изменений предварительного проекта. Совокупность итоговых проектов с замечаниями исследовательской группы является результатом работы комиссий на втором этапе.

При исследовании крупномасштабных организационно-экономических систем возникают серьезные проблемы совместной работы экспертов, поскольку их места

работы (и проживания) разбросаны на большой территории. Для таких систем была разработана ***методология заочной многовариантной экспертизы***. Основной особенностью такого варианта коллективной экспертизы является то, что эксперты обсуждают исследуемые проблемы заочно по результатам предварительного анкетирования. Она имеет важное преимущество – не требуется отбирать в одну комиссию неконфликтующих экспертов, поскольку каждому эксперту информация представляется обезличенно.

Заключение

Разработанная методология коллективной многовариантной экспертизы использовалась при решении ряда прикладных задач, в том числе: разработка стратегии реформирования регионального пассажирского автотранспорта, развитие системы регионального здравоохранения; совершенствование межбюджетных отношений федерального центра и субъектов РФ; совершенствование налоговой политики и системы сбора налогов; совершенствование системы оплаты труда в бюджетной сфере, обоснованный выбор минимального размера оплаты труда (МРОТ); анализ и совершенствование системы управления ряда крупных предприятий и организаций.

Список литературы

1. *Дорофеюк А.А., Чернявский А.Л.* Консультативная работа по совершенствованию управления в организационных системах (методологические основы) / Сб.: Методы и алгоритмы анализа эмпирических данных. / – М.: ИПУ. 2008.
2. *Панкова Л.А., Петровский А.М., Шнейдерман М.В.* Организация экспертизы и анализ экспертной информации / – М.: Наука. 1984.
3. *Дорофеюк Ю.А.* Комплексный алгоритм автоматической классификации и его применение для анализа и принятия решений в больших системах управления. / Теория активных систем. Труды международной научно-практической конференции. / – М.: ИПУ РАН. 2007. – С. 39-42.
4. *Басакер Р., Саати Т.* Конечные графы и сети / – М.: Наука. 1974.
5. *Браверман Э.М., Дорофеюк А.А., Лумельский В.Я., Мучник И.Б.* Диагонализация матрицы связи и выделение скрытых факторов / Сб.: Проблемы расширения возможностей автоматов. Вып.1. / – М.: ИАТ. 1971.

Статья поступила в редакцию 27.04.2008

УДК 612.821:615.47:616.12-008

КЛАССИФИКАЦИОННЫЙ АНАЛИЗ ХАРАКТЕРИСТИК ПУЛЬСОВОГО СИГНАЛА В ЗАДАЧАХ ДИАГНОСТИКИ СЕРДЕЧНО-СОСУДИСТЫХ ЗАБОЛЕВАНИЙ[1]

© Дорофеюк А.А., Гучук В.В., Десова А.А., Дорофеюк Ю.А., Покровская И.В.

Институт проблем управления РАН

Abstract. The results of using especially developed complex of range analysis algorithms and pattern recognition for cardiovascular disease diagnostic by the example of the initial stage hypertension diagnostics in infancy and adolescence period are described.

Введение

Многочисленные исследования по анализу и оценке вариабельности сердечного ритма (ВСР) свидетельствуют о существенной диагностической значимости показателей ВСР при различных патологических состояниях человека, в том числе связанных с заболеваниями сердечно-сосудистой системы.

На базе специально разработанных стандартов [1] были выявлены основные колебательные компоненты, присущие волновой структуре сердечного ритма, каждый из которых отражает соответствующую физиологическую область регуляции. В области, так называемых, «медленных» (секундные, декасекундные, многоминутные) волн выделяются следующие колебательные компоненты:

– «быстрые колебания» (HF) с частотой $0.15 \div 0.4$ гц,

– «медленные волны 1-го порядка» (LF) с частотой $0.04 \div 0.15$ гц,

– «медленные волны 2-го порядка» (VLF) с частотой $0.003 \div 0.04$ гц.

Колебания HF отражают влияние парасимпатического отдела вегетативной нервной системы и связаны с вагусной активностью. Волны LF отражают преимущественное влияние симпатико-адреналовой системы. Волны VLF характеризуют влияние высших вегетативных центров на сердечно-сосудистый подкорковый центр.

Ритмологические исследования дают ценную информацию о состоянии организма, о различных его патологиях, позволяют давать прогноз течения болезни при лекарственных и операционных воздействиях, в том числе на ранних и скрытых периодах заболевания [2].

Большинство современных ритмологических исследований основаны на анализе какого-либо одного параметра сигнала, - чаще всего исследуются RR - интервалы сердечного ритма. Однако, в ряде работ (см., например, [3]) отмечается необходимость исследования других параметров биосигнала с оценкой их взаимосвязи, что может значительно расширить диагностические возможности ритмологических исследований.

В настоящей работе подобное исследование проводится на примере сигнала периферического пульса лучевой артерии. Известно, что в пульсовом сигнале находят

[1]Работа выполнена при частичной финансовой поддержке РФФИ, проекты 05-08-50312-а, 08-07-00349-а.

своё отражение как процессы высших уровней регуляции, так и чисто гемодинамические показатели сердечно-сосудистой системы. Методика исследования пульсового сигнала основана на синхронном анализе колебательных компонент, присущих различным функционально - значимым элементам пульсового сигнала и оценке их взаимосвязи [4].

Анализ пульсового сигнала проводился на базе обширного экспериментального материала, полученного в ходе клинических обследований более 350 пациентов в Научном центре здоровья детей РАМН [5]. Все виды заболеваний обследуемых были разделены на 2 класса: 1-ый класс (165 пациентов) – первичная артериальная гипертензия, 2-ой класс (190 пациентов) – различные виды психосоматической функциональной патологии с нормальным артериальным давлением.

1. Информационные возможности пульсового сигнала

Форма единичного колебания пульсового сигнала лучевой артерии схематически представлена на рис.1.

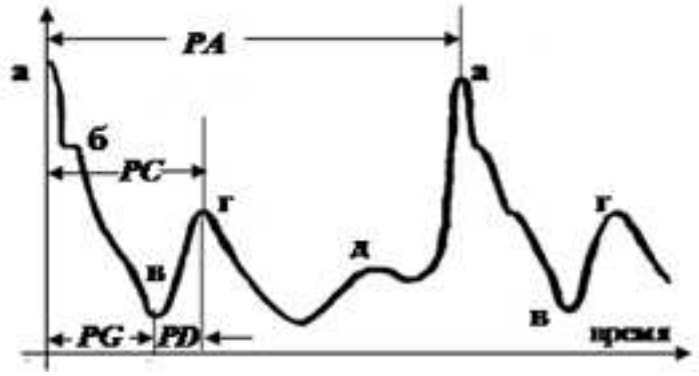


Рис. 1. Форма единичного колебания пульсового сигнала лучевой артерии

Здесь обозначены: «а» – максимум систолической волны, «б» – поздняя систоличе-ская волна, «в» – начало дикротической волны (инцизура), «г» – максимум дикротиче-ской волны, «д» – максимум пресистолической волны, PA – период пульсовой волны, PC – время достижения максимума дикротической волны, PG – время падения основной волны, PD – время подъема дикротической волны.

Информация о пульсовом сигнале, может быть представлена в двух видах – значения параметров формы пульсовой волны; информация, отражающая динамические изменения элементов формы пульсовой волны и тем самым характеризующая ритмическую структуру сигнала в целом. Автоматизированные методы анализа пульсового сигнала должны предусматривать обработку обоих видов информации.

2. Основные этапы многомерного анализа ритмической ритмической структуры пульсового сигнала

Методика анализа ритмической структуры пульсового сигнала предусматривает идентификацию значимых элементов в пределах его основного квазипериода, синхронное вычисление колебательных компонент, отражающих динамику амплитудно-временных параметров этих элементов, оценку их взаимно-корреляционных связей и степени синхронизации. Методика состоит из следующих этапов:

1. Автоматическое выделение характерных элементов пульсовой волны в пределах основного квазипериода, соответствующих основной, дикротической и пресистолической волнам (см. рис. 1), а также вычисление амплитудных и временных параметров этих элементов.

2. Формирование динамических рядов амплитудных и временных параметров характерных элементов, каждый из которых представляет собой зависимость значений данного параметра от номера периода.

3. Вычисление статистических характеристик этих динамических рядов.

4. Вычисление спектральной плотности всех сформированных динамических рядов и оценка взаимных соотношений колебательных компонент.

5. На базе полученных характеристик и оценок степени их информативности формирование исходного пространства признаков для рассматриваемой диагностической задачи. Построение в этом пространстве диагностических решающих правил с использованием алгоритмов классификационного анализа данных и распознавания образов [5].

3. Алгоритмы выделения элементов пульсового сигнала

Начальным этапом выделения функционально-значимых элементов сигнала является разбиение пульсового сигнала на квазипериоды. Сложность задачи автоматического выделения отдельных квазипериодов обусловлена такими причинами, как значительная вариабельность сигнала, наличие артефактов, наличие локальных экстремумов, большое разнообразие форм и типов сигнала и т.п. В качестве базовых параметров исходного сигнала используются амплитудные (VA, VC, VG) и временные (PA, PC, PG) параметры характерных элементов единичного квазипериода (рис. 2).

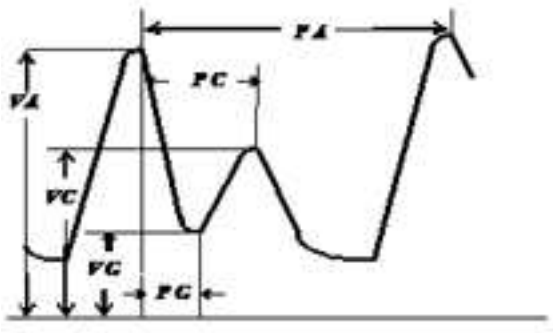


Рис. 2. Параметры характерных элементов единичного квазипериода

Для автоматического выделения основной и дополнительных волн единичного квазипериода был разработан комплексный алгоритм автоматической классификации [5], базирующийся на методах классификационного анализа данных [7].

Схему работы этих алгоритмов поясним на примере выделения основного квазипериода РА. На анализируемой записи сигнала выделяются все максимумы, то есть формируется выборка значений амплитуд типа VA и VC для всего сигнала. Затем с помощью одномерного варианта алгоритма m-локальной оптимизации [8] строится оптимальная классификация этой выборки на 3-5 классов. Обычно разбиение производится на 3 класса (малые, средние и большие значения амплитуды). Самый правый на оси значений класс (большие значения амплитуды) в большинстве случаев и будет соответствовать максимумам основного квазипериода для большей части сигнала.

Далее по стандартной схеме на реализации сигнала выделяются максимумы, попавшие в крайне правый класс, тогда отрезки сигнала между смежными выделенными максимумами и являются претендентами на искомые квазипериоды. К сожалению, на реальных пульсограммах часто наблюдаются существенные колебания значений амплитуд основных квазипериодов.

В связи с этим далее анализируется распределение выявленных зубцов на временной шкале. Если расстояние между соседними зубцами более $T_c K_a$, где T_c – средняя (типичная) длительность периода, а K_a – коэффициент аритмии $(1.5 \div 2.5)$, то для этой непериодизированной области выполняется коррекция. Процедура коррекции организована итеративным путем. А именно, в непериодизированной области ищется зубец с максимальной амплитудой и его включают в перечень основных зубцов. Затем вновь анализируется распределение зубцов на временной шкале и т.д. Остаётся проблема, связанная с неопределенностью коэффициента аритмии K_a. Впрочем, как показали тесты, влияние выбора величины этого коэффициента на качество периодизации сказывается лишь в экзотических случаях.

Предложенный алгоритм периодизации выделил практически все периоды на исследуемой (более 300 сигналов) выборке пульсограмм.

По аналогичной схеме находятся другие периодические составляющие сигнала в рамках основного квазипериода (в общем случае анализируются распределения не только максимумов, но и минимумов).

В процессе автоматического выделения основных и дополнительных волн осуществляется оценка их амплитудных и временных значений. Затем полученные значения базовых параметров используются для формирования динамических рядов, представляющих собой зависимость значений каждого параметра от номера периода.

Сформированные динамические ряды подвергаются в дальнейшем статистическому анализу. При этом предварительно осуществляется процедура проверки сформированных массивов на наличие выбросов, которые могут иметь место под воздействием артефактов и их коррекция в случае необходимости.

4. Анализ колебательных компонент динамических рядов

Динамические ряды, полученные на предыдущем этапе, рассматриваются как случайные процессы, представляющие собой суперпозицию колебаний, трендов и шумов. В работе рассматриваются только стационарные (точнее - квазистационарные) режимы функционирования организма. В связи с этим используются методы анализа стационарных случайных процессов. На первом этапе осуществляется оценка динамических рядов на стационарность и нормальность. Выявление периодических составляющих динамических рядов и их характеристик осуществляется с помощью спектрально-корреляционного анализа, прежде всего с помощью функции спектральной плотности. Пики спектральной плотности, как правило, были сосредоточены в частотных диапазонах *HF, LF, VLF*.

На исследуемом массиве данных (355 векторов) экспертным путём было выделено 7 типов спектральной плотности, отличающихся количеством и степенью выраженности указанных выше колебательных компонент.

Для исследуемого массива динамических рядов параметра PA распределение типов спектральной плотности было следующим: тип 1 - 41 %, тип 2 – 17 %, тип 3 – 13.5 %, тип 4 – 7%, тип 5 – 9 %, тип 6 – 8 %, тип 7 – 4.5 %.

Аналогичные типы спектральной плотности были выделены и для динамических рядов для других базовых параметров пульсовой волны (*PC, PG, VA, VC, VG*). Проведенные исследования показали, что для пульсового сигнала спектральные характеристики различных динамических рядов не идентичны. При этом частотные параметры колебательных компонент совпадают, а амплитудные могут существенно отличаться.

Эти результаты создают предпосылки для увеличения объёма диагностической информации при комплексном использовании спектральных характеристик различных элементов исходного сигнала.

Для получения объективных данных о типах спектральной плотности была проведена формализованная (автоматическая) классификация спектральных кривых по следующим параметрам-характеристикам спектральной кривой: *VLF/ LF*, *LF/ HF* и *HF/ VLF*. Сравнительный анализ экспертной и формализованной классификаций показал их достаточно высокую согласованность. Более того, во многих спорных случаях эксперты соглашались изменить принадлежность на тот тип, к которому их отнёс формальный алгоритм. Этот очень важный результат позволяет в дальнейшем в значительной степени автоматизировать процесс диагностики.

Было построено трёхмерное распределение указанных параметров для различных типов спектров. Рассмотрение таких распределений позволило выявить ориентировочные области расположения различных типов спектральной плотности в координатах рассматриваемых параметров. На основании имеющихся экспериментальных данных, отражающих диагнозы исследуемых векторов, получено распределение диагнозов в соответствии с выявленной типологией спектров.

Полученные результаты показывают, что, используя лишь агрегированные данные о некоторых типах спектральной плотности можно делать достаточно интересные вероятностные оценки наличия или отсутствия артериальной гипертензии, а

именно – для спектров типа 2 можно с вероятностью порядка 0.8, а для спектров типа 6 – с вероятностью порядка 0.75 говорить об отсутствии артериальной гипертензии. Правда остальные типы спектров не имеют столь выраженной зависимости от наличия или отсутствия артериальной гипертензии, здесь требуется привлечение других методов анализа, например, алгоритмов распознавания образов.

5. Оценка диагностической значимости параметров пульсового сигнала

На основе экспериментальных данных, полученных в ходе клинических обследований по выявлению ранней стадии артериальной гипертензии был проведен сравнительный анализ диагностической значимости параметров спектральной плотности динамических рядов, сформированных для различных элементов пульсового сигнала, в том числе: длительности квазипериодов PA, времени достижения максимума дикротической волны PC, времени подъема дикротической волны PD, времени падения основной волны PG.

Как уже говорилось выше выше, весь массив данных (355 векторов) был разбит на два класса: 1 класс – первичная артериальная гипертензия (165 векторов) и 2 класс – различные виды психосоматической функциональной патологии с нормальным артериальным давлением (190 векторов).

В качестве массива показателей ритмической структуры пульсового сигнала исследовались следующие характеристики : спектральные мощности трех основных колебательных компонент VLF, LF, HF для динамических рядов параметров PA, PC, PD, PG; взаимные соотношения спектральных компонент VLF, LF, HF для всех рассматриваемых динамических рядов; взаимное соотношение однотипных спектральных компонент для динамических рядов разных параметров.

Осуществлялась оценка информативности показателей для некоторой фиксированной классификации исследуемых объектов (например, по диагнозу). Степень информативности I определялась как «вес» каждого показателя в уменьшении ошибки классификации при последовательном присоединении параметров к набору параметров, в пространстве которых проводится классификация (схема аналогичная пошаговой регрессии).

Результаты расчётов позволили для данной задачи выявить 5 наиболее информативных показателей, характеризующих ритмическую структуру пульсового сигнала. А именно: $HF/\ LF$ для параметра PC ($I = 0.54$); LF для параметра PC ($I = 0.41$); VLF для параметра PG ($I = 0.25$); отношение HF для PC к HF для PA ($I = 0.22$); отношение HF для PC к HF для PG ($I = 0.18$)

Таким образом, для данной диагностической задачи наиболее информативным является отношение спектральных мощностей дыхательной волны и медленной волны для параметра PC (длительность дикротической волны). Для сравнения была оценена информативность аналогичного показателя для параметра PA (длительность основного периода). Её значение составляло лишь $I = 0.01$.

Полученные результаты свидетельствуют о том, что использование значений спектральной плотности для параметра PC, определяемого временем достижения

дикротической волны, даёт существенно лучшие результаты по информативности, чем при использовании традиционно используемых спектральных характеристик длительностей периодов. Это подтверждает важность исследования ритмической структуры биосигналов, обусловленной совокупностью колебательных компонент различных функционально-значимых элементов в пределах основного квазипериода.

Заключение

Проведенные исследования позволили провести сравнительный анализ информативности ряда показателей ритмической структуры пульсового сигнала, обусловленных динамикой функционально-значимых элементов сигнала. В частности, было показано, что временные показатели дикротической волны существенно информативнее для рассматриваемой диагностической задачи, чем традиционно используемые колебательные компоненты длительностей квазипериодов.

Получены количественные оценки информативности ряда показателей ритмической структуры пульсового сигнала.

Оценена диагностическая значимость спектральной плотности динамического ряда длительностей квазипериодов применительно к задаче выявления ранней стадии артериальной гипертензии в детском возрасте. Выявлены спектральные типы, характеризующие отсутствие артериальной гипертензии в детском возрасте с вероятностью порядка 0.75–0.8.

Список литературы

1. Heart rate variability, standarts of measurement, physiological interpretation and clinical use. Task Force of the European Society of Cardiology and the North Amer. Society of Pacing and Electrophysiology. Europ. Heart Journal. 1996. v. 17. № 3. pp. 354 –381.
2. *Миронова Т.Ф., Миронов В.А.* Клинический анализ волновой структуры синусового ритма сердца. Челябинск, 1998. – 162 с.
3. *Баевский Р.М.* Анализ вариабельности сердечного ритма: история и философия, теория и практика. Клиническая информация и телемедицина, 2004. № 1(1). – с. 54-64.
4. *Волхонская Т.А.* Исследование спектральных характеристик амплитудно-временных параметров электрокардиограммы. Кибернетика и вычислительная техника, 1989, вып. 82, – с. 25-27.
5. *Брязгунов И.П., Десова А.А., Кизева А.Г.* Исследование характеристик формы и ритмической структуры пульсового сигнала лучевой артерии при артериальной гипертензии в детском и подростковом возрасте. // Физиология человека. 2007. том 23. № 3. – с. 38 –43.
6. *Дорофеюк Ю.А.* Комплексный алгоритм автоматической классификации и его применение для анализа и принятия решений в больших системах управления. / Теория активных систем. Труды международной научно-практической конференции. / – М.: ИПУ РАН. 2007. – с. 39-42.
7. *Бауман Е.В., Дорофеюк А.А.* Классификационный анализ данных. // Труды Международной конференции по проблемам управления. Том 1. – М.: СИНТЕГ, 1999. – с. 62-77.
8. *Десова А.А., Дорофеюк А.А., Гучук В.В., Дорофеюк Ю.А., Покровская И.В.* Процедуры классификационного анализа в задаче формирования информативных признаков при исследовании ритмической структуры биосигнала / Автоматика и телемеханика. 2008, №6. –с. 143-152.

Статья поступила в редакцию 27.04.2008

УДК 62-50

МЕТОДОЛОГИЯ ЭКСПЕРТНО-КЛАССИФИКАЦИОННОГО АНАЛИЗА ДАННЫХ В ЗАДАЧАХ АНАЛИЗА РАЗВИТИЯ РЕГИОНАЛЬНЫХ СИСТЕМ[1]

© Дорофеюк А.А., Дорофеюк Ю.А., Покровская И.В.

Институт проблем управления РАН

Abstract. The expert-ranging methods of regional socio-economic systems functioning analysis are discussed. As an example the utilization of especially developed technique for constituent entity RF social development comparative evaluation is described.

Введение

Анализ функционирования крупномасштабных региональных систем, каждый из которых характеризуется достаточно большим набором показателей является достаточно сложной задачей даже при использовании современных информационно-компьютерных технологий. Эти сложности связаны в первую очередь с большой размерностью задачи: число объектов колеблется от нескольких десятков до нескольких тысяч, число показателей – в пределах нескольких десятков, число моментов времени (при анализе динамики развития) – может достигать многих десятков (при анализе помесячной динамики).

До сих пор во многих экономических исследованиях для анализа и прогнозирования развития региональных систем используются стандартные статистические методы. В большинстве из них исследуемое множество объектов рассматривается как выборка из некоторой генеральной совокупности. Тогда задача заключается в том, чтобы оценить статистические свойства всей генеральной совокупности по статистическим характеристикам и свойствам этой выборки. Однако, для многих прикладных задач вероятностная интерпретация результатов невозможна, да и сама задача не укладывается в рамки классической математической статистики. По этой причине за последние 20-30 лет появилось достаточно много работ, в которых статистический подход в его классическом виде претерпел существенные изменения (см., например, [1]). Примером такой задачи как раз и является рассматриваемая в работе задача анализа социального развития регионов Российской Федерации. Здесь статистическое оценивание играет вспомогательную роль, а главной задачей становится построение «сжатого описания» исходных данных (например, рейтингов регионов), которое можно было бы использовать для принятия качественных управленческих решений, например, распределения финансовой помощи регионам.

[1]Работа выполнена при частичной финансовой поддержке РФФИ, проекты 08-07-00349-а, 08-07-00427-а.

1. Методы экспертно-классификационного анализа в задаче анализа развития регионов

Для получения полного представления о функционировании крупномасштабной региональной системе (в данном случае – об уровне социального развития регионов) предлагается использовать методы классификационного анализа сложноорганизованных данных [2]. Была предложена концепция применения методов классификационного анализа для решения задач анализа развития региональных систем управления. Основная цель этой концепции состоит в следующем:

- структуризация исходного набора показателей с целью выявления относительно небольшого числа информативных параметров;
- структуризация исходного множества объектов, для чего необходимо выделить в пространстве выбранных информативных параметров области близко расположенных друг к другу объектов;
- анализ динамических свойств системы, в том числе выделение характерных траекторий изменения в пространстве параметров положения объектов во времени (типология траекторий), выявление зависимостей между параметрами с учетом временного сдвига и т.д.

Структуризация исходного набора параметров. Опыт использования алгоритмов классификационного анализа показывает, что классификация по всем исходным параметрам далеко не всегда приводит к желаемым результатам, например, из-за наличия помех и ошибок, негативного влияния малоинформативных и шумящих параметров. Поэтому классификацию обычно проводят в пространстве так называемых информативных параметров, которое имеет значительно меньшую размерность, чем исходное.

Структуризацию параметров предлагается проводить методами экстремальной группировки [1], при этом необходимо выбрать число групп, а также тип группировки – с фоновой группой или без неё, в зависимости от уровня «зашумлённости» параметров. Для этой цели используются специальные экспертные процедуры. Результатом экстремальной группировки являются группы параметров и факторы – интегральные параметры-характеристики групп, каждый из которых является линейной комбинацией параметров соответствующей группы и, в определённом смысле, её центром. На базе результатов экстремальной группировки выбираются информативные показатели для исследуемой системы. В качестве таковых выбираются либо сами факторы (интегральные показатели), либо исходные параметры, ближайшие к этим факторам. Основное условие – они должны быть легко интерпретируемы.

Для удобства использования интегральных показателей по каждому из них обычно делается одномерная классификация объектов. Благодаря этому интегральный показатель легко преобразовать в параметр, измеряемый в качественной шкале, например, его значения можно характеризовать в таких терминах, как «низкие», «средние» и «высокие». Для этой цели в работе используется алгоритм одномерной m-локальной оптимизации, разработанный в [3].

Структуризация исходного множества объектов. Выявление структуры объектов производится в пространстве информативных параметров X. Для этой

цели используются алгоритмы автоматической классификации [3], применение которых подразумевает выбор:

- вида критерия качества;
- классификации с фоновым классом или без него, т.е. отбрасываются ли «далёкие» (шумящие) объекты;
- типа размытости – четкая, размытая, с размытыми границами, четкая с размытым фоном, размытая с четким фоном и т.д.

Результатом классификации является вектор функций принадлежности объектов к классам $H(x) = (h_1(x), \ldots, h_r(x)),\ x \in X$ и описание самих классов (например, эталонов) [2]. Для того чтобы результаты классификации можно было использовать в практических задачах, важно не только насколько экономно она представляет исходную информацию, но и насколько эта классификация удобна для интерпретации в содержательных терминах. В этой связи в приложениях часто используются алгоритмы построения так называемых «хорошо интерпретируемых классификаций» [4].

Выявление динамических свойств исследуемой системы. Для этой цели в работе использовался алгоритм структурного прогнозирования развития сложных динамических объектов, разработанный в [5]. Основная идея предложенного метода решения этой задачи состоит в том, что требуется прогнозировать не точные значения параметров, описывающих состояние объекта, а лишь класс (тип) объекта в рамках некоторой структуры изучаемого множества объектов.

Предполагается, что вектор значений параметров $x_j(t) = (x_j^{(1)}(t),\ x_j^{(2)}(t),\ \ldots,\ x_j^{(k)}(t))$ достаточно полно характеризует состояние j-го объекта в момент времени t. А это, в свою очередь означает, что взаиморасположение множества точек $x_1(t),\ \ldots,x_n(t)$ в k-мерном пространстве признаков X отражает реальную структуру (типологию) исследуемого множества объектов. Для выявления этой структуры в работе используется комплексный алгоритм автоматической классификации [3]. С этой целью в момент времени t_1 производится автоматическая классификация (кластеризация) т точек в пространстве X на небольшое (3-5) число классов r, каждый из которых и характеризует тип объекта (в пределах изучаемого множества). Вводится понятие модели (эталона) класса $a_i(t),\ i = 1, ..., r$ (чаще всего – это центр класса) [2]. Для каждой точки кроме принадлежности к классу вычисляются расстояния до эталонов всех классов $R_{ij}(t),\quad i = 1\,, ..., r,\ j = 1\,, ..., n.$

Вопрос содержательной интерпретации полученных классов (типов) решается с помощью специальных экспертных процедур.

В момент времени t_2 и во все последующие моменты времени, для которых известны значения соответствующих параметров, каждая точка относится к тому или иному классу с помощью одного из алгоритмов распознавания образов с учителем. В работе для этой цели использовался алгоритм метода потенциальных функций, который в спрямляющем пространстве эквивалентен алгоритму ближайшего среднего [1]. На каждом шаге, после того, как определена принадлежность всех точек к тому или иному классу, производится пересчет эталонов и для каждой точки с предыдущего шага пересчитываются, а для каждой новой точки вычисляются расстояния до новых эталонов. Такая процедура выполняется для всех m моментов времени. В итоге

для каждого объекта получается последовательность (траектория) из n позиций. В каждой позиции находится $r+1$ число, первое из которых – это номер класса, к которому относился этот объект в соответствующий момент времени, а последующие числа – это значения расстояний до центров классов в тот же момент времени. Требуется спрогнозировать номер класса (тип объекта), к которому будет относиться каждый объект в момент времени t_{m+1}.

В качестве прогнозной модели для каждого объекта в этом алгоритме используется марковская цепь с r состояниями и матрицей переходных вероятностей $P_j = \|p_{ji}\|$. С помощью специального алгоритма на каждом шаге для каждого периода времени (года) производится пересчёт соответствующих переходных вероятностей p_{ji} с использованием информации о значениях расстояний от каждого объекта до центров классов [5]. Этот алгоритм позволяет учесть особенности всей прошлой траектории изменения положения каждого объекта в пространстве информативных параметров.

2. Сравнительный анализ социального развития субъектов РФ

Разработанная методика использовалась для сравнительного анализа социального развития субъектов Российской Федерации (47 показателей для 79 регионов за 3 года). Применение алгоритма экстремальной группировки позволило разбить 47 исходных показателей на 6 групп: доходы населения (13 показателей); расходы и сбережения (14 показателей); потребление продуктов питания (7 показателей); демографические характеристики (4 показателя); характеристики социальной напряженности (6 показателей); объем финансовой помощи из межрегиональных фондов (2 показателя). Для последующей классификации и формирования рейтинга регионов было отобрано шесть показателей, ближайших к факторам групп: среднедушевой доход, доля оплаты труда в среднедушевом доходе, превышение доходов над расходами, число пенсионеров на 1000 чел. населения, уровень безработицы, общий объем финансовой помощи (ООФП) на душу населения.

В результате классификационного анализа в шестимерном пространстве этих показателей было получено 7 классов регионов (с использованием процедуры экспертной коррекции). В зависимости от целей исследования классифицируются либо все регионы за все три года (каждый регион в каждом году рассматривается как самостоятельный объект), либо все регионы отдельно для каждого года. Классификация первого типа позволяет анализировать динамику развития регионов, прослеживая на протяжении исследуемого периода переходы регионов из одного класса в другой, или, что то же самое, изменение рейтинга региона. При составлении текущего рейтинга удобнее использовать классификацию второго типа, поскольку в этом случае результаты получаются более обозримыми. На базе этой классификации получен рейтинг социального развития регионов для каждого из 3 лет, проанализирована динамика рейтингов регионов по основным показателям, представленная в табл. 1.

Для укрупненной оценки социально-экономической ситуации в регионе наибольший интерес представляет пара показателей: «Среднедушевой доход» и «Уровень безработицы». По этой паре показателей по данным за последний из трёх анализируемых лет был построен рейтинг регионов.

Таблица 1. Динамика рейтингов регионов по основным показателям

Показатели	**Изменение рейтинга за 2 года (количество регионов)**			
	+2	**+1**	**0**	**- 1**
Среднедушевой доход	2	48	29	-
Доля оплаты труда в среднедушевом доходе	6	37	33	3
Превышение доходов над расходами	-	6	55	18
Число пенсионеров на 1000 чел. населения	-	9	70	-
Уровень безработицы	8	44	24	3
Общий объем финансовой помощи	-	31	48	2

Для этого с помощью алгоритма хорошо интерпретируемых классификаций [4] было построено 16 классов.

При построении рейтингов по двум и более показателям одновременно возникает проблема многокритериальности: как упорядочить два объекта, один из которых имеет более высокий рейтинг по одному показателю, а второй – по другому показателю. В общем случае в таких ситуациях приходится прибегать к мнению экспертов, оценивающих сравнительную важность приращений значений показателей. В данном случае с помощью экспертной процедуры было сформулировано следующее простое правило упорядочения в конфликтных ситуациях: при низких значениях уровня безработицы упорядочение производится по среднедушевому доходу; при низких значениях среднедушевого дохода упорядочение ведется по уровню безработицы. Тогда классы регионов упорядочиваются так, как показано в табл. 2.

Кроме того, каждый рейтинговый класс получил интегральную характеристику по всем основным показателям.

На этом же материале использовался алгоритм структурного прогнозирования социального развития субъектов Российской Федерации, разработанный в [5]. Как уже говорилось выше, в качестве прогнозной модели для каждого объекта в этом алгоритме используется марковская цепь с r состояниями и матрицей переходных вероятностей $P_j = \|p_{ji}\|$. Результаты использования полученной прогнозной модели, не смотря на малый период наблюдений (всего три года), был оценён экспертами Минздравсоцразвития как очень хорошие. Эти результаты продемонстрировали высокую эффективность использованных алгоритмов при выборе информативных показателей, «оптимального» числа классов, а также прогнозной модели.

Заключение

Описана методика сравнительного анализа развития региональных социально-экономических системы, которые характеризуются достаточно большим набором

Таблица 2. Рейтинг регионов по показателям «Среднедушевой доход» и «Уровень безработицы»

R	**П1**	**П2**	**Регионы**
1	1	1	Москва
2	2	1	Чукотский АО
3	2	2	Тюменская область
4	3	2	Респ. Коми, Респ. Саха (Якутия), Магаданская обл., Сахалинская обл.
5	3	3	Мурманская область, Камчатская область
6	4	1	Московская обл., Ярославская обл., Вологодская обл., Санкт-Петербург, Респ. Татарстан, Самарская обл.
7	4	2	Челябинская обл., Респ. Карелия, Архангельская обл., Волгоградская обл., Респ. Башкортостан, Нижегородская обл., Пермская обл., Свердловская обл., Респ. Хакасия, Красноярский край, Кемеровская обл., Омская обл., Приморский край, Хабаровский край, Еврейская автономная обл.
8	4	3	Астраханская обл., Ростовская обл., Иркутская обл., Томская обл.
9	5	1	Костромская обл., Липецкая обл., Тверская обл., Тульская обл., Новгородская обл., Ульяновская обл.
10	5	2	Белгородская обл., Брянская обл., Воронежская обл., Калужская обл., Курская обл., Орловская обл., Рязанская обл., Тамбовская обл., Калининградская обл., Ленинградская обл., Псковская обл., Краснодарский край, Ставропольский край, Респ. Мордовия, Удмуртская Респ., Чувашская Респ., Кировская обл., Пензенская обл., Саратовская обл., Алтайский край
11	6	2	Ивановская область
12	5	3	Владимирская обл., Смоленская обл., Респ. Адыгея, Карачаево-Черкесская Респ., Респ. Северная Осетия-Алания, Оренбургская обл., Курганская обл., Респ. Алтай, Новосибирская обл., Читинская обл., Амурская область
13	5	4	Республика Бурятия
14	5	3	Республика Марий-Эл
15	5	5	Кабардино-Балкарская Республика, Республика Тыва
16	6	5	Республика Дагестан, Республика Ингушетия, Республика Калмыкия

показателей. Основу методики составляют алгоритмы структуризации набора исходных параметров и множества исследуемых объектов. Рассмотрены результаты использования методики для решения задачи сравнительного анализа социального развития регионов России за 3 года.

Список литературы

1. *Браверман Э.М., Мучник И.Б.* Структурные методы обработки эмпирических данных. – М.: Наука, 1983. – 464 с.
2. *Бауман Е.В., Дорофеюк А.А.* Классификационный анализ данных. // Труды Международной конференции по проблемам управления. Том 1. – М.: СИНТЕГ, 1999. – С. 62-77.
3. *Дорофеюк Ю.А.* Комплексный алгоритм автоматической классификации и его применение для анализа и принятия решений в больших системах управления. / Теория активных систем. Труды международной научно-практической конференции. / –М.: ИПУ РАН. 2007. – С. 39-42.
4. *Дорофеюк А.А., Чернявский А.Л.* Алгоритмы построения хорошо интерпретируемых классификаций. / Проблемы управления. №2, 2007. – С. 83-84.
5. *Дорофеюк Ю.А., Дорофеюк А.А.* Методы структурно-классификационного прогнозирования многомерных динамических объектов. Искусственный интеллект, № 2, 2006. С. 138-141

Статья поступила в редакцию 27.04.2008

УДК 62-50

СТРУКТУРНО-КЛАССИФИКАЦИОННЫЕ МЕТОДЫ АНАЛИЗА И ПРОГНОЗИРОВАНИЯ В СИСТЕМАХ УПРАВЛЕНИЯ[1]

© Дорофеюк Ю.А.

Институт проблем управления РАН

Abstract. The analysis and forecasting within the loosely-formalized multivariate control system, consisting of sufficiently large number of a priori non-structured objects, problem solution method is proposed. As a forecasting model for each object the markovian chain with $r$states, where r – the number of structural units (classes), is used.

Введение

Рассматривается задача анализа и прогнозирования в слабоформализованной многопараметрической системе управления, которая состоит из достаточно большого числа формально не структурированных объектов. Идея предлагаемого метода решения этой задачи состоит в том, что исследуются не точные значения параметров, описывающих состояние каждого объекта (траектории состояний), а лишь класс, к которому принадлежит каждый объект в рамках некоторой структуры множества объектов, входящих в исследуемую систему [1]. Такое интегральное описание объектов позволяет существенно повысить эффективность результатов принимаемых управленческих решений и прогнозов. Для формализации задачи используется методология классификационного анализа [2].

1. Постановка задачи

Пусть исследуемая система состоит из n объектов, каждый из которых характеризуется набором из k параметров. Изучается поведение этого множества объектов в дискретные моменты времени. Вводится в рассмотрение k-мерное пространство параметров X, в котором j-й объект в момент времени t представляется точкой $x_j(t) = (x_j^{(1)}(t),\ x_j^{(2)}(t),\ \ldots\ ,\ x_j^{(k)}(t))$. Упорядоченная совокупность точек $x_j(t_1),\ \ldots\ , x_j(t_m)$ является известной частью траектории, характеризующей динамику j-го объекта.

В большинстве приложений для принятия управленческого решения в момент времени t_m используется совокупная информация об известных траекториях каждого объекта и прогноз значений $x_j(t_m + 1),\ j = 1, \ldots, n$. При этом, как правило, информация по каждому объекту рассматривается независимо от остальных [3]. Однако для многих прикладных задач требуется знать не точные значения параметров-характеристик в моменты времени $t_1,\ t_2,\ \ldots\ , t_m$ и прогнозировать значения в момент t_{m+1}, а знать (и прогнозировать) лишь класс, к которому принадлежит (будет принадлежать) этот объект в соответствующие моменты времени в рамках некоторой структуры (классификации) множества объектов изучаемой системы. Так, например, в процессе исследования социально-экономического развития субъектов РФ

[1]Работа выполнена при частичной финансовой поддержке РФФИ, проект 08-07-00349-а.

(в данном случае крупномасштабная система – народное хозяйство РФ) вовсе необязательно знать (и прогнозировать) значения социально-экономических параметров для каждого региона, достаточно лишь знать, в какой класс этот регион попадает в данный и прогнозируемый моменты времени (условно, в классы «хорошо», «средне» и «плохо» развивающихся регионов).

Основу предлагаемого подхода составляет процедура выявления структуры объектов, входящих в исследуемую систему. Предполагается, что вектор значений параметров $x_j(t)$ достаточно полно характеризует состояние j-го объекта в момент времени t. А это, в свою очередь, означает, что взаиморасположение точек $x_1(t)$, ... , $x_n(t)$ в пространстве X отражает реальную структуру (типологию) исследуемого множества объектов. Для выявления такой структуры использовался комплексный алгоритм автоматической классификации, специально разработанный для решения таких задач [4].

2. Динамическая структуризация исследуемых объектов

Вначале (в момент времени t_1) с помощью комплексного алгоритма автоматической классификации [4] производится структуризация n точек в пространстве X на r классов, каждый из которых и характеризует определённый тип объекта. Число классов r выбирается с помощью человеко-машинной процедуры, входящей в комплексный алгоритм автоматической классификации. Вводится понятие модели (эталона) класса $a_i(t)$, $i = 1, ..., r$ (чаще всего – это центр класса) [2]. Для каждого объекта кроме принадлежности к классу вычисляются расстояния до эталонов всех классов $R_{ij}(t)$, $\quad i = 1\,, ..., r;\ \ j = 1\,, ..., n.$

Заметим, что на практике структуризация объектов чрезвычайно редко проводится в пространстве исходных признаков, обычно сначала производится выделения набора информативных параметров. В настоящей работе для этой цели использовался алгоритм экстремальной группировки параметров «квадрат» [5]. В результате его применения получают разбиение исходных k параметров на небольшое (заданное) число групп, а также значения факторов для этих групп. В приложениях используются либо новые интегральные параметры – факторы групп, либо набор параметров, каждый из которых является ближайшим к фактору в соответствующей группе.

В большинстве приложений исходные или выделенные информативные параметры имеют неравнозначную важность для определения структуры объектов. Для выявления таких показателей важности в работе предлагается использовать процедуры экспертного оценивания. В результате экспертизы каждый параметр получает определённый вес (показатель «важности» этого параметра) для формирования структуры объектов.

В момент времени t_2 каждая точка $x_j(t_2)$ с помощью одного из алгоритмов распознавания образов с учителем относится к тому или иному классу в рамках классификации, полученной на первом шаге. В работе для этого используется алгоритм метода потенциальных функций, который в спрямляющем пространстве эквивалентен алгоритму ближайшего среднего [6]. А именно, каждая точка $x_j(t_2)$ относится к классу A_l,

для которого заданная мера близости $K\left(x_j\left(t_2\right), A_l\right)$ точки $x_j(t_2)$ к этому классу максимальна, то есть: $K\left(x_j\left(t_2\right), A_l\right) = \max\limits_i K\left(x_j\left(t_2\right), A_i\right), \quad i = 1, ..., r, \;\; j = 1, ..., n$. В качестве такой меры близости используется величина $K\left(x_j, A_l\right) = \frac{1}{n_l} \sum\limits_{x_i \in A_l} K\left(x_j, x_i\right)$, где n_l – число точек в классе A_l, $K(x, y)$ – потенциальная функция. В работе при решении прикладных задач для потенциальной функции использовалось выражение: $K(x, y) = 1/\{1 + \alpha\, R^p(x, y)\}$, где $R(x, y)$ – евклидово расстояние между точками x и y в пространстве X, α и p – настраиваемые параметры алгоритма.

После того, как определена принадлежность всех точек к тому или иному классу, производится пересчёт эталонов $a_i(t_2), \;\; i = 1, ..., r$. Для каждой точки с предыдущего шага пересчитываются, а для каждой новой точки вычисляются расстояния до новых эталонов $R\left(x_j\left(t_2\right), a_i\left(t_2\right)\right), \qquad i = 1, ..., r, \;\; j = 1, ..., n$. Такая процедура выполняется для всех m моментов времени. В итоге для каждого объекта получается последовательность (траектория) из mпозиций. В каждой позиции находится $(r + 1)$ число, первое из которых – это номер класса, к которому относился этот объект в соответствующий момент времени, а последующие числа – это значения расстояний до центров классов в тот же момент времени. Требуется спрогнозировать номер класса (тип объекта), к которому будет относиться каждый объект в момент времени t_{m+1}.

3. Алгоритм прогнозирования

В качестве прогнозной модели для каждого объекта используется марковская цепь с r состояниями, то есть на каждом шаге рассчитываются элементы матрицы переходных вероятностей $\mathrm{P}=\|p_{ji}\|, \;\; j = 1, \ldots, n; \;\; i = 1, \ldots, r$. Разработан специальный алгоритм пересчёта на каждом шаге соответствующих переходных вероятностей p_{ji} с использованием информации о значениях расстояний до центров классов и условия нормировки $\sum\limits_{i=1}^{r} p_{ji} = 1$ для всех $j = 1, ..., n$. Алгоритм работает следующим образом. Пусть после первого шага, для точек $x_j(t_1)$ подсчитаны расстояния до эталонов $R_{ji}^{(1)} = R(x_j\left(t_1\right), a_i\left(t_1\right)), \quad i = 1, ..., r, \;\; j = 1, ..., \mathrm{n}$. Тогда, элементы матрицы переходных вероятностей $p_{ji}^{(1)} = p_{ji}\left(t_1\right)$ рассчитываются следующим образом:

$$p_{ji}^{(1)} = \frac{\alpha_j^{(1)}}{R_{ji}^{(1)}}, \tag{1}$$

где нормирующий множитель $\alpha_j^{(1)}$ определяется выражением:

$$\alpha_j^{(1)} = \frac{\prod\limits_{i=1}^{r} R_{ji}^{(1)}}{\sum\limits_{l=1}^{r} \frac{1}{R_{jl}^{(1)}} \prod\limits_{i=1}^{r} R_{ji}^{(1)}}.$$

На s-ом шаге элементы матрицы переходных вероятностей (1) модифицируется при помощи следующей процедуры. Введем обозначения: $\Delta R_{ji}^{(s)} = R_{ji}^{(s-1)} - R_{ji}^{(s)}$;

$\Delta\hat{R}_{ji}^{(s)} = \frac{R_{ji}^{(s-1)} - R_{ji}^{(s)}}{R_{ji}^{(s-1)} + R_{ji}^{(s)}}$. Если j-ая точка совпадает с эталоном i_0-го класса $(x_j(t_s) = a_{i_0}(t_s))$, т.е. $R_{ji_0}^{(s)} = 0$, то $p_{ji}^{(s)} = \begin{cases} 1, & \text{если } i = i_0, \\ 0, & i = 1,..,r,\ i \neq i_0 \end{cases}$. Другими словами, если точка совпадает с эталоном некоторого класса, то вероятность для этой точки остаться в этом классе равна 1, а вероятность перехода в другой класс равна 0.

Для случая, когда $R_{ji_0}^{(s)} \neq 0$, происходит модификация всех переходных вероятностей по следующей схеме:

$$p_{ji}^{(s)} = \gamma \left[p_{ji}^{(s-1)} + \left(\frac{1 + sign(\Delta R_{ji}^{(s)})}{2} - p_{ji}^{(s-1)} sign(\Delta R_{ji}^{(s)}) \right) \Delta\hat{R}_{ji}^{(s)} \right] , \qquad (2)$$

где, как обычно: $sign(z) = \begin{cases} 1, & \text{если } z \geq 0, \\ -1, & \text{если } z < 0 \end{cases}$, а γ- нормирующий множитель, определяемый условием нормировки переходных вероятностей $\sum_{i=1}^{r} p_{ji}^{s} = 1$:

$$\gamma = \frac{1}{1 + \left(\frac{1+sign(\Delta R_{ji}^{(s)})}{2} - p_{ji}^{(s-1)} sign(\Delta R_{ji}^{(s)}) \right) \Delta\hat{R}_{ji}^{(s)}}.$$

Введение в (2) величины $sign(\Delta R_{ji}^{(s)})$ вызвано необходимостью производить различными способами модификацию переходных вероятностей для случаев увеличения и уменьшения расстояния от точки $x_j(t_s)$ до эталонов классов $a_i(t_s)$ на s-ом шаге. А именно: в случае уменьшения величины $R_{ji}^{(s)}$ по отношению к $R_{ji}^{(s-1)}$ (т.е. $\Delta R_{ji}^{(s)} < 0$) изменение соответствующей переходной вероятности происходит за счёт её увеличения на некоторую долю от $(1 - p_{ji}^{(s-1)})$; а в случае увеличения величины $R_{ji}^{(s)}$ по отношению к $R_{ji}^{(s-1)}$ (т.е. $\Delta R_{ji}^{(s)} > 0$) изменение соответствующей переходной вероятности происходит за счёт её уменьшения на некоторую долю от $p_{ji}^{(s-1)}$. Это необходимо для выполнения условий нормировки для переходных вероятностей $0 < p_{ji}^{(s)} < 1,\ i = 1, ..., r$.

Построенная при помощи описанного выше алгоритма матрица переходных вероятностей $\boldsymbol{P}$ используется для прогнозирования принадлежности объекта тому или иному классу. На практике обычно используется не рандомизированная, а байесовская схема, когда объект относится к тому классу i_0, для которого $p_{ji_0} = \max\limits_{i=1,...,r} p_{ji}$. В случае равенства переходных вероятностей p_{ji} для прогнозируемого объекта для двух или нескольких классов, он относится к классу с наименьшим номером.

4. Модификации

Разработана модификация процедуры прогнозирования, когда классификация объектов задаётся заранее (например, экспертным путём) и в последующем остаётся неизменной.

Разработан также вариант алгоритма «с памятью», когда используются данные только об s прошлых состояниях множества объектов (s – глубина памяти алгоритма).

Оказалось, что для некоторых приложений (с достаточно высоким уровнем помех при измерении параметров) существенно более эффективным оказывается использование алгоритмов размытой классификации, в том числе с фоновым классом [2].

Заключение

Разработанная методология использовалась при анализе и совершенствовании процедур принятия решений для нескольких больших систем управления, в основном регионального характера, в том числе – региональная система управления здравоохранением, пассажирскими автоперевозками, система анализа, управления и прогнозирования социально-экономического развития субъектов РФ и др. Во всех приложениях, а также при машинном моделировании была подтверждена высокая эффективность разработанной методики структурно-классификационного анализа и прогнозирования.

Список литературы

1. *Дорофеюк А.А., Дорофеюк Ю.А* Методы структурно-классификационного прогнозирования многомерных динамических объектов / Искусственный интеллект, № 2, 2006. – С.138-141.
2. *Бауман Е.В., Дорофеюк А.А.* Классификационный анализ данных / Труды Международной конференции по проблемам управления. Том 1. – М.: СИНТЕГ, 1999. – С. 62-67.
3. *Статистическое моделирование и прогнозирование.* Сборник под ред. Гранберга А.Г. . – М.: Финансы и статистика, 1990. – 382 с.
4. *Дорофеюк Ю.А.* Комплексный алгоритм автоматической классификации и его применение для анализа и принятия решений в больших системах управления. / Теория активных систем. Труды международной научно-практической конференции. / - М.: ИПУ РАН. 2007. – С. 39-42.
5. *Браверман Э.М., Мучник И.Б.* Структурные методы обработки эмпирических данных – М.: Наука, 1983.
6. *Айзерман М.А., Браверман Э.М., Розоноэр Л.И.* Метод потенциальных функций в теории обучения машин. М.: Наука. 1970.

Статья поступила в редакцию 27.04.2008

УДК 62-50

КОМПЛЕКСНЫЙ АЛГОРИТМ АВТОМАТИЧЕСКОЙ КЛАССИФИКАЦИИ И ЕГО ИСПОЛЬЗОВАНИЕ В ЗАДАЧАХ АНАЛИЗА И ПРИНЯТИЯ РЕШЕНИЙ[1]

© Дорофеюк Ю.А.

Институт проблем управления РАН

Abstract. The automatic classification (cluster analysis) complex algorithm, that was especially developed for intellectual complex-organized data handling and decision support problems, is described.

It includes: the m-local optimization algorithm with the predetermined classification performance criterion, the informative parameters selection algorithm, the initial fragmentation algorithm, the missing observation filling algorithm.

Введение

Многие крупномасштабные системы управления, в первую очередь организационно-административные, функционируют в условиях большой информационной размытости и неопределённости. Именно поэтому в последнее время для исследования таких систем стали широко использоваться структурно-классификационные методы, базирующиеся на алгоритмах классификационного анализа данных [1].

В настоящей работе рассматриваются задачи анализа функционирования крупномасштабных систем управления, при этом считается, что такая система состоит из достаточно большого числа объектов, каждый из которых характеризуется многочисленным набором разнородных параметров. Основная идея предлагаемого метода решения этой задачи состоит в том, что исследуются не точные значения параметров, описывающих состояние каждого объекта, а лишь структура взаиморасположения этих объектов в пространстве параметров [2]. Такое интегральное описание объектов, входящих в крупномасштабную систему, позволяет существенно повысить эффективность анализа поведения системы, а также устойчивость и робастность процедур принятия управленческих решений. Для формализации такой задачи используется методология классификационного анализа данных [1].

1. Комплексный алгоритм автоматической классификации

Пусть исследуемая система состоит из n объектов, каждый из которых характеризуется набором из k параметров. Вводится в рассмотрение k-мерное пространство параметров X, в котором каждый объект представляется точкой $x_j = (x_j^{(1)}, x_j^{(2)}, \ldots, x_j^{(k)})$, $j = 1, \ldots, n$. Предполагается, что вектор значений параметров x_j достаточно полно характеризует состояние j-го объекта, а это, в свою

[1]Работа выполнена при частичной финансовой поддержке РФФИ, проект 08-07-00349-а.

очередь, означает, что взаиморасположение множества точек $x_1, \ldots, x_n$ в пространстве X отражает реальную структуру (типологию) исследуемого множества объектов. Для выявления такой структуры в работе используется комплексный алгоритм автоматической классификации, специально разработанный для решения таких задач. Комплексный алгоритм включает алгоритмы: m-локальной оптимизации заданного критерия J, выбора информативных параметров, выбора начального разбиения, выбора числа классов, заполнения пропущенных наблюдений. Рассмотрим каждый из этих алгоритмов в отдельности.

1.1. **Алгоритм m – локальной оптимизации.** Вначале опишем работу алгоритма 1-локальной оптимизации. Для простоты изложения рассматривается случай двух классов $r = 2$. Пусть задано начальное разбиение R_0 всех точек классифицируемой выборки $x_1, \ldots, x_n$. Обозначим через $x_j \in A_1$ точки, относящиеся к первому классу, а через $x_j \in A_2$ – ко второму. Алгоритм итерационный, – на каждом шаге рассматривается одна точка из последовательности $x_1, \ldots, x_n, x_1, \ldots, x_n, x_1, .$ («зацикленная» исходная последовательность). Отнесение точки к одному из двух классов обозначается с помощью индекса $\rho(x_j) = \begin{cases} 1, \text{ если } x_j \in A_1 \\ -1, \text{ если } x_j \in A_2 \end{cases}$. Тогда алгоритм 1-локальной оптимизации определяется следующим образом: $\rho\,(x_j) = sign\,[J\,(x_j \in A_1) - J\,(x_j \in A_2)]$.

В итоге точка x_j относится к тому классу, при отнесении к которому, значение критерия J будет больше (если эти значения равны, то для определённости точка относится к классу с меньшим номером). Алгоритм заканчивает работу, если на некотором цикле среди точек $x_1, \ldots, x_n$ не будет сделано ни одной «переброски» точки из класса в класс.

Алгоритм m-локальной оптимизации – это поэтапное применение к выборке алгоритмов s-локальной оптимизации, $s = 1 \div m$. На s-ом этапе алгоритм работает по той же схеме, только на каждом его шаге происходит пробная «переброска» из класса в класс не одной, а s точек. Подсчитывается значение критерия J до и после «переброски», Принадлежность каждой из sточек к классу либо остаётся неименной (J до «переброски» больше, чем после), либо меняется на другой класс – в противном случае. В данном случае цикл – это число шагов, равное числу всевозможных различных наборов, в каждый из которых входит s точек, выбранных из n точек исходной выборки. Доказана сходимость алгоритма за конечное число шагов к локальному максимуму критерия J.

Разработан эвристический алгоритм сокращённого перебора, который на каждом шаге для пробной «переброски» использует s точек в определённом смысле ближайших к границе между классами.

При моделировании и в приложениях в качестве критерия J использовался функционал J_1 средней близости точек в классах, определяемый через потенциальную функцию [3] близости точек x и y:

$$K(x, y) = 1/\{1 + \alpha\, R^p(x, y)\}, \qquad (1)$$

где α и p – настраиваемые параметры алгоритма. Средняя близость точек в классе определяется как:

$$K(A_i, A_i) = \frac{2}{n_i(n_i-1)} \sum_{i=1}^{n_i} \sum_{j>i} K(x_i, x_j), \tag{2}$$

где $K(x_i, x_j)$ определяется формулой (1), n_i– число точек в классе A_i. Тогда критерий J_1 определяется как:

$$J_1 = \sum_{i=1}^{r} \frac{n_i}{n} K(A_i, A_i). \tag{3}$$

1.1.1. *Алгоритм m-локальной оптимизации, одномерный случай.* Необходимо специально отметить частный случай алгоритма m-локальной оптимизации для k=1 (одномерный случай). Дело в том, что одномерный случай имеет уникальное свойство, существенно упрощающее процедуру целенаправленного перебора, используемые при автоматической классификации, а именно: ввиду одномерной упорядоченности классов границей между двумя классами (в детерминированном случае) служит только одна точка, и таких границ может быть не более двух (для крайне правого и крайне левого классов – только одна). Далее описана работа детерминированного (в отличие от общего – размытого) варианта этого алгоритма [4].

Пусть задано начальное разбиение R_0 всех точек классифицируемой выборки $x_1, . \ . \ ., \ x_n$ на r классов. Очевидно, что ввиду упорядоченности классов на оси единственного параметра, на каждом конкретном шаге алгоритма достаточно рассматривать только пару соседних классов, для определённости будем обозначать через A_1 левый из этой пары классов, а через A_2 – правый. Алгоритм содержит mциклов, на s-м цикле (s= 1, ..., m) производится локальная оптимизация классификации, полученной на предыдущем цикле, за счёт процедуры «переброски» s точек из одного класса в другой для каждой пары соседних классов.

На первом цикле производится «переброска» по одной точке. Здесь классификация, полученная на предыдущем цикле, – это начальная классификация R_0. Поясним эту процедуру для первого этапа этого цикла, когда рассматривается пара классов, расположенная в самой левой части диапазона значений x_j. Обозначим через A_1 и A_2 соответственно первый и второй классы начального разбиения R_0. В классе A_1 находится точка $x_j^{1,1,1}$(индексы сверху – номера цикла, этапа и номера класса соответственно), ближайшая к границе рассматриваемой пары классов. Обозначим через $\rho_0\left(x_j^{1,1,1}\right)$ индекс этой точки (для аналогичной точки на s-м цикле это обозначение будет иметь вид $\rho_{s-1}\left(x_j^{s,1,1}\right)$). По построению $\rho_0\left(x_j^{1,1,1}\right)$=1. Затем «перебросим» эту точку в класс A_2 и подсчитаем её индекс на первом цикле:

$$\rho_1\left(x_j^{1,1,1}\right) = sign\left[J\left(x_j^{1,1,1} \in A_1\right) - J\left(x_j^{1,1,1} \in A_2\right)\right], \tag{4}$$

где $J\left(x_j^{1,1,1} \in A_1\right)$ – значение критерия качества классификации J, подсчитанное только для точек классов A_1 и A_2 при условии, что точка $x_j^{1,1,1}$ принадлежит классу A_1, аналогично определяется $J\left(x_j^{1,1,1} \in A_2\right)$. Из (4) следует, что точка $x_j^{1,1,1}$ остаётся в первом классе ($\rho_1\left(x_j^{1,1,1}\right) = \rho_0\left(x_j^{1,1,1}\right) = 1$), если $J\left(x_j^{1,1,1} \in A_1\right) \geq J\left(x_j^{1,1,1} \in A_2\right)$,

и переходит во второй класс ($\rho_1\left(x_j^{1,1,1}\right) = -1$) в противном случае. Если точка $x_j^{1,1,1}$ перешла во второй класс, то аналогичная процедура проделывается с точкой $x_{j-1}^{1,1,1}$, которая является ближайшей к новой границе между классами A_1 и A_2 среди всех точек первого класса (в данном случае – это предыдущая точка классифицируемой последовательности). И так продолжается до тех пор, пока точка $x_{j-l}^{1,1,1}$ не останется в первом классе, т.е. на первом этапе первого цикла из первого класса во второй будут «переброшены» l ближайших к границе точек. Если точка $x_j^{1,1,1}$ осталась в первом классе, то аналогичная процедура проводится с точками второго класса начиная с точки $x_j^{1,1,2}$, которая является ближайшей к границе рассматриваемой пары классов. После того как закончится «переброска» точек из второго класса в первый (если это будет иметь место) либо не произойдёт «переброски» точки $x_j^{1,1,2}$, происходит переход на второй этап первого цикла.

На втором этапе вся последовательность процедур первого этапа повторяется, только через A_1 обозначаются точки, входящие во второй класс после завершения первого этапа первого цикла, а через A_2 – третий класс начального разбиения R_0. И так далее, до тех пор, пока не будут пройдены все (r -1) этапов первого цикла.

На всех этапах s-го цикла описанные процедуры повторяются с точностью до числа «перебрасываемых» точек – «перебрасывается» не по одной, а по s точек, ближайших к границе текущей пары классов. Процедура не применяется для классов A_i, число точек n_i в которых меньше, чем $(s+2)$.

Значение m (глубина перебора) должно выбираться из условия: в классификации, полученной после $(m-1)$-го цикла, должен быть хотя бы один класс, число точек в котором не меньше $(m+2)$. Этим правилом можно воспользоваться для автоматического выбора максимально возможной глубины перебора.

Завершение m-го цикла является окончанием первой итерации. На второй итерации повторяются все процедуры первой, только на первом цикле вместо начального разбиения R_0 используется результирующая классификация первой итерации.

Алгоритм прекращает работу, если в пределах одной итерации не произойдёт ни одной «переброски» точек из класса в класс.

1.2. **Алгоритм выбора информативных параметров.** Этот алгоритм базируется на одном из алгоритмов экстремальной группировки параметров, а именно на алгоритме «квадрат» [3]. В результате его применения получают разбиение исходных k параметров на небольшое (заданное) число групп, а также значения факторов для этих групп. В приложениях используются либо новые интегральные параметры – факторы групп, либо набор параметров, каждый из которых является ближайшим к фактору в соответствующей группе.

В большинстве приложений исходные или выделенные информативные параметры имеют неравнозначную важность для определения структуры объектов. Для выявления таких показателей важности в работе предлагается использовать процедуры экспертного оценивания. Наиболее хорошие результаты дает процедура многовариантной экспертизы [5], когда для получения параметра важности для каждого оцениваемого параметра используется несколько групп экспертов – специалистов в

различных аспектах исследуемой проблемы. В результате экспертизы каждый параметр получает определённый вес (показатель «важности» этого параметра) для формирования структуры объектов.

1.3. **Алгоритм построения начального разбиения.** На первом шаге из всех точек выборки $x_1, \ldots, x_n$ находится пара наиболее удаленных друг от друга точек, x_l и x_p, одна из которых – x_l, относится к первому классу, а другая x_p – ко второму. Если n достаточно велико, то используется усеченный вариант первого шага, а именно: x_l выбирается случайно, а x_p ищется как точка, наиболее от неё удаленная.

На втором шаге ищутся точки x_{l+1} и x_{p+1} – ближайшие, соответственно, к точкам x_l и x_p; точка x_{l+1} относится к первому классу, а x_{p+1} – ко второму.

На (s+1)-ом шаге ищутся точки x_{l+s} и x_{p+s}, ближайшие в среднем к уже найденным точкам, соответственно, первого и второго классов. Точка x_{l+s} определяется следующим образом:

$$x_{l+s} = x_j / \min_{x_j} \frac{1}{s} \sum_{m=0}^{s} K(x_j, x_{l+m}). \tag{5}$$

Точка x_{p+s} определяется аналогично. Если возникает «конфликт», т.е. одна и та же точка является ближайшей к первому и ко второму классам одновременно, то эта точка относится к первому классу. Процедура (5) повторяется до тех пор, пока не будут исчерпаны все точки выборки. Полученное разбиение принимается в качестве начального разбиения R_0.

1.4. **Алгоритм выбора числа классов.** Для выбора числа классов используется специальная экспертно-компьютерная процедура, которая работает следующим образом. Сначала эксперт-пользователь оценивает диапазон $(r_{\min}, r_{\max})$, в пределах которого заведомо находится искомое число классов. Далее, используя любой алгоритм автоматической классификации (в настоящей работе применялся алгоритм m-локальной оптимизации), проводится разбиение анализируемого множества объектов на $r_{\min},\ r_{\min}+1,\ \ldots,\ r_{\max}$ классов. Качество каждой из полученных классификаций оценивалось с помощью критерия $J_3 = J_1 - qJ_2$, где J_1 вычисляется по формуле (3), J_2, а также некоторые вспомогательные величины вычисляются по формулам:

$J_2 = \frac{1}{r-1} \sum_{i=1}^{r} \sum_{j>i} \frac{n_i+n_j}{n} K(A_i, A_j)$; $K(A_i, A_j) = \frac{1}{n_i\, n_j} \sum_{x_l \in A_i} \sum_{x_p \in A_j} K(x_l, x_p)$ – мера близости классов $A_i,\ A_j$; где потенциальная функция $K(x_i, x_j)$ определяется формулой (1); q, α и p из (1) – настраиваемые параметры алгоритма. Фактически, параметр q является масштабирующим параметром, приводящим к соизмеримым средним значениям функционалов J_1 и J_2; на практике величина q имеет значение порядка 2-7 (обычно во столько раз отличается средняя близость внутри классов от средней близости между самими классами).

Формально, в качестве «оптимального» можно выбрать такое число классов r_{opt}, которое соответствует максимальному значению $J_3(r_j)$, т.е. $r_{opt} = r_j$, для которого $\max J_3(r_j), \quad r_j = r_{\min}, \ldots, r_{\max}$. Однако наличие существенной, но неиспользованной при классификации информации, например, ввиду отсутствия данных, может привести к тому, что полученное таким способом r_{opt} не будет «истинно оптимальным».

Для компенсации этого недостатка предлагается использовать следующую экспертную процедуру. Экспертам – специалистам в соответствующей предметной области представляются значения $J_3(r_j)$, $r_j = r_{\min}, \ldots, r_{\max}$, представленные для удобства в виде графика, на котором отмечается значение r_{opt} (оно соответствует максимальной точке на графике $J_3(r_j)$). Используя эту информацию, эксперты могут корректировать выбираемое число классов. В подавляющем числе случаев экспертное число классов либо совпадает с r_{opt}, либо незначительно (± 1) отличается от него.

При классификации многомерных объектов во время такой экспертизы анализируется также классификация каждого объекта. Для этой цели экспертам сообщается информация о мере близости $K(x_i, c_j)$ каждой точки x_i до центров классов $c_j \quad j = 1, \ldots, r_{opt}$ в оптимальной классификации, т.е. матрица близости $\|K(x_i, c_j)\|$, $i = 1, \ldots, n$, $j = 1, \ldots, r_{opt}$. Перенесение точки (объекта) x_i из j-го класса в l-й считается допустимым, если величины $K(x_i, c_j)$ и $K(x_i, c_l)$ отличаются незначительно. Другими словами, содержательно обоснованное перенесение допустимо для точек, расположенных вблизи границы между соответствующими классами.

1.5. **Алгоритм заполнения пропущенных наблюдений.** Во многих приложениях имеются пропуски в данных. В этой ситуации нужно либо использовать специальные процедуры подсчета расстояний между объектами, в параметрах которых имеются пропуски, либо разрабатывать специальные процедуры заполнения таких пропусков. В подавляющем большинстве случаев, пропуски по каждому параметру заполняются средним известных значений соответствующего параметра (для исходной выборки). В настоящей работе была разработана специальная процедура заполнения пропусков в исходных данных с использованием алгоритмов автоматической классификации. Основная идея процедуры состоит в следующем. Если множество изучаемых объектов структурировано (т.е. их можно разделить на классы, достаточно компактно расположенные в пространстве параметров X), то дисперсия (диапазон) изменения каждого параметра в пределах каждой группы, как правило, будет существенно меньше, чем этот показатель для значения этого параметра по всей выборке. Таким образом, если по данным с пропусками удастся определить реальную структуру взаиморасположения точек (т.е. провести классификацию, адекватную этой структуре), то заполнять пропущенное значение l-го параметра для объекта из i-го класса можно средним этого параметра по его известным значениям для всех объектов, попавших в i-ый класс. Исходя из сделанного предположения, отклонение полученного значения от «истинного» должно быть существенно меньше (в среднем), чем обычная схема заполнения по общему среднему.

Заключение

Разработанный комплексный алгоритм использовался для интеллектуализации анализа сложноорганизованных данных, а также при совершенствовании процедур принятия решений для нескольких крупных систем управления, в основном регионального характера. Во всех приложениях, а также при машинном моделировании, была подтверждена высокая эффективность разработанного комплексного алгоритма.

Список литературы

1. *Бауман Е.В., Дорофеюк А.А.* Классификационный анализ данных / Труды Международной конференции по проблемам управления. Том 1. – М.: СИНТЕГ, 1999. – С. 62-67.
2. *Дорофеюк А.А., Дорофеюк Ю.А* Методы структурно-классификационного прогнозирования многомерных динамических объектов / Искусственный интеллект, № 2, 2006. – С.138-141.
3. *Браверман Э.М., Мучник И.Б.* Структурные методы обработки эмпирических данных – М.: Наука, 1983.
4. *Десова А.А., Дорофеюк А.А., Гучук В.В., Дорофеюк Ю.А., Покровская И.В.* Процедуры классификационного анализа в задаче формирования информативных признаков при исследовании ритмической структуры биосигнала / Автоматика и телемеханика. 2008, №6. – С. 143-152.
5. *Дорофеюк А.А., Покровская И.В., Чернявский А.Л.* Экспертные методы анализа и совершенствования систем управления / Автоматика и телемеханика. 2004, №10. – С. 172-188.

Статья поступила в редакцию 27.04.2008

УДК 519.7

МИНИМИЗАЦИЯ ФУНКЦИОНАЛОВ, АССОЦИИРОВАННЫХ С ЗАДАЧАМИ КРИПТОГРАФИЧЕСКОГО АНАЛИЗА АСИММЕТРИЧНЫХ ШИФРОВ

© Дулькейт В.И., Файзуллин Р.Т., Хныкин И.Г.

Омский государственный университет им. Ф.М. Достоевского
пр-т Мира, 55а, г. Омск, 644077, Россия

e-mail: *r.t.faizullin@mail.ru*

Abstract. Global optimization problems associated with cryptographic analysis of asymmetric ciphers. The aim of this article is to establish relation between well-known problems of cryptographic analysis and global optimization problems which can be associated with SAT representation of cryptographic algorithms where bits of key is part of SAT solution string. There was constructions SAT forms for factorization problem, SAT forms for logarithmic problem and logarithmic problem on elliptic curves. For numerical solution was adapted some low relaxation algorithms and for example results of numerical experiments give to us strong more then 50% bits for unknowns in SAT factorization form.

Введение

В работе рассматривается модификация метода логического криптоанализа [1], основанная на процедуре минимизации функционалов, ассоциированных с задачами криптографического анализа асимметричных шифров. Предложены алгоритмы сведения криптографических алгоритмов RSA, дискретного логарифмирования и дискретного логарифмирования на эллиптических кривых к задаче ВЫПОЛНИМОСТЬ и последующему сведению к проблеме поиска глобального минимимума ассоциированных функционалов. Показано, что для задачи факторизации подход позволяет получить строго более чем 60% бит определяющих ключ бит.

1. Концепция

Пусть $L(x)$ – пропозициональная формула в конъюнктивной нормальной форме на множестве булевых переменных $x \in B^N\{0,1\}$. Задача ВЫПОЛНИМОСТЬ (SAT) заключается в том, что бы найти решающий набор $x_0 \in B^N\{0,1\}$, такой что $L(x_0) =$ ИСТИНА или доказать, что решающего набора не существует.

Рассмотрим переход от задачи ВЫПОЛНИМОСТЬ к задаче поиска глобального минимума функционала вида (1).

Пусть дана КНФ:

$L(x) = \bigwedge\limits_{i=1}^{M} c_i$, где c_i – дизъюнкты вида $c_i = \bigvee\limits_{j<N} q_{i,j}(x_j)$. Здесь $q_{i,j}(x_j) = x_j$ или $\bar{x}_j$.

Сделаем переход к эквивалентной ДНФ:

$D = \tilde{L}(x) = \bigvee\limits_{i=1}^{M} \tilde{C}_i$, где $\tilde{C}_i$ – конъюнкты вида $\tilde{C}_i = \bigwedge\limits_{j<N} \tilde{Q}_{i,j}(x_j)$. Здесь $\tilde{Q}_{i,j}(x_j) = \bar{q}_{i,j}(x_j)$

Рассмотрим функционал вида:

$$\min_{x\in E^N} F(x) = \sum_{i=1}^{M} C_i(x),$$

где $C_i(x) = \prod_{j=1}^{N} Q_{i,j}(x_j)$ и где

$$Q_{i,j}(x_j) = \begin{cases} (1-x_j)^2 & \text{, если } x_j \in C_i(x) \\ x_j^2 & \text{, если } \overline{x}_j \in C_i(x) \\ 1 & \text{, иначе} \end{cases} \tag{1}$$

Суммирование ведется по всем M конъюнктам ДНФ, эквивалентной исходной КНФ. Соответствие между булевыми и вещественными переменными следующее: ЛОЖЬ→0, ИСТИНА→1.

Переход от булевой формуле к вещественной основан на использовании соответствия:

$$\begin{cases} y_i \vee y_j \to x_i + x_j \\ y_i \wedge y_j \to x_i^2 x_j^2 \\ \overline{y}_i \to (1-x_i) \end{cases} \text{, где } \{y_i \in B, x_i \in R\}$$

Легко заметить, что $\min_{x\in E^N} F(x) = 0$ соответствует достижению значения ИСТИНА на исходной КНФ.

Без потери общности можно рассмотреть 3-ДНФ, эквивалентную исходной КНФ и ассоциированый функционал:

$J(x) = \sum_{\xi} z_i^2 z_j^2 z_k^2,$

$$\text{где} z_i = \begin{cases} 1-x_i, \text{если} x_i \in c_i(x) \\ x_i, \text{если} \overline{x_i} \in c_i(x) \end{cases}, \ c_i(x) - i \tag{2}$$

триплет Дифференцируя функционал по всем переменным x_i, получаем систему уравнений:

$\sum_{\xi\in\Xi} z_j^2 z_k^2 x_i = \sum_{\xi\in\Lambda} z_j^2 z_k^2, i = 1, 2, ..P,$

$$\text{где } \begin{array}{l} \Xi = \{\xi,\ i \in \xi : x_i \in c_i(x)\} \\ \Lambda = \{\xi,\ i \in \xi : \overline{x_i} \in c_i(x)x\} \end{array} \tag{3}$$

Коэффициент при x_i A_i и правая часть B_i связаны соотношением: $A_i(x_1, .., x_{i-1}, x_{i+1}, ..x_n) \geq B_i(x_1, .., x_{i-1}, x_{i+1}, ..x_n)$

Поясним выбор представления исходной КНФ именно в виде эквивалентной 3-ДНФ. Дифференцируя функционал $F(x)$ (см.(1)) по всем переменным x_i, получаем систему уравнений аналогичную (3), но количество вкладов в A_i и B_i определятся длиной скобок. Любая процедура решения этой системы при произвольной длине скобок будет естественным образом приводить к большим ошибкам округления. Ограничивая число переменных в скобках, мы исключаем эту техническую трудность.

Рассмотрим систему (3), как нелинейное операторное уравнение: $\Phi(x) = 0$.

Как показано в [2] применение метода Ньютона к решению данного уравнения неэффективно, т.к. решение принадлежит ядру производного оператора. Как альтернатива был предложен метод последовательных приближений с «инерцией» [3]:

$$(\sum_{p=0}^{K}\sum_{\xi\in\Xi}\alpha_p x_i(t-p)^2 x_j(t-p)^2)x_k(t+1) = \sum_{\xi\in\Lambda} x_j^2(t)x_k^2(t) \sim A\cdot x_k(t+1) = B$$

$$\sum_{p=1}^{K}\alpha_p = 1 \tag{4}$$

Имеется ввиду то, что итерации происходят для вещественных чисел, а итоговый или промежуточный вектор проектируется на $\mathrm{B}^{\mathrm{N}}\{0,1\}$, и уже на булевом векторе проверяется SAT. Ниже мы опишем различные модификации и гибридизации метода последовательных приближений с «инерцией» в применении к решению задачи K-SAT, и покажем способы повышения эффективности алгоритма.

2. Гибридизация алгоритма

Основная процедура состоит из последовательных итераций, которые совмещают метод последовательных приближений и сдвиг по антиградиенту, т.к. дефект (3) это не что иное, как антиградиент исходного функционала.

Итерация состоит из двух блоков. Первый блок, определяется формулой (5), используется схема Зейделя. Суть схемы Зейделя в том, что при нахождении очередного $x_i(t+1)$ на $(t+1)$-й итерации, это значение подставляется вместо $x_i(t)$. Необходимо отметить, что реализация алгоритма допускает использование схемы Якоби, когда найденные $x_i(t+1)$ не используются в текущей итерации. Тесты показали, что схема Зейделя более устойчива в применении к решаемой задаче. Именно, после каждой итерации по схеме Зейделя значение функционала монотонно уменьшается, чего нельзя сказать о схеме Якоби. Кроме того, во всех случаях схема Зейделя быстрее приводила к решению. Отметим, что данное обстоятельство препятствует напрашивающейся простой схеме распараллеливания процесса решения с разделением данных.

Второй блок – реализация сдвига по градиенту. Рассмотрим (3). Пусть $\bar{x}(t)$ является решением, тогда $\Phi(\bar{x}(t)) = 0$.

Уравнение (3) переписывается в виде $A(\bar{x}(t))\cdot\bar{x}(t) - B(\bar{x}(t)) = 0$. Это необходимое условие, которому должен удовлетворять вектор решения. Если текущее t-е приближение $\bar{x}(t)$ не является решением, то $A_i(\bar{x}(t))\cdot x_i(t) - B_i(\bar{x}(t)) = p_i \neq 0$. Для итеративной формулы: $A_i(\bar{x}(t))\cdot x_i(t+1) - B_i(\bar{x}(t)) = p_i$. Следовательно, что бы удовлетворить необходимому условию, необходимо перейти к вектору:

$\bar{x}_i(t+1) = \bar{x}_i(t) + p_i/A_i$

Очевидно, что после реализации (6) возможна ситуация, когда $\bar{x}_i(t+1) \notin R[0,1]$. В этом случае необходимо штрафным способом ограничивать $\bar{x}_i(t+1)$, иначе метод

начинает экспоненциально расходиться. Особенно это проявляется на K-SAT формулах при K>4. При приближении к решению скорость сходимости может сильно уменьшаться, т.е. алгоритм формирует цикл лежащий на некотором плато (поверхность определяемая функционалом) и траектория, образованная последовательными приближениями более не выходит за пределы этого плато. Чтобы сойти с плато и продолжить сходимость к решению применяется т.н. метод смены траектории.

Метод смены траектории заключается в поиске нового вектора приближения, который бы обладал свойствами не худшими, чем текущий вектор приближения, но позволял бы продолжить поиск решения. Суть метода нахождения такого вектора в следующем.

Рассмотрим 3-КНФ, эквивалентную исходной 3-ДНФ:

$$K(x) = \prod_{\xi} (z_i + z_j + z_k), \text{ где } z_i = \begin{cases} x_i, \text{если} x_i \in c_i(x) \\ 1 - x_i, \text{если} \overline{x_i} \in c_i(x) \end{cases}, \ c_i(x) - i \text{ триплет} \quad (5)$$

$$K(x) = 1 \Leftrightarrow (z_i + z_j + z_k) \neq 0, \forall \xi$$

Для данного приближения, рассмотрим множество переменных:

$$E0 = \{ x_i \mid \exists \text{ триплет } c_i : x_i \text{или} \neg x_i \in c_i \& c_i(\bar{x}) = 0 \}$$

С вероятностью m_p поменяем значения x_i на противоположные. При этом, вероятность того, что другие триплеты станут невыполнимыми не высока. Экспериментально установлено, что полученный вектор x_0 обладает свойствами не худшими, чем $\bar{x}$ (количество невыполнимых триплетов до и после операции примерно одинаково). Используя данный вектор x_0 в качестве нового начального приближения, алгоритм очень быстро (в большинстве случаев за 5-10 итераций) находит следующее приближение, на котором функционал $F(x)$ достигает значения не хуже, чем на векторе $\bar{x}$. При этом, очень часто, удается проскочить плато, но при дальнейшем движении по новой траектории метод может зациклиться на другом плато. Тогда метод смены траектории повторяется.

Дополнительно рассмотрим множество:

$$E1 = \{ x_i \mid \exists \text{ триплет } c_i : x_i \text{или} \neg x_i \in c_i \& c_i(\bar{x}) = 1 \}$$

Введем вероятности m_{p0}, m_{p1}. С вероятностью m_{p0} при смене траектории будем использовать множество $E0$. С вероятностью m_{p1} – множество $E1$. Вероятность m_{p0} влияет на величину изменения вектора и при этом количество невыполнимых триплетов не увеличивается. Вероятность m_{p1} влияет на качественное изменение вектора, количество невыполнимых триплетов может увеличиться. В принципе, чем выше m_{p1}, тем меньшую роль играют рестарты. Метод смены траектории применяется при достижении условия $|F(\bar{x}_2) - F(\bar{x}_1)| < \varepsilon_2$

3. Преобразование исходной КНФ методом резолюции

Данное преобразование позволяет получить КНФ с меньшим количеством дизъюнктов и литералов, эквивалентную исходной. «Резольвента» – дизъюнкция конъюнктов, отличающихся знаком по единственной переменной. Все возможные резольвенты добавляются к КНФ и используются для вычисления других резольвент.

Дублирующие конъюнкты и тавтологии удаляются, и используется сокращенная процедура с глубиной рекурсии 1. Вычислительная сложность процедуры $O(n \cdot \log n)$. Метод резолюции в применении к КНФ, ассоциированных с задачами факторизации и дискретного логарифмирования (см. ниже) позволяет уменьшить исходное число конъюнктов до 50% и иногда разрешить до 20% переменных. Подробно о методе резолюций можно найти в [4].

4. Способы распараллеливания алгоритма

Гибридизированный алгоритм допускает целый набор способов распараллеливания. Исходная формула, делится на определенное количество подформул. Для каждой подформулы находится вектор решения. Найденные вектора некоторым образом объединяются в один, который потом используется для поиска решения всей формулы. Были исследованы два способа реализации параллельного алгоритма.

ПРОЦЕДУРА 1

ДНФ, эквивалентная исходной КНФ делится на n подформул. Для каждой из подформул, с помощью основного алгоритма, ищется выполнимый набор. Полученные вектора используются в качестве инициализирующего набора для следующей подформулы. После n-1 итерации вычисляется усредненный набор. Данный набор используется в качестве инициализирующего для итерационной процедуры, которая применяется ко всей формуле. Вычислительные эксперименты показали что, выполнимый набор для подформул, состоящих из 90% (и более) от исходного скобок числа находится во всех случаях и за минимальное количество итераций (не более 20). Данный способ показал неплохие в среднем результаты при решении различных типов примеров.

ПРОЦЕДУРА 2

ДНФ, эквивалентная исходной КНФ делится на две подформулы. Ищется выполнимый набор для каждой из двух подформул. Вычислительные эксперименты показали, что в полученных наборах решений значения литералов совпадают на 60%. Далее применяется следующая процедура. Осуществляется формирование нового набора приближений для итерационной схемы путём усреднения значений переменных двух наборов, полученных на предыдущем этапе:

$$\mathrm{x}_i = \frac{x_{1i} + x_{2i}}{2}.$$

Далее запускается основная схема решений, но уже для функционала J, который ассоциирован со всей формулой. Тесты показали, что данная процедура увеличивает число выполнимых скобок, но не всегда находит выполнимый набор для всей формулы. Для улучшения результатов была рассмотрена следующая естественная модификация. Каждый вектор решений для соответствующей подформулы, определяет точку в n-мерном пространстве. Между полученными точками проводится отрезок прямой. Двигаясь по этой прямой с некоторым шагом l вычисляются новые вектора x_i по формуле: $x_{li} = \min(x_{1i}, x_{2i}) + \frac{|x_{1i} - x_{2i}|}{k} \cdot l$, где l – это номер шага. Для каждого x_l вычисляется значение функционала J. Затем выбирается тот набор значений $x_{li} = \min(x_{1i}, x_{2i}) + \frac{|x_{1i} - x_{2i}|}{k} \cdot l$, где l, значение, при котором значение данного

функционала минимально. Этот вектор и будет являться новым начальным набором приближений для итерационной процедуры, которая запускается для функционала, ассоциированного со всей формулой.

Как показали тесты, данная процедура наиболее эффективно находит набор решений для многих тестовых формул. Конечно, есть примеры, для которых данный метод не вычисляет точного набора значений. Но описанная процедура позволяет максимально приблизиться к решению. В формуле остаётся до 2% скобок невыполнимыми. Кроме того, 2.5% переменных остаются не использованными, то есть независимо от того какое значение они будут принимать, выполнимые скобки будут по-прежнему принимать значение ИСТИНА. Но при изменении их значения на противоположное количество выполнимых скобок увеличивается.

5. Увеличение разрядности вычислений

Была исследована сходимость алгоритма при увеличении разрядности вычислений. Испытания с типами DOUBLE и FLOAT показали преимущество вычислений с двойной точностью. При переходе на тип DOUBLE количество решенных примеров увеличивается на 10%, скорость сходимости в среднем также увеличивается. Дальнейшее увеличение разрядности к значимому эффекту не приводит.

6. Результаты численных экспериментов

После каждой модификации проводилось тестирование алгоритма для определения эффективности проделанных изменений. При тестировании использовалось несколько типов примеров: тесты с соревнований решателей SAT 2005 года www.lri.fr/∼simon/, тесты библиотеки SATLib www.cs.ubc.ca/∼hoos/, тесты, сформированные для задач факторизации и дискретного логарифмирования, тесты для КНФ больших размерностей, сформированные случайным образом. Подробные результаты представлены в [3].

Оказалось, что сдвиг по градиенту хорошо сокращает погрешности и ускоряет сходимость алгоритма. Вычислительные эксперименты со случайными формулами показали заметное уменьшение времени решения тестов. Число решенных примеров увеличилось примерно на 20%. Применение данного приема позволило достаточно эффективно решать тесты uf20-91 (из 1000 тестов решены 703). Однако некоторые тесты решались только после задания определенного начального приближения, что говорит о необходимости рестартов. На примерах uf250-1065 алгоритм показал результат 6% от стандартных трудных тестов (предыдущая версия алгоритма – 1%). Тесты SAT-2005 (OKGenerator10000-42000 – 10000 переменных, 42000 скобок) использовались в сокращенной форме. Максимально удалось решить подформулы из 36000 скобок (предыдущая версия алгоритма – 35000 скобок). На подформулах из более чем 36000 скобок метод зацикливается на некотором плато, что говорит о необходимости смены траектории.

Метод смены траектории существенно увеличил число решаемых примеров. Результаты представлены в таблице1.

Таблица 1. Результаты тестирования алгоритма + метод смены траектории

Наименование теста	Литералы (N)	Дизъюнкты (M)	Тесты	Успех %	Итерации
BSI, 3-SAT					
RTI	100	429	500	98,6	19988
BMS	100	<429	500	79,8	29831
CBSI, 3-SAT					
CBS_b10	100	403	1000	100	38972
CBS_b10	100	449	1000	100	38880
CBS_b90	100	449	1000	98	29738
UF 3-SAT					
uf20-91	20	92	1000	100	448
uf250-1065	250	1065	100	98	9731
"Flat"Graph Colouring Problems					
flat30-60	90	300	100	100	4317

Таблица 2. Результаты тестирования параллельного алгоритма (процедура 1).

Наименование теста	Литералы	Скобки (M)	% решенных тестов	Число итераций
RTI	100	429	85	14
BMS	100	<429	85	14
sat05-1663	2000	8400	55	200
sat05-1676	4000	16800	50	200
sat05-1656	12000	50400	50	200
uf20-91	20	91	90	14
uf250-1065	250	1065	90	21

Результаты тестирования параллельного алгоритма на различных типах примеров приведены в таблицах 2,3. Можно сделать заключение о том, что процедура 2 более эффективна.

7. Сведение к КНФ задачи факторизации

В последнее время наблюдается повышенный интерес к проблеме кодирования криптографических алгоритмов в терминах задачи выполнимости (SAT). В работе [5] эскизно иллюстрируется подход, позволяющий в принципе свести задачу факторизации к SAT. Далее будет представлен алгоритм построения алгоритмов генерации эквивалентных, но различных КНФ и последующей минимизации ассоциированного функционала, для задачи факторизации, что позволит применить процедуры распаралелливания 1 и 2.

Таблица 3. Результаты тестирования параллельного алгоритма (процедура 2).

Наименование теста	Количество литералов (N)	Количество скобок (M)	% решенных тестов	Число итераций (часть формулы)	Число итераций (вся формула)
RTI	100	429	100	10	14
BMS	100	< 429	100	7	14
sat05-1663	2000	8400	99	20	200
sat05-1676	4000	16800	99	20	200
sat05-1656	12000	50400	99	20	200
uf20-91	20	91	100	10	14
uf250-1065	250	1065	100	20	21

Рассмотрим непосредственно алгоритм сведения к КНФ задачи факторизации. Требуется для заданного числа n получить КНФ, решающий набор которой существует тогда, и только тогда когда n составное число. Кроме того, решающий набор должен содержать все биты двоичного представления нетривиальных делителей n. Без потери общности, рассмотрим классический алгоритм умножения "столбиком". Будем отождествлять биты сомножителей и результата с литералами (свободными логическими переменными). Результат умножения первого сомножителя на i-ый бит второго можно представить в виде вектора $\overline{P_i}$. Именно суммирование всех этих векторов представляет основную сложность. Поэтому предлагается выполнять эту операцию последовательно с сохранением результата в промежуточных векторах $\overline{S_k}$.

Весь процесс вычисления можно разбить на три этапа относительно операции сложения:

1. Сложение векторов составленных из произведений двух литералов. Выполняется один раз. В результате этой операции будет заполнен вспомогательный вектор сумм $\overline{S_1}$ и вычислено два младших бита результата. Условно данный этап можно записать так:$\bar{P}_1 + \bar{P}_2 = (\bar{S}_2, r_2, r_1)$.

2. Суммирование вектора $\overline{S_k}$ с вектором произведений . Выполняется N-3 раз. В результате заполняется массив $\overline{S_{k+1}}$ и вычисляется очередной бит результата

3. Последнее суммирование вектора $\overline{S_k}$ с вектором произведений. Выполняется один раз. В результате вычисляются оставшиеся биты результата.

Теперь перейдём к рассмотрению идеи генерации КНФ. Простейший случай – приравнивание одного литерала другому: $x = y$. Данное равенство будет справедливо тогда и только тогда, когда истинна формула $(\bar{x} \vee y)\,(x \vee \bar{y})$.

Другим часто встречающимся выражением является: $x = A \oplus B \oplus C$ (8)

Поступая аналогичным образом, получаем эквивалентную формулу:

$$(\overline{x} \vee (A \oplus B \oplus C))\left(x \vee \overline{(A \oplus B \oplus C)}\right) = (Ax \oplus Bx \oplus Cx \oplus \overline{x})\,(A\overline{x} \oplus B\overline{x} \oplus C\overline{x})$$

Для представления правой части в виде КНФ воспользуемся леммами 1, 2.

Лемма 1.

$\bigoplus\limits_{i=1}^{N} x_i = \prod\limits_{\{\delta_i\}\in M_N} \left(x_1^{\delta_1} \vee x_2^{\delta_{21}} \vee \cdots \vee x_N^{\delta_N}\right)$, где в левой части сумма по модулю 2, M_N – множество двоичных векторов длины N, содержащих чётное число нулей. Операция "возведения в степень" имеет стандартный для булевой алгебры смысл: $x^{\delta} = \begin{cases} \overline{x}, & \delta = 0 \\ x, & \delta = 1 \end{cases}$

Лемма 2.

$$\bigvee_{i=1}^{N} x_i^{\delta_i} \vee \prod_{j=1}^{L} y_j^{\sigma_j} = \prod_{\{\pi_k\}\in 2^L / \{0,0,\ldots,0\}} \left(\bigvee_{i=1}^{N} x_i^{\delta_i} \vee \bigvee_{j=1}^{L} \left(y_j^{\sigma_j}\right)^{\pi_k}\right)$$

После применения леммы 1 будут получены конъюнкты следующего вида:

$\left(d \vee \overline{a\overline{b}c} \vee x\overline{y}c\right)$, т.е. можно выделить 3 вида дизъюнктов внутри каждого конъюнкта:

1. Одиночные литералы (то к чему следует стремиться: в правильной КНФ все дизъюнкту должны быть одиночными литералами).

2. Дизъюнкты вида $\overline{\prod x_i^{\delta_i}}$, которые по правилу де Моргана можно легко свести к одиночным литералам.

3. Дизъюнкты вида $\prod x_i^{\delta_i}$, наиболее трудный случай, сведение скобок с такими дизъюнктами к КНФ иллюстрируется леммой 2.

Отдельного рассмотрения заслуживает операция вычисления переноса в следующий разряд при суммировании трёх слагаемых. Перенос может быть вычислен через соответствующую сумму: $c = carry\,(x, y, z, sum) = \overline{(sum \oplus xyz \oplus \overline{xyz})}$

Выполнение данного равенства эквивалентно истинности следующей формулы: $\left(\overline{c} \vee \overline{(sum \oplus xyz \oplus \overline{xyz})}\right) \cdot (c \vee (sum \oplus xyz \oplus \overline{xyz}))$.

Приведённое выражение можно преобразовать положив: $x = \overline{c}$, $A = sum$, $B = xyz$, $C = \overline{xyz}$. И далее описанной выше процедурой можно построить соответствующую КНФ. Трудоемкость полученного алгоритма оценивается, как $O(n^2)$ в зависимости от количества бит исходного числа. Для факторизации числа, представляемого двоичным вектором длиной 1024 бит получились КНФ с 500 000 переменными 12 000 000 скобок. Отметим, что результат о превышении числа верных бит после нескольких тысяч итераций алгоритма (4) относится именно к числу переменных, например, из 500 000 переменных мы получаем в среднем более чем 300 000 верных, что в силу очевидных соотношений между битами переноса по строке, отвечающей нулевому биту, может существенно облегчить решение основной задачи.

8. Сведение к КНФ других задач криптоанализа

Были получены аналогичные алгоритмы сведения для задач дискретного логарифмирования и дискретного логарифмирования на эллиптической кривой. Предложен алгоритм генерации множества эквивалентных КНФ для задачи факторизации, учитывающий неделимость на малые простые числа. Последнее обстоятельство позволяет строить параллельные версии приведенных выше алгоритмов и методикой "голосования бит" определять верные биты с достаточно большой верятностью.

Таблица 4. Результаты тестирования полного алгоритма для задачи факторизации.

Число бит	Литералы	Дизьюнкты	Время решения	Время RANOV	Время SATz
20	254	4979	0.1 с	0.1 с	0.1 с
32	801	17867	2 м	>1 ч	1.8м
40	990	22333	7 м	>1 ч	12 м
44	1199	27291	36 м	>1 ч	>1 ч
48	1428	32741	3,5 ч	>10ч	>10 ч
56	1946	45141	36 м	>10ч	>10 ч
60	2235	52079	10,2ч	>20 ч	>20 ч
68	2873	67455	79 ч	>100 ч	>100 ч
72	3222	75881	168ч	>200 ч	>200 ч

9. Результаты работы алгоритма для задачи факторизации

Результаты приведены в табл. 4. Для группы задач длины до 72 бит факторизуемого числа были получены точные решения. При этом эффективность предложенного метода превосходит известные нам алгоритмы.

Другой интересный результат в том, что уже после первых нескольких сотен итераций метод находит более 60% верных бит решения. Трудность заключается в определении, какие именно биты были определены верно. При этом при росте числа бит исходного факторизуемого числа, происходит рост процентного отношения числа верных бит для соотношения, определяющего факторизуемое число (см. Рис. 1).

Для тестирования использовалось 50 различных тестов для каждой размерности. В качестве сомножителей выбирались числа, удовлетворяющие всем тестам, гарантирующим криптостойкость RSA. На рисунке представлены усредненные данные. Обратим внимание на то, что вне зависимости от размерности задачи для каждого теста проводилось всего 1000 итераций. Исходя из рисунка можно сделать предположение о том, что рост числа верных бит в зависимости от длины факторизуемого числа имеет логарифмический характер.

Аналогичные результаты получены и для задачи дискретного логарифмирования, но основным препятствием здесь является высокая размерность получающихся задач, так для 1024 бит число переменных в функционале оценивается величиной 1000 000 000, а число дизъюнктов на порядок больше. Результаты точного решения КНФ, эквивалентных задаче дискретного логарифмирования приведены в таблице 5.

Заключение

Разработанный метод не уступает известным методам решения SAT на многих группах тестовых примеров и превосходит их на тестах задачи факторизации больших размерностей. Показан рост относительного числа верно найденных бит для задачи факторизации с ростом размерности задачи, так для рабочего числа

Рис. 1. Рост процентного отношения числа верных бит к числу бит факторизуемого числа.

Таблица 5. Результаты тестирования полного алгоритма для задачи дискретного логарифмирования.

Размерность	Литералы	Дизьюнкты	Время (сек)	RANOV (сек)	SATz (сек)
18	28224	448018	63.57	97.23	81.16
20	38840	623239	108.20	>1800	>1800
22	51832	839032	182.73	>1800	>1800
24	67440	1099630	277.46	>1800	>1800
26	85904	1409250	417.71	>1800	>1800
Знак '>' означает, что за указанное время решение найдено не было					

бит 1024 среднее значение верных бит равно 65%, что больше стартового значения в 62% для 50 бит. Кроме того, показано наличие "слабых" размерностей, для которых метод является эффективным.

Список литературы

1. *Cook S.A.* The Complexity of Theorem Proving Procedures. Proceedings Third Annual ACM Symposium on Theory of Computing, May 1971.
2. *Файзуллин Р.Т., Хныкин И.Г., Дулькейт В.И., Салаев Е.В.* Алгоритм минимизации функционала, ассоциированного с задачей 3-SAT и его практические применения, г.Челябинск, 2007.
3. *Файзуллин Р.Т.* О решении нелинейных алгебраических систем гидравлики // Сибирский журнал индустриальной математики. – 1999. -№2. – С. 176-184.
4. *Хныкин И.Г.* Модификация КНФ, эквивалентных задачам криптоанализа асимметричных шифров
методом резолюции // ИТМУ № 8, 2007.
5. *Беспалов Д.В. Семёнов А.А.* О логических выражениях для задачи 2-ФАКТОРИЗАЦИЯ // Вычислительные технологии. – 2002. – Т.7 – Ч.2.

Статья поступила в редакцию 01.05.2008

УДК 514.1

ОЦЕНКА АСИММЕТРИИ ЛИЦА ПО ТРЕХМЕРНОМУ ПОРТРЕТУ

© Дышкант Н.Ф., Местецкий Л.М.

Московский государственный университет им. М.В. Ломоносова
факультет Вычислительной математики и кибернетики
Воробьевы горы, МГУ им. М.В. Ломоносова, 2-й учебный корпус, г. Москва, 119992, Россия

e-mail: *nfd3001@gmail.com, L.mest@ru.net*

Abstract. The estimate for the assymetry of a face of a man by its 3D model is proposed in the paper. It contains the detailed algorithm constructed for this estimate calculation. The used face modes are obtained by a methof of 3D scanning and are represented in the form of points cloud. The problem of detection of a face symmentry plane is solved. The algorithm was tested on 200 samples of face models. This testing shows the stability of the proposed estimate.

Введение

Вследствие быстрого развития технологий трехмерного сканирования объектов появилась необходимость в умении анализировать полученные поверхности и сравнивать их между собой. В данной работе *предлагается оценка асимметрии человеческого лица по его трехмерной модели.*

Известно, что лицо человека обладает лишь *приближенной* зеркальной (двусторонней) симметрией относительно плоскости, делящей его на правую и левую половины. Предлагается мера, позволяющая оценивать степень такого приближения, т.е. степень *асимметрии.*

Задача оценки асимметрии лица возникает при профилактике аномалий зрения у детей (см. [1]), в медицинских и психологических исследованиях, профотборе и в других приложениях. *Изучение работ, посвященных рассматриваемой проблеме, показало, что тремя основными направлениями, в которых пытаются использовать асимметрию лица в последнее время, являются:*

1. *идентификация личности — распознавание по лицу (*[2]*);*
2. *распознавание эмоций(*[3], [4]*);*
3. *определения пола (*[5]*).*

Асимметрия объекта входит в его структурное описание, и оценка асимметрии не может быть получена с помощью каких-либо локальных критериев. Для получения такой оценки необходимо анализировать объект целиком и, кроме того, уметь определять *плоскость симметрии.* В работе показано, какое влияние оказывает определение плоскости симметрии на оценку, и *предложен алгоритм нахождения плоскости симметрии.* При этом решается экстремальная задача: делается предположение о том, что наиболее точно найденной плоскости соответствует наименьшая оценка.

В работе предлагается метрика для сравнения трехмерных моделей и алгоритм ее вычисления. При оценивании асимметрии лица производится сравнение исходной модели и модели, отраженной относительно плоскости симметрии.

В указанных выше работах ([2]–[5]) используются модели, приведенные к *регулярным* сеткам: поэтому в качестве оценки асимметрии в них предлагается вычислять сумму или среднее значение по всем разницам высот между исходной и отраженной моделями (левой и правой половинами модели). Рассматриваемые нами модели заданы на таких сетках, что точки, в которых известны значения поверхности лица, расположены несимметрично относительно плоскости симметрии модели. Такие сетки могут быть как регулярными, так и *нерегулярными. Поэтому в данной задаче требуется введение более сложной оценки для вычисления количественного значения асимметрии.*

Поверхность лица, полученная с помощью 3D сканирования, описывается в виде облака точек и является однолистной функцией двух переменных, заданной на некотором дискретном множестве — нерегулярной сетке. Сравнение двух моделей лиц сводится к сравнению соответствующих им функций F_1 и F_2, заданных на разных нерегулярных сетках G_1 и G_2. Основные этапы алгоритма сравнения таких функций приводились авторами в [6]; в основе предложенного алгоритма лежит идея *восполнения* значений каждой из них в узлах другой сетки через построение триангуляций Делоне и локализацию их друг в друге.

Целями данной работы являются: введение оценки асимметрии человеческого лица по его трехмерной модели и разработка алгоритма ее вычисления, обладающего высокой вычислительной эффективностью.

Авторы выражают благодарность сотрудникам «Artec Group company» (http://www.artec-group.com) за предоставленные модели лиц.

Данная работа выполнена при поддержке Российского фонда фундаментальных исследований (гранты 08-01-00670, 08-07-00305-а).

1. База данных 3D моделей

Трехмерные модели, которые мы использовали в данном исследовании, были получены с помощью 3D камеры, разработанной в «Artec Group company». Была собрана база, содержащая 191 модель лиц 8 разных людей. При этом съемка анфас происходила при нейтральном выражении лиц снимающихся.

Каждая модель представлена набором точек заданных своими координатами (x, y, z) в пространстве. Координаты точек предоставлены с точностью до 10^{-6} в масштабе $1 : 1$, т.е. соответствуют реальным размерам лица человека. Количество точек в моделях варьируется от 1 000 до 3 000, а в среднем по базе составляет 1 500–2 000.

В системе координат модель расположена (нормализована) таким образом, что кончик носа соответствует началу координат, ось Oz направлена вдоль оси визирования, а ось Oy проходит вдоль лица (см. рис. 1). При таком расположении плоскость Oyz можно считать плоскостью симметрии модели. Следует отметить, что описанное расположение является приближенным и может быть уточнено с помощью преобразований системы координат – малых сдвигов и поворотов на малые углы.

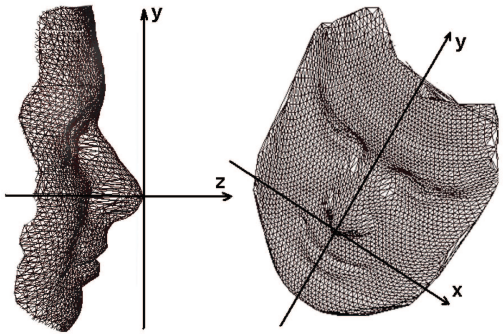


Рис. 1. Трехмерная модель лица

2. Постановка задачи

Поверхность лица, полученная методом трехмерного сканирования, представляет собой облако точек и может быть рассмотрена как однолистная функция двух переменных, заданная на некоторой нерегулярной сетке.

Определение 1. Нерегулярной двумерной (плоской) сеткой G называется конечное множесто точек из пространства $\mathbb{R}^2$:

$$G = \left\{(x^i, y^i) \in \mathbb{R}^2 \;\middle|\; i = 1, \ldots, N\right\}.$$

Приведем описание содержания математической задачи, решение которой будет использовано для оценки асимметрии модели. Пусть исходная маска задана функцией F_1, а маска, полученная отражением исходной по отношению к плоскости Oyz — функцией F_2. Сравнение двух моделей сводится к сравнению соответствующих им функций.

Пусть даны две нерегулярные двумерные сетки G_1 и G_2:

$$G_1 = \left\{(x^i, y^i)\right\}_{i=1}^N, \qquad G_2 = \left\{(-x^i, y^i)\right\}_{i=1}^N.$$

Пусть обе сетки заключены внутри некоторого общего прямоугольника R. Пусть в точках сетки G_1 задана функция F_1, в точках сетки G_2 — функция F_2. Таким образом, функции заданы наборами своих значений $\left\{f_1^i\right\}_{i=1}^N$, $\left\{f_2^i\right\}_{i=1}^N$ в соответствущих точках:

$$f_1^i = F_1(x^i, y^i), \qquad f_2^i = F_2(-x^i, y^i).$$

Требуется разработать метод, позволяющий вычислять меру различия между двумя моделями в виде объема разности между ними, т.е. в следующем виде:

$$\iint\limits_R \left|\hat{F}_1(x, y) - \hat{F}_2(x, y)\right| \mu(x, y)\, dxdy,$$

где $\hat{F}_1$ и $\hat{F}_2$ — непрерывные на R функции, аппроксимирующие функции F_1 и F_2 соответственно, а $\mu(x, y)$ — функция, определяющая вес различных фрагментов поверхности с точки зрения важности их сходства.

3. Алгоритмы

Основные шаги предлагаемого подхода к решению задачи получения оценки асимметрии лица изображены на схеме (рис. 2).

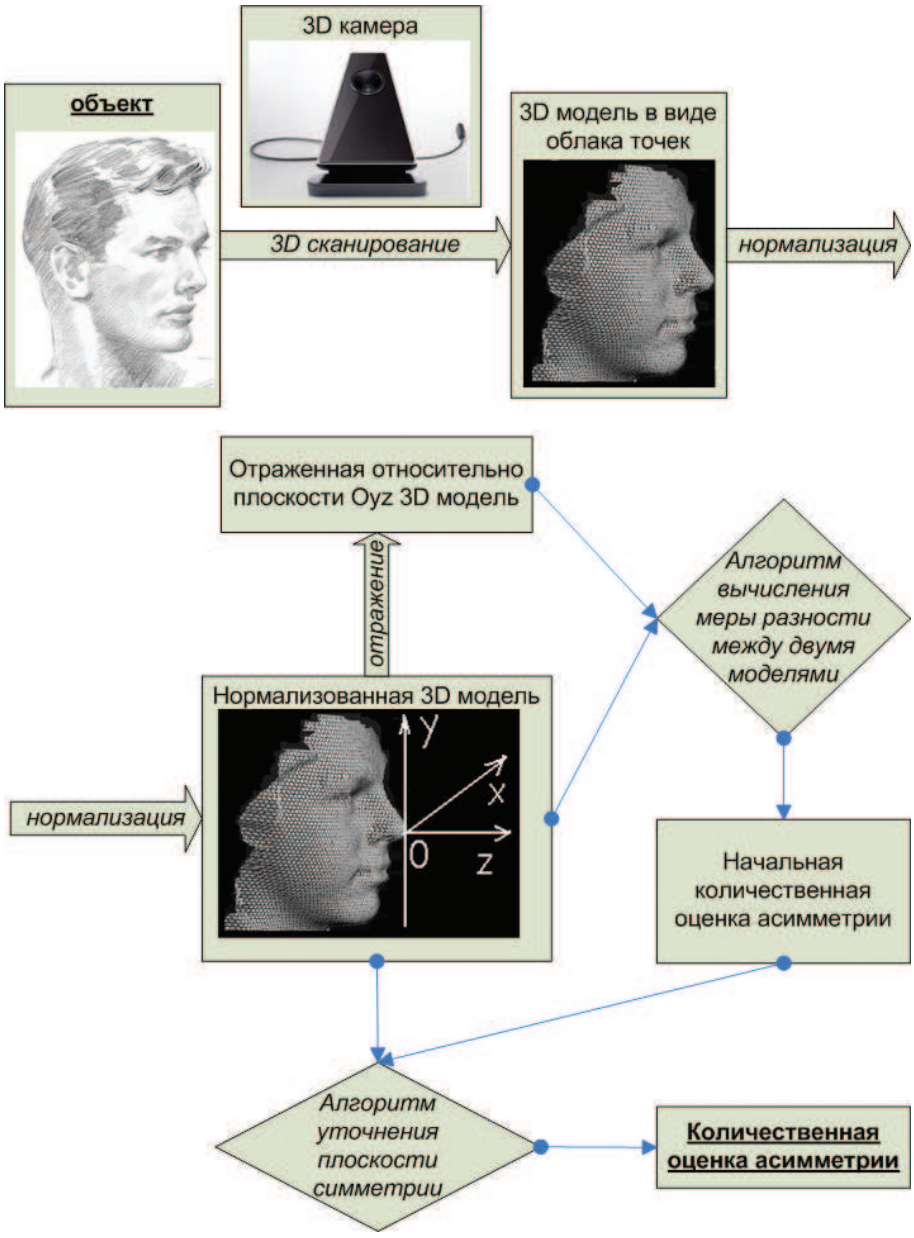


Рис. 2. Общая схема предлагаемого подхода

После нормализации модели в системе координат строится ее симметричное отражение относительно плоскости Oyz (см. рис. 3 — участки лица, на которых отраженная маска выше исходной, выделены более темным цветом).

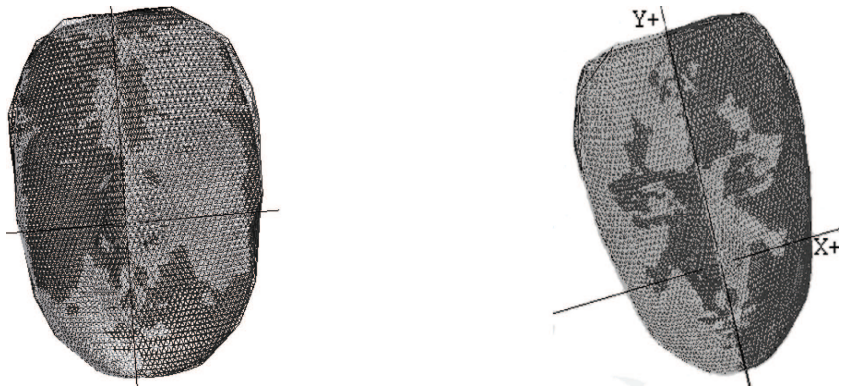


Рис. 3. Примеры разных исходных масок и их отражений относительно плоскостей симметрии

3.1. **Получение начальной оценки асимметрии.** Вначале на каждой из сеток G_1 и G_2 строится триангуляция Делоне. При этом используется алгоритм, описанный в [7] и [8], в основе которого лежит парадигма рекурсивной декомпозиции («разделяй и властвуй»): разделение исходного множества на два примерно равных подмножества, рекурсивное построение триангуляций этих подмножеств и слияние двух разделенных триангуляций. Вычислительная сложность данного алгоритма $O(N \log N)$.

После построения триангуляций следует *этап локализации* их друг в друге.

Локализовать точку Q в триангуляции Делоне T означает указать треугольник триангуляции, содержащий внутри себя эту точку. В тех случаях, когда точка Q совпадает с вершиной триангуляции или принадлежит одному из ее ребер, можно указать любой из инцидентных данной вершине или данному ребру треугольников. В рассматриваемой нами задаче точка Q, являющаяся вершиной одной триангуляции, локализуется в другой триангуляции T. Так как по условию обе триангуляции заключены в прямоугольник R, то точка Q не может заходить за границы R, и локализация всегда корректна.

Идея алгоритма, с помощью которого решается задача локализации точки, состоит в выборе некоторой начальной точки M, локализация которой в триангуляции известна, и постепенном переходе от M к Q вдоль прямой MQ. При этом на каждом шаге осуществляется переход на смежный (соседний по стороне) треугольник. Таким образом, в процессе локализации точки строится путь из треугольников триангуляции, каждый из которых (кроме начального) является смежным с предыдущим — *путь локализации* (см. рис. 4). Трудоемкость локализации одной точки определяется количеством треугольников, расположенных вдоль отрезка $[MQ]$, и составляет $O(\sqrt{N})$ в среднем и $O(N)$ в худшем случае.

Локализовать двумерную сетку в триангуляции означает локализовать все точки данной сетки в этой триангуляции.

Предлагается алгоритм локализации сетки, использующий *остовное дерево*, вершинами которого являются точки данной сетки. В этом случае пути локализации

проходят вдоль ребер остовного дерева. Так как остовное дерево не содержит в себе циклов и проходит через все узлы сетки G, алгоритм будет работать корректно: он не зациклится и произведет локализацию абсолютно всех точек сетки. На одном из последующих этапов предлагаемый подход использует общую триангуляцию двух сеток, построенную методом слияния, предложенным и описанным в [9]. Этот метод использует *минимальные остовные деревья* (МОД) обеих сеток. Поэтому удобнее использовать при локализации сетки именно *минимальное* остовное дерево. Тогда пути локализации будут оптимальны (см. рис. 5).

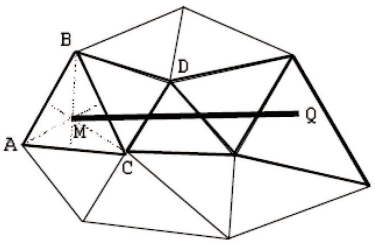


Рис. 4

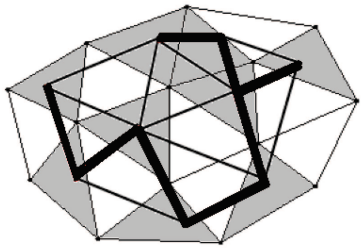


Рис. 5

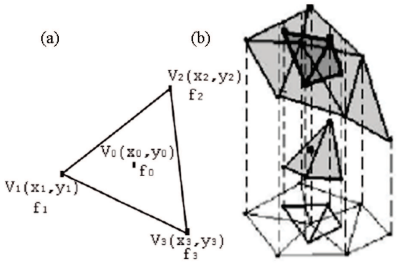


Рис. 6

Известно, что из триангуляции Делоне минимальный остов можно построить за *линейное* время. Линейное время достигается благодаря операции чистки, предложенной Черитоном и Тарьяном в [10], и использованию структуры данных «фибоначчиева куча», введенной Фредманом и Тарьяном (см. [11], [12]).

Далее происходит *интерполяция функций* на основе локализации.

Пусть точка $V_0(x_0, y_0)$ локализована в треугольнике $\Delta\ (\ V_1(x_1, y_1), V_2(x_2, y_2), V_3(x_3, y_3)\ )$. В точках этого треугольника задана функция F (см. рис. 6а):

$$F(x_1, y_1) = f_1; \quad F(x_2, y_2) = f_2; \quad F(x_3, y_3) = f_3.$$

Требуется проинтерполировать значение функции F в точке V_0.

Будем использовать барицентрические коорината:

$$f_0 = \lambda_1 f_1 + \lambda_2 f_2 + \lambda_3 f_3, \qquad \lambda_1, \lambda_2, \lambda_3 \geqslant 0.$$

где

$$\begin{cases} V_0 = \lambda_1 V_1 + \lambda_2 V_2 + \lambda_3 V_3 \\ 1 = \lambda_1 + \lambda_2 + \lambda_3 \end{cases} \Leftrightarrow \begin{cases} x_0 = \lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3 \\ y_0 = \lambda_1 y_1 + \lambda_2 y_2 + \lambda_3 y_3 \\ 1 = \lambda_1 + \lambda_2 + \lambda_3 \end{cases}.$$

Вычислительные эксперименты показали, что сложность этапа интерполяции сетки составляет в среднем $O(N)$.

Указанным способом значения функции F_1 интерполируются во всех точках двумерной сетки G_2, а значения функции F_2 — в точках сетки G_1. После этого в каждой

точке обеих сеток известны значения двух функций: одно из них было задано, а второе получено в результате интерполяции.

Затем на точках обеих сеток строится общая триангуляция Делоне. За счет того, что нам известна локализация точек одной триангуляции в треугольниках другой, наиболее эффективным здесь оказывается использование алгоритма слияния неразделенных триангуляций, описанного в [9].

Оценка асимметрии вычисляется в виде суммы объемов разности функций F_1 и F_2 на каждом треугольнике общей триангуляции. Для этого рассматриваются все случаи взаимного пространственного расположения треугольников (см. рис. 7).

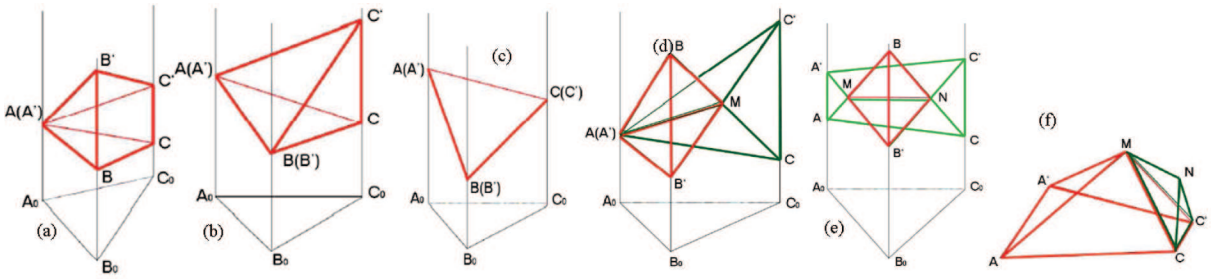


Рис. 7. Случаи пространственного расположения треугольников

Для подсчета объема разности между функциями на треугольнике необходимо считать либо объем пирамиды — треугольной или четырехугольной (см. рис. 7a-c), либо объем двух треугольных пирамид (см. рис. 7d), либо объем треугольной пирамиды и клина (см. рис. 7e), при этом объем клина ищется в виде суммы объемов четырехугольной и треугольной пирамид (см. рис. 7f).

Суммируя значения объемов разности по всем треугольникам общей триангуляции, мы получаем значение *начальной количественной оценки асимметрии.*

3.2. **Нахождение плоскости асимметрии.** Далее следует этап уточнения плоскости симметрии с помощью элементарных преобразований системы координат. Так как нормализация моделей в системе координат не является абсолютно точной, мы пробуем преобразовать координаты с помощью небольших сдвигов по осям координат и поворотов вокруг осей на малые углы. Уточнение происходит с целью нахождения такого положения плоскости симметрии Oyz, при котором значение количественной оценки асимметрии минимально (то есть, при котором две маски совпадают наилучшим образом). При этом минимизация будет происходить не по всем шести параметрам преобразований, а лишь по трем: сдвиги по осям Oy и Oz не будут давать вклад в оценку асимметрии, поворот вокруг оси Ox не рассматривается в силу того, что лица снимаются анфас. Минимизация будет происходить по трем оставшимся параметрам: сдвигу по оси Ox и поворотам вокруг осей Oy и Oz.

Пусть T — общая триангуляция Делоне двух нерегулярных сеток; A, B, C — вершины треугольника; $F_1(A,B,C)$ — функция, возвращающая массив значений функции лица в вершинах треугольника ΔABC; $F_2(A,B,C,x,\varphi,\psi)$ — функция, возвращающая массив значений функции лица, отраженного относительно плоскости Oyz, сдвинутого на x по оси Ox, повернутого на угол φ вокруг оси Oy и на угол ψ вокруг оси Oz, в вершинах ΔABC. Тогда общая триангуляция будет зависеть от параметров сдвига $T = T(x,\varphi,\psi)$.

Поставим следующую оптимизационную задачу:

$$\Phi(T) = \Phi(T(x,\varphi,\psi)) = \Phi(x,\varphi,\psi) = \sum_{\Delta ABC} |F_1(A,B,C) - F_2(A,B,C,x,\varphi,\psi)| \to \min_{x,\varphi,\psi},$$

где суммирование ведется по всем треугольникам ΔABC общей триангуляции, а под модулем разности подразумевается объем разности между двумя пространственными треугольниками — заданными исходной функцией F_1 и отраженной F_2 в точках A, B, C.

Для минимизации функции Φ использовались два метода: метод градиентного спуска и покоординатного спуска. Метод покоординатного спуска с использованием метода золотого сечения обладает большей вычислительной эффективностью в данной задаче.

В качестве первоначального приближения берется точка, соответствующая совпадению плоскости симметрии с плоскостью Oyz в исходной системе координат, т.е. точка $M_0(x_0,\varphi_0,\psi_0) = (0,0,0)$.

При минимизации учитывается, что функция является овражной: изменение Δ переменных φ и ψ приводит к бо́льшему изменению значения функции, чем такое же изменение Δ переменной x.

В таблице приведены значения начальной количественной оценки асимметрии и оценки асимметрии после уточнения плоскости симметрии для 4 моделей лица одного и того же человека:

Таблица Объем разности между моделью и ее отражением (все значения приведены в кубических миллиметрах).

Номер модели	*I*	*II*	*III*	*IV*
Начальная оценка асимметрии	47 466,361	49 192,110	43 473,767	46 280,040
Оценка асимметрии	24 072,518	25 205,272	24 421,316	22 263,813

Величина объема разности между исходной и отраженной моделями, измеренная в столовых ложках жидкости, составляет примерно полторы столовые ложки[1].

При этом оптимальный сдвиг по оси Ox для этих моделей составлял от 2,4 до 2,6 мм, а оптимальные углы поворота вокруг осей координат Oy и Oz — примерно 0,015 рад.

Используемый метод позволяет анализировать асимметрию модели локально. Таким образом можно определить, какие участки лица наиболее асимметричны с точки зрения предложенной оценки. Для этого для каждого треугольника общей триангуляции T вычисляется значение отношения интеграла модуля разности (метрика l_1)

[1]Объем столовой ложки составляет 15 000 мм^3.

функций F_1 и F_2 на этом треугольнике к его площади. На рис. 8 изображены по три модели лиц для двух разных людей. Каждая грань общей триангуляции моделей окрашена в соответствии с нормированным значением указанного отношения — наиболее светлые участки являются наиболее асимметричными.

Рис. 8. Зонная (локальная) асимметрия лица
Слева: три разные модели лица одного человека, справа: три разные модели лица другого человека

Заключение

В работе *определена оценка для асимметрии лица* человека по его трехмерной модели. *Описаны алгоритмы вычисления данной оценки и определения плоскости симметрии модели.*

Предложенный метод позволяет вычислять оценки с использованием меры важности асимметрии на каждом из участков лица. Таким образом, предложенная оценка может быть адаптирована для каждого конкретного приложения.

Перспективы дальнейших исследований заключаются в понимании того, насколько предложенная оценка удовлетворяет гипотезе компактности с помощью проведения дальнейших вычилительных экспериментов на существующей базе 3D моделей и на базах большего объема, и в адаптации предложенной оценки с помощью введения меры на основе определения тех участков лица, которые наиболее важны с точки зрения асимметрии в задачах идентификации личности или медицинских приложениях.

Алгоритм вычисления оценки асимметрии допускает распараллеливание и обладает высокой вычислительной эффективностью.

Список литературы

1. *Книжников Ю.Ф., Гельман Р.Н., Крыченков В.Ф.* Применение цифровой фотограмметрии при диагностике патологии стереоскопического зрения, 2005.
2. *Mitra S., Lazar N., Liu Y.* 2007. Understanding the Role of Facial Asymmetry in Human Face Identification // Statistics and Computing, Vol. 17, pp. 57 - 70, January, 2007.
3. *Teng K., Liu Y.* Expression Classification using Wavelet Packet Method on Asymmetry Faces // tech. report CMU-RI-TR-06-03, Robotics Institute, Carnegie Mellon University, January, 2006.
4. *Mitra S., Liu Y.* Local Facial Asymmetry for Expression Classification // Proceedings of the 2004 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'04), Vol. 2, pp. 889 - 894, June, 2004.

5. *Liu Y., Palmer J.* A Quantified Study of Facial Asymmetry in 3D Faces // Proceedings of the 2003 IEEE International Workshop on Analysis and Modeling of Faces and Gestures, in conjunction with the 2003 International Conference of Computer Vision (ICCV '03), October, 2003.
6. *Дышкант Н.Ф., Местецкий Л.М.* Сравнение 3D портретов при распознавании лиц // доклады конференции «ММРО-13», 2007.
7. *Препарата Ф., Шеймос М.* Вычислительная геометрия: введение // М.: Мир, Москва, 1989.
8. *Скворцов А.В., Костюк Ю.Л.* Эффективные алгоритмы построения триангуляции Делоне // Геоинформатика. Теория и практика. Вып. 1. Томск: Изд-во Томского ун-та, 1998. 22-47.
9. *Местецкий Л.М., Царик Е.В.* Триангуляция Делоне: рекурсия без пространственного разделения точек // In Graphicon, International Conference on computer graphics, Moscow, 2004.
10. *Cheriton D., Tarjan R.E.* Finding minimum spanning trees // SIAM J.Comput., 5(4), p.724-742 (Dec.), 1976.
11. *Tarjan R.E.* Fibonacci heaps and their uses in improved network optimization algorithms // In journal of the ACM., 1987.
12. *Fredman M.L., Tarjan R.E.* Data Structures and Network Algorithms // Society for Industrial and Applied Mathematics, 1989.

Статья поступила в редакцию 30.04.2008

УДК 519.714

ИССЛЕДОВАНИЕ АЛГЕБРАИЧЕСКИХ ЗАМЫКАНИЙ АЛГОРИТМОВ РАСПОЗНАВАНИЯ: ОПЕРАТОРЫ РАЗМЕТКИ

© Дьяконов А.Г.

Московский государственный университет им. М. В. Ломоносова,
г. Москва, Россия

e-mail: djakonov@mail.ru

Abstract. The results received last years within the limits of the algebraic approach to the recognition problems are described. Algebraic closures of algorithms of classical estimation algorithm model are investigated.

Введение

В конце 70х годов прошлого века академиком РАН Ю.И. Журавлёвым был предложен алгебраический подход к решению задач распознавания образов [1]. Основная идея алгебраического подхода: из некорректных алгоритмов (которые делают ошибки на контрольной выборке) можно получать корректные, компенсируя недостатки одних достоинствами других. В качестве основной модели исследований была выбрана модель алгоритмов вычисления оценок (АВО), в описании которой нашли отражение основные концепции решения задач распознавания [2].

Каждый алгоритм модели АВО представляется в виде суперпозиции распознающего оператора (РО) и решающего правила (РП). Над РО, которые по исходной информации получают матрицы оценок принадлежности объектов к классам, были введены операции: сложения, умножения, умножения на константу (как операции над соответствующими матрицами). Алгебра над операторами индуцирует алгебру над алгоритмами (используется фиксированное РП). В [1] показано, что корректный алгоритм может быть выписан в явном виде, как полином над некорректными алгоритмами.

К началу XXI века в алгебраическом подходе оставались открытыми несколько интересных проблем: не известна точная оценка степени корректного полинома; не предложены упрощения понятия корректность, которые позволили бы получать более простые и эффективные алгебраические конструкции; не исследованы некоторые другие естественные операции над алгоритмами (например, деление); не описаны «хорошие» и «плохие» задачи, с точки зрения алгебраических замыканий моделей алгоритмов. Ниже представлены решения этих проблем и техника [3]–[5], позволившая их решить.

1. Основные понятия

Напомним постановку задачи распознавания образов, описание модели АВО и основные понятия алгебраического подхода, немного обобщив классические определения [2].

Множество допустимых объектов M разбито на классы: $M = K_1 \cup \ldots \cup K_l$. Для каждой пары $(S^t, S_i) \in M \times M$ объектов можно вычислить значения функций $\rho_\Omega(S^t, S_i) \in \mathrm{E}_A$, $\Omega \in \Omega_A$, Ω_A – конечное множество параметров, E_A – частично упорядоченное множество (порядок на котором будем обозначать символом $\leq$).

Функция ρ_Ω имеет смысл расстояния, однако мы не будем накладывать на неё никаких ограничений. Каждому допустимому объекту S соответствует бинарный вектор классификации $\tilde{\alpha}(S) = (\alpha_1(S), \ldots, \alpha_l(S))$, где $\alpha_j(S)$ – значение предиката «$S \in K_j$», $j \in \{1, 2, \ldots, l\}$.

Задача распознавания образов состоит в том, чтобы построить алгоритм A, который по набору (эталонных, обучающих) объектов $\tilde{S}^m = \{S^t\}_{t=1}^m$ с известными векторами $\tilde{\alpha}(S^1), \ldots, \tilde{\alpha}(S^m)$ для набора (контрольных, тестовых) объектов $\tilde{S}_q = \{S_t\}_{t=1}^q$ строит их векторы классификации – классифицирует (распознаёт).

РО B модели АВО вычисляет матрицу оценок $\Gamma[B] = \|\Gamma_{ij}[B]\|_{q\times l}$ такую, что

$$\Gamma_{ij}[B] = \sum_{a,b=0,0}^{1,1} x_{ab}(j) \sum_{\Omega\in\Omega_A} \sum_{S^t\in\tilde{K}_j^a} w^t w(\Omega) \mathrm{B}_\Omega^{\tilde{e},b}(S^t, S_i),$$

где $w^t \in \mathbf{Q}^+$ при $t \in \{1, 2, \ldots, m\}$ (*вес t-го объекта*), $w(\Omega) \in \mathbf{Q}^+$ при $\Omega \in \Omega_A$ (*вес учёта Ω-й близости*), $\mathrm{B}_\Omega^{\tilde{e},b}(S^t, S_i)$ – *функция близости*, $\tilde{e}$ – параметры функции (из множества E_A),

$$\mathrm{B}_\Omega^{\tilde{e},0}(S^t, S_i) = 1 - \mathrm{B}_\Omega^{\tilde{e},1}(S^t, S_i), \quad \mathrm{B}_\Omega^{\tilde{e},1}(S^t, S_i) = \begin{cases} 1, & \rho_\Omega(S^t, S_i) \le \tilde{e}, \\ 0, & \rho_\Omega(S^t, S_i) \not\le \tilde{e}, \end{cases}$$

$$\tilde{K}_j^a = \begin{cases} \tilde{S}^m \cap K_j, & a = 1, \\ \tilde{S}^m \backslash K_j, & a = 0. \end{cases}$$

Далее в работе рассматриваем модель АВО с параметрами $x_{00}, x_{11} \in \{0, 1\}$, $x_{01}, x_{10} \in \{0, -1\}$, $x_{ab} = x_{ab}(j)$ для всех $j \in \{1, 2, \ldots, l\}$, $a \in \{0, 1\}$, $b \in \{0, 1\}$ (модель без нормировок). Предполагаем также, что существуют параметры $\tilde{e}_1 = \tilde{e}_1(\tilde{S}^m, \tilde{S}_q)$ такие, что $\rho_\Omega(S^t, S_i) \le \tilde{e}_1$ для всех $S^t \in \tilde{S}^m$, $S_i \in \tilde{S}_q$, $\Omega \in \Omega_A$.

РП C алгоритма модели АВО по матрице оценок классифицирует объекты. При этом "разумно" относить объект к классу с "достаточно большой" оценкой принадлежности. Простейшее РП (пороговое):

$$C\left(\|\Gamma_{ij}[B]\|_{q\times l}\right) = \|\alpha_{ij}\|_{q\times l}, \quad \alpha_{ij} = \begin{cases} 1, & \Gamma_{ij} > c, \\ 0, & \Gamma_{ij} \le c, \end{cases} \quad c \in \mathbf{Q}^+.$$

Следующие операции над РО (сложение, умножение на константу, умножение) индуцируют соответствующие операции над алгоритмами при фиксированном РП [1]:

$$\|\Gamma_{ij}[B_1 + B_2]\|_{q\times l} = \|\Gamma_{ij}[B_1]\|_{q\times l} + \|\Gamma_{ij}[B_2]\|_{q\times l}, \quad \|\Gamma_{ij}[cB]\|_{q\times l} = c\,\|\Gamma_{ij}[B]\|_{q\times l},$$

$$\|\Gamma_{ij}[B_1 B_2]\|_{q\times l} = \|\Gamma_{ij}[B_1]\|_{q\times l} \circ \|\Gamma_{ij}[B_2]\|_{q\times l}$$

($\circ$ – адамарово умножение матриц). Множество $U^k(B^*)$ полиномов (с нулевым свободным членом) степени не выше k от операторов из множества B^* операторов АВО называется *алгебраическим замыканием k-й степени*. Замыкание $U^1(B^*)$ называется также *линейным*.

2. Операторы разметки

Определение 1. *Оператором разметки* называется оператор, который получает оценки по формуле

$$\Gamma_{ij}[\cdot] = I[\rho_\Omega(S^t, S_i) = \tilde{e}] \cdot I[\alpha_j(S^t) = a],$$

$$I[\rho_\Omega(S^t, S_i) = \tilde{e}] = \begin{cases} 1, & \rho_\Omega(S^t, S_i) = \tilde{e}, \\ 0, & \rho_\Omega(S^t, S_i) \neq \tilde{e}, \end{cases}$$

$$I[\alpha_j(S^t) = a] = \begin{cases} 1, & \alpha_j(S^t) = a, \\ 0, & \alpha_j(S^t) \neq a, \end{cases} = \begin{cases} 1, & S^t \in \tilde{K}_j^a, \\ 0, & S^t \notin \tilde{K}_j^a. \end{cases}$$

Идея использования таких операторов была взята из работ [6, 7]. Их матрицы оценок имеют простой вид, кроме того, справедлива

Лемма 1. *Для множества D^* операторов разметки справедливо равенство $U^k(B^*) = U^k(D^*)$.*

Рассмотрение алгебраического замыкания k-й степени с помощью операторов разметки, с технической точки зрения, не отличается от рассмотрения линейного замыкания. Далее опишем проблемы, окончательные решения которых были получены в последние годы с помощью этой техники.

3. Полученные результаты

1. Исследование квазикорректности (корректности относительно семейства РП). Корректность является центральным понятием алгебраического подхода к решению задач распознавания (см. [1, 2, 6, 7]). Под корректностью модели понимают возможность реализовать произвольную матрицу оценок с помощью операторов этой модели. Интуитивно такое требование представляется завышенным. Кроме того, оно является прямым следствием использования одного фиксированного РП, а не семейства правил.

Определение 2. Модель называется *корректной относительно семейства РП*, если для любой матрицы классификации $||\alpha_{ij}|| \in \{0,1\}^{q \times l}$ существует РО модели и РП семейства, суперпозиция которых получает эту матрицу.

Понятия корректности вводятся при фиксированной постановке задачи (относительно этой постановки). На семейство РП естественно накладывать требования монотонности. Справедливы следующие результаты [4], которые показывают, что переход к «естественным» семействам РП не изменяет критериев корректности.

Теорема 1. *Модель РО $U^k(B^*)$ корректна относительно семейства построчно и постолбцово монотонных РП тогда и только тогда, когда она корректна.*

Теорема 2. *Модель РО $U^k(B^*)$ корректна относительно семейства построчно монотонных РП тогда и только тогда, когда она корректна или $l = 1$.*

2. Неулучшаемая оценка степени корректного полинома. Улучшаемые (для модели АВО) оценки были получены В.Л. Матросовым ($k \leq q + l - 2$, 1981 г.), Т.В. Плохониной ($k \leq m$, 1985 г.), К.В. Рудаковым ($k \leq [\log_2(ql)]$, 1989 г.).

Теорема 3. *Если корректна модель* $U^{\infty}(B^*)$, *то корректна модель* $U^k(B^*)$, *где* $k = [\log_2 q] + [\log_2 l]$. *Существует регулярная задача распознавания, в которой модель* $U^k(B^*)$ *некорректна при* $k < [\log_2 q] + [\log_2 l]$.

Здесь $[x]$ – целая часть снизу числа x. Понятие регулярной задачи введено в [1], оно отражает необременительные требования на условие задачи (начальная информация непротиворечива), при которых алгебраическое замыкание $U^{\infty}(B^*)$ корректно.

3. Исследование пополненной алгебры над АВО. Как изменятся корректные алгоритмы (их вид, сложность), если алгебру над алгоритмами пополнять новыми операциями? Приведём результаты исследования алгебры, пополненной делением и её «слабыми формами»: нормировками.

Теорема 4. *Замыкание операторов вида* $\sum\limits_{j\in X} c'_j \frac{1}{\sum\limits_{i\in X_j} c_i B_i} + \sum\limits_{t\in X'} c''_t B_t$ *корректно в регулярной задаче распознавания.*

Запись $\frac{1}{B}$ обозначает оператор, матрица оценок которого получается из матрицы оператора B заменой каждого элемента на обратный (по умножению). Таким образом, деление является очень мощной операцией: ею достаточно пополнить линейное замыкание АВО.

Определение 3. *Нормировкой по сумме* называется операция N_{Σ} над вещественными матрицами:

$$N_{\Sigma}(||\gamma_{ij}||_{q\times l}) = \left|\left| \frac{\gamma_{ij}}{\sum\limits_{s=1}^{l} \gamma_{is}} \right|\right|,$$

которая определена, когда в каждой строке матрицы $||\gamma_{ij}||_{q\times l}$ сумма элементов отлична от нуля.

Аналогично определяется нормировка по максимуму (делим на максимальный элемент в строке). Интересно, что структура алгебраических замыканий в алгебре, пополненной операцией нормировки, зависит от некоторых множеств A, Θ векторов, которые однозначно выписываются по операторам разметки. Приведём лишь некоторые результаты для нормировок по сумме и максимуму [5].

Теорема 5. *Линейное замыкание, пополненное нормировкой по сумме, имеет размерность* $q(\mathrm{rank(A)} - 1) + \mathrm{rank}(\Theta)$ *и является корректным тогда и только тогда, когда* $\mathrm{rank}(\Theta) = q$, $\mathrm{rank(A)} = l$.

Теорема 6. *Линейное замыкание, пополненное нормировкой по максимуму, имеет размерность* $\mathrm{rank}(\Theta)$ *при* $l = 1$ *и* $q \cdot \mathrm{rank(A)}$ *при* $l > 1$, *оно является корректным тогда и только тогда, когда* $\mathrm{rank(A)} = l > 1$ *или* $(\mathrm{rank}(\Theta) = q)\&(l = 1)$.

4. Точное описание «хороших» и «плохих» задач для алгебраических замыканий. Необходимо получить полное описание задач, которые решаются алгоритмами ограниченной сложности. Частично проблема была решена В.Л. Матросовым в [7], однако полученные им результаты не допускали простой геометрической интерпретации.

Оказывается, что каждой задаче соответствует метрика на множестве QL пар (контрольный объект, класс). Разрешимость в линейном замыкании соответствует невырожденности определителя матрицы попарных расстояний между этими парами. Переход к алгебраическому замыканию k-й степени соответствует специальному преобразованию этой матрицы. Кроме того, найдено семейство операторов, которое достаточно рассматривать при решении практически любой задачи в рамках алгебраического подхода (например, поиск корректного алгоритма минимальной степени, или определение возможности реализации заданной классификации операторами из $U^k(B^*)$).

Теорема 7. *Если матрица оценок может быть получена оператором из* $U^k(B^*)$, *то она может быть получена оператором вида* $\sum\limits_{(a,b)\in QL} c_{(a,b)}B^k_{(a,b)}$, *где* $B_{(a,b)}$ *– сумма некоторых операторов разметки.*

Список литературы

1. *Журавлёв Ю.И.* Корректные алгоритмы над множествами некорректных (эвристических) алгоритмов. I-II // Кибернетика, 1977, №4, 6, С. 5–17, 21–27.
2. *Журавлёв Ю.И.* Об алгебраическом подходе к решению задач распознавания или классификации // Пробл. кибернетики. М.: Наука, 1978, Вып.33, С. 5–68.
3. *Дьяконов А.Г.* Алгебра над алгоритмами вычисления оценок: Учебное пособие. М.: Изд. отдел ВМК МГУ, 2006.
4. *Дьяконов А.Г.* Корректность относительно семейства решающих правил // Искусственный интеллект, 2006, №2, С. 61-64.
5. *Дьяконов А.Г.* Алгебра над алгоритмами вычисления оценок: нормировка и деление // Ж. вычисл. матем. и матем. физ., 2007, Т.47, №6, С. 1099-1109.
6. *Матросов В.Л.* Корректные алгебры ограниченной ёмкости над множеством алгоритмов вычисления оценок // Ж. вычисл. матем. и матем. физ., 1981, Т.21, №5, С. 1276-1291.
7. *Матросов В.Л.* О критериях полноты модели алгоритмов вычисления оценок и её алгебраических замыканий // Докл. АН СССР. 1981, Т.258, №4, С. 791-796.

Статья поступила в редакцию 19.04.2008

УДК 004.932

СОПОСТАВЛЕНИЕ СТЕРЕОИЗОБРАЖЕНИЙ КАК ЗАДАЧА О НАЗНАЧЕНИИ

© Жук Д.В., Тузиков А.В.

Объединенный Институт Проблем Информатики НАН Беларуси
ул. Сурганова, 6, г. Минск, 220012, Беларусь

e-mail: *dzhuk@tut.by, tuzikov@newman.bas-net.by*

Abstract. An algorithm for calculating dense disparity map for stereo images is proposed. Stereo correspondence problem is formulated in terms of an assignment problem in this algorithm. The algorithm processes each image scan-line individually similar to a dynamic programming approach to the dense stereo correspondence problem, however unlike dynamic programming, it does not rely on ordering constraint. Ordering constraint elimination is useful for the scenes with narrow objects in front of the camera. The proposed algorithm supports occlusions detection and guaranties uniquness of found matches.

Введение

Задача сопоставления стереоизображений состоит в нахождении точек на стереопаре, которые являются проекциями одной и той же точки пространства. Такие точки называются *сопряженными*. *Диспаратностью* называется сдвиг проекции точки на одном изображении относительно проекции той же точки на другом изображении. Множество всех диспаратностей для двух изображений называется *отображением диспаратности*. После того, как определены диспаратности, можно реконструировать трехмерную модель изображенной сцены, если известны параметры камер. Таким образом, задача нахождения сопряженных точек на изображениях играет значительную роль в общей задаче восстановления трехмерной модели по нескольким изображениям. Корректность отображения диспаратности является ключевой для успешного построения модели.

Из-за различных неоднозначностей, возникающих в процессе сопоставления (например, невидимость некоторых регионов на одном из изображений, зеркальные отражения, отсутствие текстуры), часто используются ограничения (например, эпиполярная геометрия) и предположения (например, одинаковая степень освещенности и гладкость поверхностей). Использование эпиполярной геометрии сводит область поиска соответствий для данной точки к прямой, поэтому эпиполярные ограничения часто используются в различных методах поиска сопряженных точек. Обычно считается, что входные изображения прошли предварительную обработку и являются выпрямленными, чтобы наиболее эффективно использовать эпиполярные ограничения [1].

В зависимости от используемых ограничений, методы поиска сопряженных точек можно разбить на две группы: локальные и глобальные [2]. Ограничения на небольшое количество пикселей в окрестности данного называются *локальными ограничениями*. Ограничения на строки изображения или изображения целиком называются *глобальными ограничениями*. Примерами локальных методов являются блочный

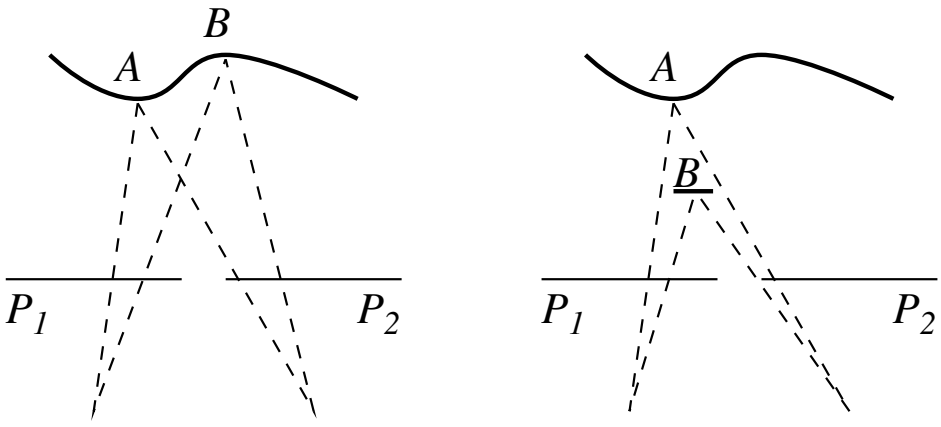
метод и методы, использующие характерные особенности изображений (углы, границы). Методы на основе динамического программирования и разрезов в графе — примеры глобальных методов. Локальные методы часто очень эффективны, но чувствительны к локально-неоднозначным областям изображений, таким как регионы с невыраженной текстурой. Глобальные методы, менее чувствительные к таким областям, как правило, более вычислительно емкие.

Блочный метод находит диспаратность в заданной точке изображения путем сравнения небольшой окрестности (шаблона) этой точки с такими же областями другого изображения, лежащими внутри ограниченной области поиска. Для сравнения двух областей часто используются следующие меры: нормированная кросскорреляция, сумма абсолютных разностей, сумма квадратов разностей, ранговое преобразование. Ограничение данного метода состоит в том, что уникальность соответствий гарантируется только для одного изображения. Однако одна точка другого изображения может соответствовать нескольким точкам исходного изображения. Кроме того, данный метод не определяет точки видимые только на одном из изображений, таким образом находя соответствия для всех точек исходного изображения.

Методы, использующие динамическое программирование, находят путь минимальной стоимости в матрице всех попарных стоимостей между двумя соответствующими стоками изображений. В [3] частично видимые области обрабатываются явным образом за счет присвоения соответствующего состояния узлам пути на основании заданной стоимости перекрытия. Данный метод также гарантирует однозначность соответствий. Для возможности применения метода динамического программирования вводится ограничение на порядок следования объектов: если объект A находится слева от объекта B на одном изображении, то A также будет слева от B и на втором изображении. Таким образом, предполагается, что порядок следования соответствующих пикселей сохраняется на обоих изображениях (рис. 1(а)), что в общем случае, например при наличии узких объектов на переднем плане, не так (рис. 1(б)).

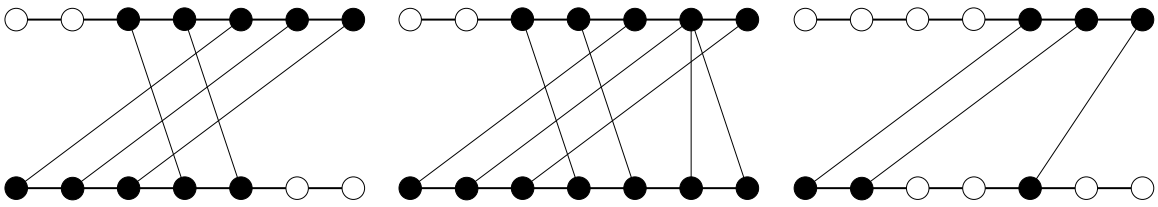
Допустим, что пиксели двух строк изображений I (исходное изображение) и I' соответствуют, как показано на рис. 2(а). Возможный результат работы блочного метода показан на рис. 2(б): все пиксели исходного изображения имеют соответствие на изображении I' из-за отсутствия обработки пикселей, видимых только на одном из изображений, а некоторые пиксели изображения I' имеют более одного соответствия, поскольку алгоритм не учитывает ограничения на однозначность соответствий. Огранчение на порядок следования также не позволит алгоритму динамического программирования корректно определить соответствия в рассматриваемом случае: некоторые пиксели могут быть некорректно классифицированы как не имеющие соответствия или сформируют неверное соответствие (рис. 2(в)).

Ограничение порядка используется также рядом других методов. Однако ограничение порядка сокращает круг сцен, для которых возможно применение этих методов. Избегание данного ограничения может быть полезным в некоторых приложениях. Предлагаемый подход на основе задачи о назначении работает со строками изображений и поддерживает однозначность соответствий и определение частично



(а) (б)

Рис. 1. Ограничение на порядок следования объектов: (a) порядок следования не меняется; (b) узкий объект на переднем плане изменяет порядок



(а) (б) (в)

Рис. 2. Сопряженные точки: (a) истинное соответствие; (b) блочный метод; (c) динамическое программирование

видимых пикселей, подобно методу динамического программирования. Он не использует ограничение на порядок следования объектов и может быть полезен при реконструкции сцен, нарушающих это ограничение, например, сцен с узкими объектами на переднем плане.

1. Поиск сопряженных точек как задача о назначении

В общем виде задача о назначении формулируется следующим образом. Пусть имеется некоторое количество работников и работ. Любой работник может быть назначен для выполнения любой из имеющихся работ. Стоимость выполнения работы

Таблица 1. Задача поиска сопряженных точек как задача о назначении

	a_1	a_2	a_3	a_4	a_5	a_6	...	a_{m-1}	a_m	a'_1	a'_2	...	a'_{n-1}	a'_n
t_1	$c_{1\,1}$	$c_{1\,2}$	$c_{1\,3}$	$c_{1\,4}$			...			c_{occ}		...		
t_2		$c_{2\,2}$	$c_{2\,3}$	$c_{2\,4}$	$c_{2\,5}$		...				c_{occ}	...		
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
t_n								$c_{n\,m-1}$	$c_{n\,m}$			...		c_{occ}

варьируется в зависимости от рабочего и от работы. Требуется выполнить все работы, назначив в точности одного рабочего на каждую работу, таким образом, чтобы общая стоимость выполнения работ была минимальна. Эта задача может быть решена за время $O(n^3)$ при помощи венгерского алгоритма, который был разработан Г. Куном [4].

Задача поиска сопряженных точек может быть сформулирована как задача о назначении. Пусть каждый пиксель p_i строки изображения I представляет собой работу t_i, а каждый пиксель p'_j изображения I' — рабочего a_j. Стоимость $c_{i\,j}$ выполнения работы t_i рабочим a_j определим как стоимость соответствия пикселей $c(p_i, p'_j)$:

$$c_{i\,j} = \begin{cases} c(p_i, p'_j), & \text{если } 0 \le d_{i\,j} \le d_{\max}; \\ +\infty, & \text{иначе,} \end{cases} \tag{1}$$

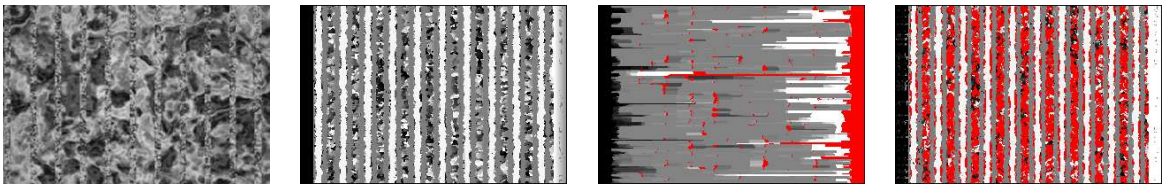
где $d_{i\,j}$ – значение диспаратности для p_i и p'_j, $d_{\max}$ – максимальное значение диспаратности. Для нахождения областей, видимых только на одном из изображений, можно добавить дополнительного рабочего a'_i для каждой работы t_i со стоимостью c_{occ}.

Сформулированная задача может быть представлена в более удобном виде, как показано в таблице 1. В таблице через n и m обозначено количество пикселей в строке изображений I и I' соответственно. Пустые ячейки обозначают, что два пикселя не могут быть сопряженными ($c_{i\,j} = +\infty$).

Требование, чтобы каждая работа была назначена одному из рабочих, и чтобы каждый рабочий выполнял не более одной из работ, обеспечивает выполнение однозначности назначений. Дополнительные рабочие a'_i позволяют учитывать видимость пикселей только на одном из изображений.

Решение сформулированной задачи о назначении может быть преобразовано в отображение диспаратности следующим образом:

- если работа t_i выполняется рабочим a_j, то пиксель p_i соответствует пикселю p'_j;
- если работа t_i выполняется рабочим a'_i, то пиксель p_i виден только на изображении I;
- если ни одна из работ не назначена рабочему a_j, то пиксель p'_j виден только на изображении I'.



(а) (б) (в) (г)

Рис. 3. Результат вычисления изображения диспаратности по синтезированным изображениям: (a) одно из исходных изображений; (b) блочный метод; (c) динамическое программирование; (d) метод на базе задачи о назначении

2. Сравнение с другими алгоритмами

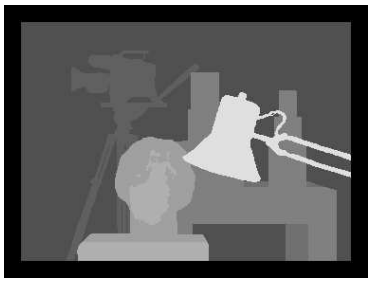
Основным преимуществом предлагаемого метода является возможность работы с изображениями, на которых присутсвуют узкие объекты вблизи объектива камеры. Для оценки качества определения отображения диспаратности использовались синтезированные черно-белые изображения сцены, содержащей узкие полосы на переднем плане и плоскость на заднем плане. Все объкеты сцены обладают четко выраженной текстурой, что позволяет избежать локальных неоднозначностей, преодоление которых проблематично для таких методов, как блочный. В качестве меры сходства использовалась абсолютная разность интенсивностей по блоку 3×3 пикселей. Результаты представлены на рис. 3.

Как видно, блочный метод (рис. 3(б)) корректно определил полосы на переднем плане, а также области фона. Однако области, видимые только на одном из изображений, содержат некорректные значения. Из-за нарушения ограничения на следование объектов метод динамического программирования (рис. 3(в)) показал неудовлетворительные результаты. Предлагаемый метод (рис. 3(г)) показал результаты, схожие с блочным методом, однако области, видимые только на одном снимке, были корректно определены.

Результаты вычисления диспаратности с использованием предлагаемого метода и блочного метода, а также метода динамического программирования для пары фотоснимков, показаны на рис. 4.

Значительное различие между методом динамического программирования и предлагаемым методом состоит в использовании стоимости перекрытия c_{occ}. Стоимость перекрытия в методе динамического программирования влияет на гладкость получающегося пути (высокая стоимость приводит к постоянным значениям диспаратности на всем изображении). Метод на базе задачи о назначении использует стоимость перекрытия для избегания соответствий с высокой стоимостью. Использование малых значений стоимости приведет к отображению диспаратности, практически полностью состоящему из частично видимых областей.

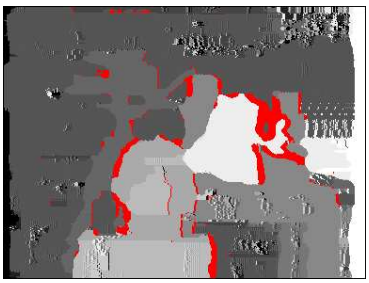
(а)



(б)

(в)

(г)



(д)

Рис. 4. Результат вычисления отображения диспаратности по фотоснимку: (a) одно из исходных изображений; (b) истинное значение диспаратности; (c) блочный метод; (d) динамическое программирование; (e) метод на базе задачи о назначении

Таблица 2. Точность определения диспаратности

Алгоритм	P_1	P_2
Блочный	48.85%	73.07%
Динамическое программирование	56.76%	74.58%
Задача о назначении	71.28%	84.36%

Для сравнения точности определения диспаратности использовались изображения с известными значениями диспаратности. Точность определялась исходя из следующих метрик:

- P_1 – доля пикселей с точно определенной диспаратностью;
- P_2 – аналогично P_1, но допускается ошибка в один пиксель.

Границы изображений не включались в вычисление метрик, поскольку многие алгоритмы не дают адекватные результаты в этой области. Во всех тестах использовались одинаковые настройки (максимальное значение диспаратности и меры сходства). Результаты представлены в таблице 2.

Заключение

Предлагаемый метод поиска сопряженных точек на стереоизображениях на базе задачи о назначении поддерживает однозначность соответствий и определение областей, видимых только на одном из изображений. Данный метод позволяет улучшить точность определения диспаратности в локально-неоднозначных областях по сравнению с блочным методом за счет нахождения однозначных соответствий, а также имеет некоторые преимущества по сравнению с методом динамического программирования.

Тестирование на изображениях с узкими объектами на переднем плане показывает, что предлагаемый метод корректно определяет частично видимые области и диспаратность в областях с нарушением ограничения порядка. Однако, из-за отсутствия ограничения на гладкость диспаратности, ошибки определения диспаратности могут быть значительными в некоторых регионах по сравнению с методом динамического программирования.

Список литературы

1. *R. Hartley, A. Zisserman* Multiple View Geometry in Computer Vision, Cambridge University Press, 2001.
2. *M. Brown, D. Burschka, G. Hager* Advances in Computational Stereo // IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25 (8), 2003, P. 993-1008.
3. *A. Bobick, S. Intille* Large Occlusion Stereo // International Journal of Computer Vision, Vol. 33 (3), 1999, P. 181–200.
4. *H. Kuhn* The Hungarian method for the assignment problem // Naval Research Logistics Quarterly, Vol. 2, 1955, P. 83–97.

Статья поступила в редакцию 20.04.2008

УДК 517.9

КЛЮЧЕВЫЕ АНТИЦЕПИ РЕШЕТКИ ОПИСАНИЙ ИНТЕРВАЛОВ ПРИЗНАКОВОГО ПРОСТРАНСТВА

© Ильченко А.В.

Таврический национальный университет им. В.И. Вернадского
факультет математики и информатики
пр-т Вернадского, 4, г. Симферополь, 95007, Украина

Abstract. The notion of the key antichain for the lattice of characteristic space interval descriptions, key antichains family property, the algorithm of the key antichains construction are conside the paper.

Введение

Алгоритмы и методы кластерного анализа во многом составляют основу приложений анализа данных. Рост объема данных, подлежащих обработке, предъявляет ряд требований к используемым алгоритмам кластеризации: возможность находить кластеры в пространстве большой размерности, наглядность, легкость интерпретации полученных результатов, отсутствие необходимости приведения исходных данных к какому-либо каноническому виду и так далее.

В некоторых случаях выполнение многих из этих требований могут обеспечить алгоритмы и методы анализа формальных понятий [1], [2].

Целью статьи является рассмотрение понятия ключевой антицепи решетки описаний интервалов признакового пространства, изучение свойств семейства ключевых антицепей, построение и обоснование алгоритма поиска ключевых антицепей, основанного на использовании некоторых конструкций теории решеток, классического и обощенного анализа формальных понятий.

1. Признаковое пространство и его области

S – признаковое пространство, каждое измерение (признак, атрибут) которого является конечным линейно упорядоченным множеством.

G – множество объектов. Каждому объекту g из множества G соответствует некоторая точка s признакового пространства S – проекция объекта g в признаковое пространство S. Значениями координат такой точки являются значения, принимаемые атрибутами на объекте g.

Точки признакового пространства, которые являются проекциями объектов из множества G, удобно называть «занятыми» точками ($\bar{0}$-точками) пространства S.

$H_{\bar{0}}$ – проекция множества G в признаковое пространство S («занятая» область, $\bar{0}$-область пространства S).

$H_0 = S \setminus H_{\bar{0}}$ – дополнение проекции H_0 до пространства S («свободная» область, $\bar{0}$-область пространства S).

$S = H_0 \cup H_{\bar{0}}$ – представление признакового пространства в виде объединения «свободной» и «занятой» областей.

2. Сокращенная интервальная структура признакового пространства [4]

Интервал I, содержащийся в области H, называется максимальным интервалом области H, если не существует его собственного надинтервала, содержащегося в области H.

Множество всех максимальных интервалов области H называется сокращенной интервальной структурой этой области.

$I_m(H)$ – обозначение множества всех максимальных интервалов области H.

Каждую область H признакового пространства S можно представить в виде объединения интервалов ее сокращенной интервальной структуры.

Множество всех максимальных интервалов, определяемых «свободной» и «занятой» областями пространства S, называется сокращенной интервальной структурой признакового пространства, порождаемой проекцией множества G в пространство S.

Пространство S можно представить в виде объединения интервалов его сокращенной интервальной структуры.

3. Решетка интервалов признакового пространства

$P(S)$ – множество всех подмножеств признакового пространства S. Отношение включения «$\subseteq$» определяет отношение предшествования (порядка) на этом множестве. Известно [3], что множество $P(S)$, упорядоченное таким образом, является полной решеткой. Нижняя и верхняя грани такой решетки совпадают с теоретико-множественными операциями пересечения и объединения соответственно.

$I(S)$ – семейство всех интервалов признакового пространства. Это семейство обладает свойством замыкания [3]. Поэтому $I(S)$, рассматриваемое вместе с унаследованным отношением предшествования «$\subseteq$», является полной решеткой. В этой решетке нижняя грань совпадает с теоретико-множественным пересечением интервалов: $I \sqcap J = I \cap J$, а верхняя грань $I \sqcup J$ – это наименьший интервал, содержащий интервалы I и J (интервальная оболочка, интервальное замыкание объединения интервалов I и J). Наименьший элемент этой решетки – пустое множество, наибольший элемент – признаковое пространство S, рассматриваемое как интервал.

$I^{\cap}(S) \subseteq I(S)$ – подмножество интервалов, неразложимых в пересечение ($\cap$-неразложимых).

$I^{\cap}(I) = \{J \in I^{\cap}(S) | I \subseteq J\}$ – подмножество неразложимых в пересечение интервалов, содержащих интервал I.

$I^{\cap}_{min}(I) = min\ I^{\cap}(I)$ – антицепь из минимальных неразложимых в пересечение интервалов, содержащих интервал I.

В силу того, что решетка $(I(S), \subseteq)$ является конечной, семейство $I^{\cap}(S)$ является плотным по пересечению подмножеством этой решетки [3]. Поэтому каждый интервал $I \in I(S)$ представим в виде пересечения (разложим в пересечение) интервалов некоторого подмножества семейства $I^{\cap}(S)$, например

$$I = \cap I^{\cap}(I) \tag{1}$$

В (1) для вычисления пересечения достаточно использовать только минимальные элементы подмножества $I^{\cap}(I)$:

$$I = \cap I^{\cap}_{min}(I) \tag{2}$$

То есть, справедливо

Утверждение 1. *Для каждого интервала семейства $I(S)$ существует антицепь из $\cap$-неразложимых интервалов, являющаяся $\cap$-разложением этого интервала:*

$$\forall I \in I(S) \exists A \subseteq I^{\cap}(S) : A - \text{антицепь и } I = \cap A. \tag{3}$$

Например, антицепь $I^{\cap}_{min}(I)$.

Примечание 1. Для $\cap$-неразложимого интервала в качестве такой антицепи рассматривается одноэлементное подмножество, элементом которого является сам этот интервал.

Утверждение 1 справедливо для элементов любой конечной решетки. Для тех интервалов решетки $(I(S), \subseteq)$, которые не являются пустым множеством, справедливо более сильное утверждение.

Утверждение 2. *Для каждого интервала семейства $I(S)$, не являющегося пустым множеством, существует единственная антицепь из $\cap$-неразложимых интервалов, являющаяся $\cap$-разложением этого интервала:*

$$\forall I \in I(S) : I \neq \emptyset \Rightarrow \exists! A \subseteq I^{\cap}(S) : A - \text{антицепь и } I = \cap A \tag{4}$$

Очевидно, что $I^{\cap}_{min}(I)$ является такой антицепью.

4. Решетка описаний интервалов признакового пространства

Фиксируется некоторый способ описания интервалов признакового пространства. При этом, $d(I)$ – это обозначение описания интервала I, $I(d)$ – обозначение интервала, определяемого описанием d. Описание интервала рассматривается как предикат, определенный на пространстве S. Если сокращенная интервальная структура области H известна, то дизъюнкция описаний интервалов семейства $I_m(H)$ определяет сокращенную дизъюнктивную нормальную форму области H:

$$D_c(H) = \bigvee_{I \in I_m(H)} d(I) .$$

Считается, что знание сокращенной ДНФ одной из областей признакового пространства достаточно для определения сокращенной ДНФ другой области.

$D(S)$-обозначение семейства описаний всех интервалов признакового пространства S. Способ описания интервалов выбирается так, что каждому интервалу соответствует единственное его описание и каждое описание определяет единственный интервал, то есть соответствие между семействами $D(S)$ и $I(S)$ является взаимно однозначным.

На множестве $D(S)$ определяется отношение предшествования «$\leq$», порождаемое отношением предшествования «$\subseteq$», определенным на множестве интервалов $I(S)$:

$$d(I) \leq d(J), \text{если} I \supseteq J \tag{5}$$

Отношение «$\leq$» определяет частичный порядок на элементах семейства $D(S)$.

В силу определения $D(S)$ и способа определения отношения порядка «$\leq$» на этом множестве, $D(S)$ и $I(S)$ – двойственные множества. Поэтому $(D(S), \leq)$ – полная решетка. В этой решетке нижняя грань $d(I) \sqcap d(J)$ является описанием наименьшего интервала, содержащего объединение интервалов I и J, верхняя грань $d(I) \sqcup d(J)$ – это описание интервала, являющегося пересечением интервалов I и J.

Наименьший элемент этой решетки – описание признакового пространства, рассматриваемого как интервал. Наибольший элемент этой решетки – описание пустого множества, рассматриваемого как интервал.

$D^{\sqcup}(S) \subseteq D(S)$ – подмножество $\sqcup$-неразложимых описаний семейства $D(S)$.

$D^{\sqcup}(d) = \{m \in D^{\sqcup}(S) |\ m \leq d\}$ – подмножество $\sqcup$-неразложимых описаний, предшествующих описанию d.

$D^{\sqcup}_{max}(d) = max\ D^{\sqcup}(d)$ – антицепь из максимальных $\sqcup$-неразложимых описаний, предшествующих описанию d.

В силу отмеченной двойственности, в решетке $(D(S), \leq)$ справедливо утверждение, двойственное утверждению 2:

Утверждение 3. *Для каждого описания семейства $D(S)$, не являющегося описанием пустого множества, существует единственная антицепь из $\sqcup$-неразложимых описаний, являющаяся $\sqcup$-разложением этого описания:*

$$\forall d \in D(S) : I(d) \neq \emptyset \Rightarrow \exists! A \subseteq D^{\sqcup}(S) : A - \text{антицепь и } d = \sqcup\ A \tag{6}$$

Очевидно, что $D^{\sqcup}_{max}(d)$ является такой антицепью.

5. Соответствия Галуа и оператор замыкания

Объекту множества G соответствует некоторая точка пространства S. Точка признакового пространства может рассматриваться как одноточечный интервал и, следовательно, ей соответствует некоторое описание из множества $D(S)$. Поэтому, можно считать, что каждому объекту g множества G соответствует описание $d(g) \in D(S)$ одноточечного интервала, порождаемого проекцией объекта в признаковое пространство.

$P(G)$ – множество всех подмножеств множества G. Рассматриваются отображения $\varphi : P(G) \to D(S)$ и $\psi : D(S) \to P(G)$, которые определены следующим образом.

$\varphi(A)$-наибольшее (по отношению предшествования «$\leq$») интервальное описание, предшествующее описаниям элементов подмножества A:

$$\varphi(A) = \sqcap_{g \in A} d(g), \tag{7}$$

$\psi(A)$-подмножество тех элементов множества G, описания которых следуют за описанием d (проекции которых содержатся в интервале $I(d)$)

$$\psi(d) = \{g \in G |\ d \leq d(g)\}, \tag{8}$$

Пара отображений (φ, ψ) представляет собой соответствие Галуа [3] между решетками $(P(G), \subseteq)$ и $(D(S), \leq)$, а тройка $(P(G), D(S), \varphi)$ может рассматриваться как решеточный контекст [2].

Композиция отображений (φ, ψ) определяет отображение γна множестве $D(S)$:

$$\gamma = \varphi \circ \psi : D(S) \to D(S) \tag{9}$$

Отображение γ действует следующим образом: $\gamma(d)$ – это описание наименьшего (по включению подмножеств) интервала, содержащего те же точки «занятой» области пространства S, что и интервал, определяемый описанием d.

Так определенное отображение γ является оператором замыкания [3], то есть для выполняются следующие свойства:

a)	$d_1 \leq d_2 \Rightarrow \gamma(d_1) \leq \gamma(d_2)$	монотонность
b)	$d \preceq \gamma(d)$	экстенсивность
c)	$\gamma(\gamma(d)) = \gamma(d)$	идемпотентность

Элемент $d \in D(S)$ γ-замкнут, если $\gamma(d) = d$. Оператор γ порождает на элементах семейства $D(S)$ отношение γ-равносильности $R_\gamma : d_1 R_\gamma d_2 \Leftrightarrow \gamma(d_1) = \gamma(d_2)$. Отношение R_γ – это отношение эквивалентности. Оно разбивает множество интервальных описаний $D(S)$ на классы. В один класс попадают элементы, γ-замыкания которых совпадают: $[d] = \{e \in D(S) |\ \gamma(e) = \gamma(d)\}$. Такой класс, вместе с унаследованным отношением предшествования «$\leq$», является упорядоченным множеством, в котором существует единственный максимальный элемент. Это γ-замкнутый элемент класса. Минимальный элемент класса называется ключевым элементом класса. Ключевых элементов может быть несколько.

Понятие «ключевой элемент» используется в некоторых алгоритмах анализа формальных понятий [1]. Далее будет представлен один из алгоритмов поиска ключевых элементов, модифицированный для случай решетки интервальных описаний.

Ключевые элементы того класса, наибольший элемент которого является единицей решетки $(D(S), \leq)$, – это и есть описания максимальных интервалов «свободной» области. Можно воспользоваться алгоритмом, который отыскивает все ключевые элементы, порождаемые оператором γ, и, затем, среди найденных ключевых элементов отобрать те, для которых $\gamma(d) = 1$.

6. Некоторые свойства ключевых элементов конечной решетки

$K \subseteq D(S)$ – множество ключевых элементов решетки.

$K^{\sqcup} = D^{\sqcup}(S) \sqcap K$ – подмножество $\sqcup$-неразложимых ключевых элементов решетки.

$K(d) = \{k \in K |\ k < d\}$ – подмножество ключевых элементов, строго предшествующих элементу d.

$K_{max}(d) = max\ K(d)$ – антицепь из максимальных ключевых элементов, строго предшествующие элементу d.

$K^{\sqcup}_{max}(d) = max\ \{k \in K^{\sqcup} |\ k \leq d\}$-антицепь из максимальных $\sqcup$-неразложимых ключевых элементов, предшествующие элементу d.

Утверждение 4. *Элемент решетки является ключевым элементом тогда и только тогда, когда его γ-замыкание не совпадает с γ-замыканием ни одного из максимальных строго предшествующих ему ключевых элементов:*

$$d \in K \Leftrightarrow \gamma(d) \neq \gamma(k), \forall k \in K_{max}(d). \tag{10}$$

Это очевидное следствие определения ключевого элемента.

Теорема 1. *Каждый $\sqcup$-разложимый ключевой элемент конечной решетки представим в виде верхней грани строго предшествующих ему ключевых элементов:*

$$\forall d \in K \setminus K^{\sqcup} : d = \sqcup K(d). \tag{11}$$

Доказательство. Пусть $d \in K \setminus K^{\sqcup}$ – $\sqcup$-разложимый ключевой элемент решетки.

Верхнюю грань подмножества $K(d)$ ключевых элементов решетки, строго предществующих элементу d, обозначим k:

$k = \sqcup K(d)$.

По свойству верхней грани, $k \leq d$ и, следовательно, по свойству монотонности оператора замыкания

$$\gamma(k) \leq \gamma(d), \tag{12}$$

и по свойству верхней грани для сравнимых элементов

$$k \sqcup d = d. \tag{13}$$

Допустим $k \neq d$. Тогда

$$k < d. \tag{14}$$

Подмножество $\sqcup$-неразложимых элементов конечной решетки является $\sqcup$-плотным подмножеством этой решетки [3] и, следовательно, каждый элемент d решетки представим в виде $d = d \sqcup D^{\sqcup}(d)$.

Отсюда, $d \sqcup k = \sqcup\ D^{\sqcup} \sqcup k$ и

$d = \sqcup\ D^{\sqcup}(d) \sqcup k$	учтено равенство 13
$= m_1 \sqcup ...m_n \sqcup k$	считаем, что $D^{\sqcup}(d) = \{m_1, ..., m_n\}$
$\leq \gamma(m_1) \sqcup ... \sqcup \gamma(m_n) \sqcup \gamma(k)$	экстенсивность оператора замыкания
$= \gamma(m_1) \sqcup ...\gamma(m_n) \sqcup \gamma(k)$	учтено, что $\gamma(k_j) = \gamma(m_j)$, где k_j-ключевой элемент того класса, в состав которого входит $m_j, j = 1, ..., n$.
$= \gamma(k)$.	учтено, что каждый $k_j \in K(d)$. Поэтому $k_j \leq k \Rightarrow \gamma(k_j) \leq \gamma(k) \Rightarrow \gamma(k_j) \sqcup \gamma(k) = \gamma(k)$.

Таким образом, $d \leq \gamma(k)$.

Из этого неравенства, в силу монотонности оператора следует, что

$$\gamma(d) \leq \gamma(\gamma(k)). \tag{15}$$

Идемпотентность оператора γ позволяет преобразовать неравенство (15) к виду

$$\gamma(k) \leq \gamma(d). \tag{16}$$

Совместное выполнение неравенств (12) и (16) влечет равенство

$$\gamma(k) = \gamma(d). \tag{17}$$

Совместное выполнение неравенства (14) и равенства (14) приводит к тому, что элемент d не является минимальным в своем классе, то есть не является ключевым элементом. Противоречие с условием. Следовательно $d = \sqcup K(d)$.

□

В (11) для вычисления верхней грани достаточно использовать только максимальные элементы подмножества $K(d)$. Поэтому справедливо

Следствие 1. Для каждого $\sqcup$-разложимого ключевого элемента конечной решетки антицепь из максимальных строго предшествующих ему ключевых элементов является $\sqcup$-разложением этого элемента:

$$\forall d \in K \setminus K^{\sqcup} : d = \sqcup\, K_{max}(d). \tag{18}$$

Следствие 2. Для каждого $\sqcup$-разложимого ключевого элемента конечной решетки антицепь из максимальных $\sqcup$-неразложимых строго предшествующих ему ключевых элементов является $\sqcup$-разложением этого элемента:

$$\forall d \in K \setminus K^{\sqcup} : d = \sqcup\, K^{\sqcup}_{max}(d). \tag{19}$$

Примечание 2. Для $\sqcup$-неразложимого ключевого элемента решетки в качестве такой антицепи рассматривается одноэлементное подмножество, элементом которого является сам этот ключевой элемент.

Для решетки интервальных описаний $(D(S), \leq)$ следствие 2, с учетом утверждения 3 и примечания 2, позволяет сформулировать следующее

Утверждение 5. *Для каждого ключевого описания решетки $(D(S), \leq)$, не являющегося описанием пустого множества, существует единственная антицепь из $\sqcup$-неразложимых ключевых описаний, являющаяся $\sqcup$-разложением этого описания:*

$$\forall d \in K : I(d) \neq \emptyset \Rightarrow \exists!\ A \subseteq K^{\sqcup} : A - \text{антицепь и } d = \sqcup\, A. \tag{20}$$

Очевидно, что $K^{\sqcup}_{max}(d)$ является такой антицепью.

7. Ключевые антицепи

Подмножество B непосредственно предшествует (по включению) множеству A (обозначение «$\prec$»), если B является подмножеством множества A и мощность B на единицу меньше мощности A:

$B \prec A \Leftrightarrow B \subset A$ и $\mid B \mid = \mid A \mid -1$.

Определение 1. Состоящая из $\sqcup$-неразложимых ключевых элементов антицепь, верхняя грань которой является ключевым описанием, называется ключевой антицепью.

$A(K^{\sqcup})$-семейство всех антицепей из $\sqcup$-неразложимых ключевых элементов семейства $D(S)$.

$KA(K^{\sqcup}) = \{A \in A(K^{\sqcup}) | \sqcup A \in K\}$-семейство всех ключевых антицепей.

Утверждение 6. *Если d – $\sqcup$-разложимый ключевой элемент, k-максимальный ключевой элемент, строго предшествующий элементу d, то ключевая антицепь элемента k непосредственно предшествует ключевой антицепи элемента d:*

$d \in K \setminus K^{\sqcup}$ и $k \in K_{max}(d) \Rightarrow K^{\sqcup}_{max}(k) \prec K^{\sqcup}_{max}(d)$.

Утверждение 7. *Если антицепь X, состоящая из $\sqcup$-неразложимых ключевых элементов, не является ключевой антицепью, то существует ее строгое подмножество, которое является ключевой антицепью, порождающей ключевой элемент того же класса, которому принадлежит элемент $\sqcup X$:*

$$X \in A(K^{\sqcup}) \text{ и } \sqcup X \notin K \Rightarrow \exists Z \subset X : Z \in KA(K^{\sqcup}) \text{ и } \gamma(\sqcup Z) = \gamma(\sqcup X). \qquad (21)$$

Доказательство. Пусть $X \in A(K^{\sqcup})$ и $\sqcup X \notin K$. Тогда существует ключевой элемент k, который строго предшествует $\sqcup X$ и находится в том же классе, что и $\sqcup X$.

$Z = k^{\triangledown} \cap X$-антицепь из всех максимальных $\sqcup$-неразложимых ключевых элементов, предшествующих ключевому элементу k. Следовательно, в силу утверждения 5, Z является ключевой антицепью для элемента k. □

8. Порядковый идеал ключевых антицепей

Теорема 2. *Семейство ключевых антицепей $KA(K^{\sqcup})$ является порядковым идеалом на $(P(K^{\sqcup}), \subseteq)$, то есть, если $Y \in KA(K^{\sqcup})$ и $X \subseteq Y$, то $X \in KA(K^{\sqcup})$.*

Доказательство. Пусть $Y \in KA(K^{\sqcup})$ и $X \subseteq Y$. Допустим, что $X \notin KA(K^{\sqcup})$. Тогда, в соответствии с утверждением 5,

$$\exists Z \subset X : Z \in KA(K^{\sqcup}) \text{ и } \gamma(\sqcup Z) = \gamma(\sqcup X). \qquad (22)$$

Покажем, что в этом случае $\gamma(\sqcup Y) = \gamma(\sqcup(Y \setminus (X \setminus Z)))$. Для этого понадобятся несколько вспомогательных утверждений.

$$1. \quad X \setminus Z \subseteq X \Rightarrow \gamma(\sqcup(X \setminus Z)) \leq \gamma(\sqcup Z). \qquad (23)$$

Доказательство.

$X \setminus Z \subseteq X \Rightarrow$
$\Rightarrow \sqcup(X \setminus Z) \leq \sqcup X$ свойство верхней грани
$\Rightarrow \gamma(\sqcup(X \setminus Z)) \leq \gamma(\sqcup X)$ монотонность γ
$\Rightarrow \gamma(\sqcup(X \setminus Z)) \leq \gamma(\sqcup Z)$ учтено (22) □

$$2. \quad Z \subseteq Y \setminus (X \setminus Z) \Rightarrow \gamma(\sqcup Z) \leq \gamma(\sqcup(Y \setminus (X \setminus Z))) \qquad (24)$$

Доказательство.

$Z \subseteq Y \setminus (X \setminus Z) \Rightarrow$
$\Rightarrow \sqcup Z \leq \sqcup(Y \setminus (X \setminus Z))$ свойство верхней грани
$\Rightarrow \gamma(\sqcup Z) \leq \gamma(\sqcup(Y \setminus (X \setminus Z)))$ монотонность γ □

$$3. \quad \sqcup Y \leq \gamma(\sqcup(Y \setminus (X \setminus Z))) \qquad (25)$$

Доказательство.

$Y = (Y \setminus (X \setminus Z)) \cup (X \setminus Z) \Rightarrow$

$\sqcup Y = \sqcup((Y \setminus (X \setminus Z)) \cup (X \setminus Z))$

$= \sqcup(Y \setminus (X \setminus Z)) \sqcup (\sqcup(X \setminus Z))$ ассоциативность $\sqcup$

$\leq \gamma(\sqcup(Y \setminus (X \setminus Z))) \sqcup \gamma(\sqcup(X \setminus Z))$ монотонность γ

$\leq \gamma(\sqcup(Y \setminus (X \setminus Z))) \sqcup \gamma(\sqcup Z)$ учтено (22)

$\leq \gamma(\sqcup(Y \setminus (X \setminus Z))) \sqcup \gamma(\sqcup(Y \setminus (X \setminus Z)))$ учтено (24)

$= \gamma(\sqcup(Y \setminus (X \setminus Z)))$ □

4. $$\sqcup Y \leq \gamma(\sqcup(Y \setminus (X \setminus Z))) \Rightarrow \gamma(\sqcup Y) \leq \gamma(\sqcup(Y \setminus (X \setminus Z))) \quad (26)$$

Доказательство.

$\sqcup Y \leq \gamma(\sqcup(Y \setminus (X \setminus Z))) \Rightarrow$

$\Rightarrow \gamma(\sqcup Y) \leq \gamma(\gamma(\sqcup(Y \setminus (X \setminus Z))))$ монотонность γ

$\Rightarrow \gamma(\sqcup Y) \leq \gamma(\sqcup(Y \setminus (X \setminus Z)))$ идемпотентность γ □

5. $$Y \setminus (X \setminus Z) \subseteq Y \Rightarrow \gamma(\sqcup(Y \setminus (X \setminus Z))) \leq \gamma(\sqcup Y) \quad (27)$$

Доказательство.

$Y \setminus (X \setminus Z) \subseteq Y \Rightarrow$

$\Rightarrow \sqcup(Y \setminus (X \setminus Z)) \leq \sqcup Y$ свойство верхней грани

$\Rightarrow \gamma(\sqcup(Y \setminus (X \setminus Z))) \leq \gamma(\sqcup Y)$ монотонность γ □

Совместное выполнение неравенств (26) и (27) влечет равенство

$$\gamma(\sqcup(Y \setminus (X \setminus Z))) = \gamma(\sqcup Y) \quad (28)$$

Так как $Y \setminus (X \setminus Z) \subset Y$, то равенство (28) означает, что Y не является ключевой антицепью. Противоречие с условием теоремы. □

Тот факт, что $KA(K^{\sqcup})$ является порядковым идеалом на $(P(K^{\sqcup}), \subseteq)$, влечет все нижеследующие определения и утверждения текущего раздела.

$A_{-}(A) = \{B \in A(K^{\sqcup}) | B \prec A\}$ – подмножество антицепей из $\cap$-неразложимых ключевых элементов, непосредственно предшествующих антицепи A.

Определение 2. Антицепь $A \in A(K^{\sqcup})$ называется кандидатом (кандидатом в ключевую антицепь), если все ее строгие подмножества являются ключевыми антицепями и ее верхняя грань не является описанием пустого множества.

Утверждение 8. *Антицепь $A \in A(K^{\sqcup})$ является кандидатом тогда и только тогда, когда каждое непосредственно предшествующее ей подмножество является ключевой антицепью и ее верхняя грань не является описанием пустого множества:*

$A \in A(K^{\sqcup})$ – кандидат $\Leftrightarrow A_{-}(A) \subseteq KA(K^{\sqcup})$ и $\sqcup A \neq d(\emptyset)$.

Утверждение 9. *Антицепь-кандидат является ключевой антицепью тогда и только тогда, когда γ-замыкание ее верхней грани не совпадает с γ-замыканием верхней грани ни одной из непосредственно предшествующей ей антицепей.*

Определение 3. Подмножество $A \in P(K^{\sqcup})$ называется подмножеством-потенциальным кандидатом, если хотя бы одно из непосредственно предшествующих ему подмножеств является ключевой антицепью:

$A \in P(K^{\sqcup})$ – подмножество-потенциальный кандидат $\Leftrightarrow A_{-}(A) \cap KA(K^{\sqcup}) \neq \emptyset$.

Определение 4. Антицепь $A \in A(K^{\sqcup})$ называется антицепью-потенциальным кандидатом, если хотя бы одно из непосредственно предшествующих ему подмножеств является ключевой антицепью:

$A \in A(K^{\sqcup})$ – антицепь-потенциальный кандидат $\Leftrightarrow A_{-}(A) \cap KA(K^{\sqcup}) \neq \emptyset$.

Примечание 3. Требование $\sqcup A \neq d(\emptyset)$ включается в определения 2, 3 или 4 в зависимости от того, где удобнее или проще вычислить $\sqcup A$. Кроме этого, выражение $\sqcup A \neq d(\emptyset)$ часто удобнее записывать в виде $I(\sqcup A) \neq \emptyset$.

9. Алгоритм поиска ключевых антицепей

Обозначения алгоритма

Обозначение	Пояснение
i	Номер уровня, номер итерации
SC_i	Семейство подмножеств – потенциальных кандидатов i-го уровня
AC_i	Семейство антицепей – потенциальных кандидатов i-го уровня
C_i	Семейство антицепей – кандидатов i-го уровня
K_i	Семейство ключевых антицепей i-го уровня

Псевдокод алгоритма

Вход.	S – признаковое пространство;
	$H_{\bar{0}} \subseteq S$ – проекция множества объектов G в пространство S.
Выход.	$KA(K^{\sqcup})$ – семейство всех ключевых антицепей.
Метод.	
1)	$K_0 := \{d(S)\}$
2)	$C_1 := \{A = \{d\} \mid d \in D^{\sqcup}(S)$ и $I(\sqcup A) \neq \emptyset\}$
3)	$K_1 := \{A \in C_1 \mid \gamma(\sqcup A) \neq \gamma(\sqcup B), \forall B \in K_{max}(A)\}$
4)	$i := 1$
5)	Цикл пока $K_i \neq \emptyset$
6)	$i++$
7)	$SC_i := \{A = \{d_1, ..., d_i\} \subseteq K^{\sqcup} \mid A_{-}(A) \cap K_{i-1} \neq \emptyset\}$
8)	$C_i := \{A \in AC_i \mid A_{-}(A) \subseteq K_{i-1}$ и $I(\sqcup A) \neq \emptyset\}$
9)	$K_i := \{A \in C_i \mid \gamma(\sqcup B) \neq \gamma(\sqcup A), \forall B \in A_{-}(A)\}$
10)	Конец цикла
11)	Возврат $\bigcup_{j=0}^{i-1} K_j$

10. Пояснения к псевдокоду алгоритма

Шаг 1. Описание признакового пространства рассматривается как одноэлементная ключевая антицепь.

Шаг 2. Все $\sqcup$-неразложимые описания, не являющиеся описаниями пустого множества, рассматривают как кандидаты (определение 2).

Шаг 3. Те кандидаты, γ-замыкания верхних граней которых не совпадает с γ-замыканием верхней грани ни одного из строго предшествующих им максимальных ключевых элементов, являются ключевыми элементами (утверждение 4).

Шаг 4. Инициализируется счетчик итераций (счетчик уровней).

Шаг 5. Условие $K_i \neq \emptyset$ определяет выполнение условия очередной итерации алгоритма, реализующей построение необходимых конструкций следующего уровня из ключевых антицепей предшествующего уровня.

Шаг 6 определяет номер следующей итерации (номер следующего уровня).

Шаг 7 формирует семейство подмножеств – потенциальных кандидатов, непосредственно следующих за ключевыми антицепями семейства K_{i-1} (определение 3).

Шаг 8 отбирает из подмножеств – потенциальных кандидатов те, которые являются антицепями, формируя семейство антицепей – потенциальных кандидатов i-го уровня (определение 4).

Шаг 9 отбирает из антицепей – потенциальных кандидатов те, которые являются кандидатами (утверждение 8).

Шаг 10 отбирает из кандидатов тех, которые являются ключевыми антицепями (утверждение 9).

Шаг 12 объединяет результаты, полученные на разных уровнях, и возвращает это объединение как результат работы.

Примечание 4. На шаге 3 требуется знать множество $K_{max}(A)$. Здесь антицепь $A = \{d\}$ является одноэлементным подмножеством, состоящим из $\sqcup$-неразложимого интервального описания. Поэтому, для этой антицепи, множество $K_{max}(A)$ состоит из одного максимального ключевого элемента, строго предшествующего элементу d.

Для проверки условия этого шага удобно выписать все максимальные цепи, порождаемые $\sqcup$-неразложимыми описаниями $D^{\sqcup}(S)$ семейства $D(S)$. Тогда очевидно, что $K_{max}(A)$ состоит из максимального ключевого описания, строго предшествующего элементу d в цепи, содержащей d.

После шага 3 полученное множество K_1 – это фактически $K^{\sqcup}$ – множество $\sqcup$-неразложимых ключевых описаний.

Примечание 5. Процедура генерации элементов семейства SC_i (шаг 7) здесь не фиксируется. Можно применять любой вариант, обеспечивающий получение семейства SC_i, например, может быть применена процедура, использующая тот или иной линейный порядок на элементах множества $\sqcup$-неразложимых ключевых описаний [1].

Заключение

В статье рассмотрена решетка описаний интервалов признакового пространства. Доказана теорема о представимости каждого $\sqcup$-разложимого ключевого элемента этой решетки в виде верхней грани строго предшествующих ему ключевых

элементов. Следствием этой теоремы является утверждение о единственности $\sqcup$–разложения ключевых элементов, не являющихся описанием пустого множества, в антицепь из $\sqcup$-неразложимых ключевых элементов.

Приведено определение понятия ключевой антицепи и исследованы свойства семейства ключевых антицепей. Вчастности показано, что семейство ключевых антицепей является порядковым идеалом на множестве-степени $\sqcup$-неразложимых ключевых элементов, упорядоченных отношением включения.

Предложен алгоритм поиска всех ключевых антицепей решетки интервальных описаний.

Список литературы

1. *Ganter, B., Wille, R.* Formal Concept Analysis – Mathematical Foundations. Springer-Verlag, Berlin., 1999.
2. *Gugisch, R.* Lattice Contexts – a Generalization in Formal Concept Analysis. Handuot to ICCS 2000, Darmstadt(2000) http://www.mathe2.uni-bayreuth.de/ralfg/papers/diplom.ps.gz.
3. *Биркгоф Г.* Теория решеток: Пер. с англ. – М.: Наука, 1984. – 568 с.
4. *Ильченко А.В.* Компактная компонентная и сокращенная интервальная структуры признакового пространства, порождаемые эмпирическими данными // Сб. Таврический вестник информатики и математики. – 2005. – №2. – С. 126-142.

Статья поступила в редакцию 30.04.2008

УДК 519.7

МНОГОМЕРНОЕ ШКАЛИРОВАНИЕ В СЛУЧАЕ МАТРИЦ ПОПАРНЫХ РАССТОЯНИЙ С ЭЛЕМЕНТАМИ ИЗ КОНЕЧНОГО МНОЖЕСТВА

© Иофина Г.В.

Московский физико-технический институт (государственный университет)
факультет управления и прикладной математики
РФ, г. Долгопрудный, Институтский переулок, 9, 141700

e-mail: *giofina@gmail.com*

Abstract. The paper considers multidimensional scaling problem with proximity matrix consisting of two different non-diagonal elements. Elements of the superdiagonal matrix don't decrease along the string and don't increase along the column. The structure of considered matrix is described. In one-, two- and three-dimensional spaces all available proximity matrices are listed. Matrices that can be considered as proximity matrices in t-dimensional euclidian space have their dimensions in a definite interval. The exact bounds of this interval are also found.

Введение

Многомерное шкалирование – один из методов анализа данных, основанный на исследовании близостей [1]. Под понятием близости часто понимаются сходства или различия между объектами в евклидовом пространстве. В простейшем случае в качестве близости между объектами берется расстояние между ними. Задачей многомерного шкалирования является поиск представления матрицы близости системой точек в евклидовом пространстве некоторой размерности.

В некоторых случаях можно считать, что расстояния между объектами принимают одно из двух значений. Это можно сделать, например, когда важен факт близости или дальности объектов, а не их точные значения. Тогда средние расстояния между близкими объектами можно считать равным 1, а далекими – M.

Известно, что в данном случае максимальное число точек, которое можно расположить наилучшим образом в пространстве размерности $t = 1$, равно трем, для $t = 2$ – пяти (точки располагаются в вершинах правильного пятиугольника), для $t = 3$ — шести (точки находятся в вершинах правильного октаэдра или правильной треугольной призмы, или правильного пятиугольника и соответствующей точке на его оси). Точных значений для $t \geq 4$ неизвестно, но для них известны довольно точные границы: $t(t+1)/2 \leq N \leq (t+1)(t+2)/2$. Для случая расположения точек на сфере в пространстве размерности $t \in \{2, 6, 22\}$ Дельсартом было показано, что можно разместить $t(t+3)/2$ точек [2].

Может оказаться, что объекты можно линейно упорядочить по убыванию по удаленности вначале от первого объекта, потом от второго и т. д. Это накладывает на матрицы попарных расстояний условие порядка – неубывания по строкам и невозрастания по столбцам элементов матрицы.

Очевидно, что условию порядка подчиняются не все матрицы попарных расстояний, удовлетворяющие аксиомам метрик. В первой части работы находится общий

вид матриц попарных расстояний, элементы которых удовлетворяют условию порядка. Во второй части описаны все возможные матрицы попарных расстояний в евклидовых пространствах размерностей $t = 1, 2, 3$. В заключительной части работы находятся ограничения на количество объектов, которые можно разместить в пространстве размерности t.

1. Структура матриц

Определение 1. Будем говорить, что элементы матрицы $A = \{a_{ij}\}$ размерности $N \times N$ удовлетворяют аксиомам метрики, если они удовлетворяют следующим условиям:

1. $a_{ij} = 0 \Leftrightarrow i = j$,
2. $a_{ij} = a_{ji}$,
3. $a_{ij} + a_{jk} \geq a_{ik}$, $\forall i, j, k = 0, \ldots, N-1$.

Рассмотрим матрицу $A = \{a_{ij}\}$, элементы которой удовлетворяют условию порядка и первым двум аксиоам метрики, то есть она симметричная, с нулевой диагональю, и для элементов, находящихся выше главной диагонали, выполнены условия

$$a_{ij} < a_{ik} \forall i, j, k : i > j, j < k;$$
$$a_{ij} > a_{kj} \forall i, j, k : i > j, i < k.$$

Далее будем рассматривать элементы, находящиеся выше главной диагонали. Элементы, находящиеся ниже главной диагонали, определяются автоматически из-за симметричности матрицы.

Условия означают, что в каждой строке после диагонального нуля стоит какое-то количество единиц, а потом идут элементы, равные M. Причем, если в строке i первая тройка стоит на l-ом месте, то в строке $j > i$ первая тройка будет стоять на месте с номером $k \geq l$.

Наложим на элементы матрицы выполнение третьей аксиомы метрики – неравенство треугольника.

Первый случай. $M > 2$.

Пусть в первой строке матрицы первые l элементов равны единицам, а остальные M, то есть $a_{0i} = 1, \forall i \leq l$ и $a_{0i} = M, \forall i > l$.

Так как $a_{0i} = a_{0j} = 1$, $\forall i, j \leq l$, то для выполнения неравенств треугольника должно быть выполнено $a_{0i} + a_{0j} = 1 + 1 \geq a_{ij} \in \{1, M\}$. Поэтому $a_{ij} = 1$.

Если $i \leq l$, $j > l$, то есть $a_{0i} = 1$, $a_{0j} = M$, то для выполнения неравенств треугольника необходимо, чтобы $a_{0i} + a_{ij} = 1 + a_{ij} \geq a_{0j} = M$. Так как $a_{ij} \in \{1, M\}$, то $a_{ij} = M$.

Поэтому, если в строке есть единица, то она полностью определена. Следовательно, определена первая $l + 1$ строка . В них $\forall i \leq l$ $a_{ji} = 1$ и $\forall i > l$ $a_{ji} = M$. По симметрии определяются первые $l + 1$ столбцов.

Остальные элементы матрицы можно определять независимо от первых $l + 1$ строк (столбцов). Действительно, $\forall i > l + 1$ для выполнения неравенств треугольника с элементами из первых строк необходимо, чтобы

$a_{ki} + a_{kj} = M + M = 2M \geq a_{ij} \in \{1, M\}$, что будет выполняться для любых a_{ij} (так как $a_{ki} + a_{ij} = M + a_{ij} \geq a_{kj} = M$ верно для любых $a_{ij} \in \{1, M\}$).

Следовательно, можно выкинуть определенные строки и столбцы и рассматривать матрицу размером $N - l - 1$. Таким образом можно определить всю матрицу.

Причем число матриц размерности $M \geq 3$ может быть определено, как $f(M) = \sum\limits_{k=0}^{M-1} f(M-1-k) = \sum\limits_{k=0}^{M-1} f(k)$, где $f(0) = f(1) = 0$, $f(2) = 2$, $f(3) = 4$. Или в явном виде: $f(M) = 6 \cdot 2^{M-3}$, $\forall M \geq 4$.

Второй случай. $M \leq 2$. Единственный невырожденный случай это $M = 2$. В данном случае $a_{ij} \in \{1, 2\}$. Любая комбинация трех элементов из этого множества не будет нарушать неравенство треугольника.

Число матриц в данном случае $f(2) = \frac{C_{2\cdot 3}^{3}}{3+1} = 5$.

2. Матрицы, задающие конфигурации в пространствах размерности $t = 1, 2, 3$

Итак, рассматриваемые матрицы имеют структуру, состоящую из единичных блоков, расположенных по диагонали, которые отделены друг от друга строками, состоящими из элементов, равных M (кроме нулевых диагональных элементов). Такая структура дает в евклидовом пространстве кластеризацию близких точек, имеющих структуру правильных симплексов. Точки из различных кластеров находятся на расстояниях равных M.

Найдем всевозможные матрицы, удовлетворяющие рассматриваемым условиям, которые задают расположения объектов на прямой, плоскости и в пространстве. Вначале заметим, что для пространства произвольной размерности выполнена

Теорема 1. *Если в пространстве размерности $t > 1$ есть правильный симплекс, состоящий из $t+1$ точки, то в пространство можно добавить только одну точку, отстоящую от других на равных расстояниях, и эта точка – центр симметрии симплекса.*

Доказательство. Действительно, точка должна отстоять ото всех вершин фигуры на одинаковых расстояниях, следовательно, находиться в центре описанной сферы (окружности в двумерном случае) данного симплекса. Из геометрии известно, что, если около фигуры можно описать сферу, то это можно сделать единственным образом. Также известно, что около правильного многогранника, которым является правильный симплекс, можно описать сферу. □

2.1. **Пространство размерности $t = 1$.** Так как максимальный симплекс в пространстве размерности $t = 1$ состоит из двух точек, то возможны только следующие случаи.

- **Максимальный симплекс в рассматриваемой конфигурации состоит из 2 точек.**
  - Два максимальных симплекса. На прямой это возможно только когда одна точка находится от других на расстояниях равных 1, то есть симплексы пересекаются. Тогда расстояние между оставшимися точками $M = 2$.

Действительно, если бы симплексы не пересекались, на прямой располагалось бы 4 точки. Их можно пронумеровать слева направо. Все расстояния от первой точки до остальных трех различны, что не удовлетворяют условию.
Отсюда также следует, что число точек, которое можно расположить на прямой не превышает 3.

– Один максимальный симплекс. Две точки находятся на расстоянии 1. К ним можно добавить только одну точку, находящуюся на одинаковых расстоянии от двух других. Это единственная точка, находящаяся в середине отрезка. В этом случае расстояние $M = \frac{1}{2}$.

- **Максимальный симплекс в рассматриваемой конфигурации состоит из 1 точки.** Все точки находятся на расстояниях равных M. Всего на прямой можно расположить только две таких точки.

2.2. **Пространство размерности $t = 2$.** Так как максимальный симплекс в пространстве размерности $t = 2$ состоит из 3 точек, то возможны только следующие случаи.

- **Максимальный симплекс в рассматриваемой конфигурации состоит из 3 точек.** Максимальный симплекс единственнен по теореме 1. К нему можно добавить единственную точку, находящуюся на одинаковых расстояниях от других точек. Это центр описанной около треугольника окружности, отстоящий на расстоянии $M = \frac{\sqrt{3}}{3}$ от вершин.
- **Максимальный симплекс в рассматриваемой конфигурации состоит из 2 точек.**

  – Два максимальных симплекса. Расположим две точки на расстоянии 1 друг от друга. Остальные точки находятся на одинаковых расстояниях от данных точек, то есть на серединном перпендикуляре отрезка, соединяющего эти точки. На нем можно расположить только 2 точки, отстоящие на расстоянии $M \neq 1$, симметрично отрезку-симплексу (если бы расстояние равнялось 1, то максимальный симплекс состоял бы из 3 точек). Добавление еще одной точки даст минимум одно дополнительное расстояние, что не будет согласовано с условием. Если эти две точки составляют симплекс, то из простых геометрических соображений можно получить, что $M = \frac{\sqrt{2}}{2}$.
  – Один максимальный симплекс. Точки, находящиеся на серединном перпендикуляре, находятся на расстоянии 1 от вершин первого симплекса и на расстоянии $M = \frac{\sqrt{3}}{3}$ друг от друга.

- **Максимальный симплекс в рассматриваемой конфигурации состоит из 1 точки.** Все точки находятся на расстоянии M. Всего на плоскости можно расположить три таких точки, находящихся в вершинах равностороннего треугольника.

2.3. **Пространство размерности** $t = 3$**.** Так как максимальный симплекс в пространстве размерности $t = 3$ состоит из 4 точек, то возможны только следующие случаи.

- **Максимальный симплекс в рассматриваемой конфигурации состоит из 4 точек.** Максимальный симплекс единственнен по теореме 1. К нему можно добавить единственную точку, отстоящую на одинаковых расстояниях от других точек. Это центр сферы, описанной около тетраэдра и отстоящий на расстоянии $M = \frac{\sqrt{6}}{4}$ от вершин.
- **Максимальный симплекс в рассматриваемой конфигурации состоит из 3 точек.** Три точки, находящиеся на расстоянии 1, образуют правильный треугольник. Остальные точки могут находиться только на прямой, проходящей через его центр и перпендикулярной плоскости треугольника. Поэтому можно добавить только две точки, симметричные относительно плоскости треугольника. Добавление еще одной точки даст минимум одно дополнительное расстояние, что не будет согласовано с условием. Эти точки могут находиться на расстоянии 1, тогда они находятся на расстоянии $M = \sqrt{\frac{7}{12}}$ от вершин первого симплекса. Или они могут находиться на расстоянии M друг от друга и от первого симплекса, тогда из геометрических соображений $M = \frac{2}{3}$.
- **Максимальный симплекс в рассматриваемой конфигурации состоит из 2 точек.** Две точки находятся на расстоянии 1 друг от друга. Остальные точки находятся в плоскости, перпендикулярной данному отрезку на окружности радиуса $\sqrt{M^2 - 1/4}$.
  Так как максимальная размерность симплекса в расссматриваемом случае равна 2, то в плоскость можно поместить дополнительный максимальный симплекс-отрезок. Это отрезок, чьи вершины лежат на окружности радиуса $\frac{1}{2}$, чей центр проходит через серидину первого симплекса. Второе расстояние определяется автоматически и равно $M = \frac{\sqrt{2}}{2}$. Более того, заметим, что, если ко второму максимальному симплексу добавить две точки, они будут лежать на той же окружности и автоматически окажутся на расстоянии 1 друг от друга, то есть составят еще один максимальный симплекс-отрезок.
  Если в плоскость не помещать симплекс-отрезок, то точки, которые можно добавить к первому симплеску, будут находиться на равных расстояниях друг от друга, не равных 1. Максимальное число точек, которое можно поместить таким образом на окружность, равно трем и при этом $M = \sqrt{\frac{7}{12}}$.
- **Максимальный симплекс в рассматриваемой конфигурации состоит из 1 точки.** Все точки находятся на расстоянии M. Всего в пространстве можно расположить четыре таких точки, находящихся в вершинах правильного тетраэдра.

3. Максимальное число объектов в евклидовом пространстве размерности t

Из рассмотренных случаев видно, что число объектов, которые можно поместить в пространство конечной размерности так, чтобы попарные расстояния задавались матрицей близости, удовлетворяющей рассматриваемым условиям, конечно, а расстояния M иногда могут быть не произвольными, а строго определенными. Для максимального числа объектов, помещенных в пространство конечной размерности, справедлива следующая теорема.

Теорема 2. *Число точек, чья матрица попарных расстояний состоит из недиагональных элементов, принимающих значения из множества* $\{1, M\}$ *и удовлетворяющих аксиомам метрики и условию порядка, которые можно поместить в пространство размерности* t *не превышает* $2t$.

Доказательство. Для случаев $t = 1, 2, 3$ утверждение проверено (см. выше). Пусть утверждение доказано для случая $k \leq t - 1$, докажем для случая $k = t$.

I. Пусть максимальная размерность единичного блока равна $d > 1$. Тогда построим в пространстве этот симплекс. Это можно сделать в пространстве размерности $d-1 \geq 1$. Тогда остальные точки находятся на одинаковом расстоянии от точек этого блока, то есть в гиперплоскости размерности $t - (d - 1)$.

По предположению индукции, в гиперплоскости размерности $t - (d - 1)$ число точек $k \leq 2(t-(d-1))$. Тогда общее число точек в пространстве размерности t будет $k \leq d + (2(t - (d - 1)) = 2t - d + 2 \leq 2t$ при $d \geq 2$.

II. Пусть максимальная размерность единичного блока равна $d = 1$. Это означает, что в пространство нужно поместить какое-то число пар объектов, находящихся на расстоянии друг от друга, равном 1, и какое-то число объектов, находящихся на расстоянии M от всех остальных объектов.

Построим конфигурацию, состоящую из пар точек, находящихся на расстоянии 1. Расположим первую пару произвольным образом. Следующие две точки должны находиться на одинаковом расстоянии от уже построенных точек, следовательно, они должны находиться в гиперплоскости, проходящей через середину отрезка, соединяющем первую пару точек. Точки должны располагаться симметрично относительно этого отрезка и находиться друг от друга на расстоянии 1. Заметим, что размерность пространства, в котором расположены точки, при таком построении увеличилась с 1 до 2. Далее добавляем еще две точки симметрично относительно построенной конфигурации так, чтобы расстояние между ними было равно 1, и отрезок, соединяющий их был перпендикулярен двум предыдущим отрезкам, то есть их содержащей плоскости. Это, очевидно, можно сделать в пространстве размерности 3. При добавлении нового отрезка, размерность пространства увеличивается на 1. Поэтому число точек, которое можно разместить в пространстве размерности t таким образом будет равно $2t$.

Это число будет максимальным. Действительно, добавим к получившейся конфигурации одну точку, равноотстоящую от других точек. Размерность пространства увеличилась на 1, число точек также увеличилось на 1. Следующую точку можно

добавить симметрично последней точке относительно конфигурации перпендикулярных отрезков, не увеличив при этом размерности пространства. Добавление третьей точки ведет к тому, что 3 точки будут находиться на одинаковых расстояниях равных M друг от друга. Эти точки находятся в вершинах правильного симплекса в пространстве размерности 2. Для его построения требуется 3 точки, в то время как конфигурация с перпендикулярными отрезками обеспечила бы в плоскости расположение 4-ех точек. Следовательно, данная конфигурация не будет оптимальной.

□

Итак, если в евклидовом пространстве размерности t задана конфигурация точек, попарные расстояния между точками которой принимают одно из двух значений, то число точек в этой конфигурации не превышает $2t$. Если данную конфигурацию точек нельзя поместить в пространство меньшей размерности, то число точек в этой конфигурации не может быть меньше $t+1$, что соответствует случаю t-мерного симплекса. Поэтому верно следующее утверждение.

Следствие 1. *Размерность пространства t, требуемая для размещения N произвольных точек, находится в интервале* $[\lceil N/2 \rceil, N-1]$.

Заключение

В работе найден общий вид матриц, для элементов которых выполняются аксиомы метрики и условие порядка, которые могут являться матрицами близости для объектов из евклидова пространства. В пространствах размерностей $t = 1, 2, 3$ описанны все возможные матрицы близости, удовлетворяющие рассматриваемым условиям. Доказана теорема о том, что в евклидово пространство размерности t, можно поместить $N \leq 2t$ объектов, матрица попарных расстояний которых удовлетворяет рассматриваемым условиям. Получены нижняя и верхняя оценки для размерности пространства t, требуемой для размещения N точек.

В дальнейшем исследовании предполагается более глубокое изучение структуры матриц попарных расстояний для размещения объектов в пространствах размерности $t \geq 4$ и нахождение условия, когда произвольные матрицы задают конфигурации в евклидовом пространстве. Также планируется найти новые критерии для размещения объектов в евклидовых пространствах по матрице попарных расстояний, удовлетворяющей рассматриваемым свойствам, при неточном соответствии элементов матрицы расстояниям между объектами.

Список литературы

1. *Дэйвисон М.* Многомерное шкалирование. — М.: Финансы и статистика, 1988.— 254 с.
2. *Hallard T. Croft, K. J. Falconer , Richard K. Guy* Unsolved problems in Geometry. — Springer-Verlang, 1991. — 198 p.

Статья поступила в редакцию 27.04.2008

УДК 004.9

МЕТОДЫ ПОИСКА БИОМЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ В БАЗАХ ДАННЫХ ПО ИХ СОДЕРЖАНИЮ

© Ковалев В.А.

Объединенный институт проблем информатики НАН Беларуси
ул. Кирова, 32-А, 246050 Гомель, Беларусь,
лаб. Анализа биомедицинских изображений

e-mail: *vassili.kovalev@gmail.com*

Abstract. This paper is dedicated to the problem of image retrieval from large databases of biomedical images using the query by example paradigm. The specific characteristics of biomedical images are briefly discussed in the context of similarity retrieval and possible ways of image searching are outlined. A versatile method is suggested which is based on extended multi-sort and multi-dimensional co-occurrence matrices. The use of the method is illustrated on a variety of image retrieval tasks using real databases of biomedical images of different types.

Введение

Несмотря на появление и развитие разнообразных методов количественного анализа изображений для медицинской диагностики, одним из популярных диагностических приемов по-прежнему остается сравнение полученного изображения с предыдущими и/или с похожими снимками и случаями из клинической практики [1]. Технической основой для повышения эффективности диагностических процедур, основанных на оперативном привлечении информации о похожих случаях, является развитие так называемых систем архивирования и передачи медицинских изображений [2], которые известны по их англоязычной аббревиатуре PACS (Picture Archiving and Communication Systems), а так же современных технологий баз данных (БД). Однако в подавляющем большинстве случаев, существующие методы и программные средства БД обеспечивают поиск не по содержанию самих изображений, а по ассоциированным с ними метаданным. Указанные метаданные представляют собой информацию о содержании изображений БД, подготовленную вручную, а так же клиническую и другую информацию о самих пациентах. В обоих случаях такие индексирующие данные представлены в обычной фактографической, но не в визуальной (зрительной) форме.

По мере развития технических средств регистрации медицинских изображений, объем хранимых видеоданных растет экспоненциально. По данным организации «Грид Компьютинг Нау» при Министерстве торговли и промышленности Великобритании [3], полный переход на цифровые диагностические изображения приведет к тому, что суммарный объем изображений, производимых госпиталями, может достигнуть петабайт, т.е. величины 10^{15}. Естественно, что при подобных сверх-больших объемах хранимых изображений никакой речи о «ручном», т.е. визуальном поиске в БД «похожих» снимков и подборе аналогичных диагностических ситуаций речь идти не может.

Таким образом, разработка средств автоматического индексирования содержания и поиска изображений БД по заданному образцу является одной из актуальных проблем в области обработки, анализа и распознавания медицинских изображений.

1. Особенности поиска медицинских изображений

Проблема поиска изображений по образцу активно разрабатывается в области машинного зрения и распознавания образов на протяжении последнего десятка лет. За это время достигнуты определенные успехи использования данной технологии для изображений общего характера типа цветных цифровых фотографий. Одним из основных итогов исследований данного направления является выделение трех групп признаков изображений, используемых для организации поиска, которые условно определяются как текстура, цвет и форма. Медицинские же изображения значительно отличаются от изображений общего характера как минимум по следующим трем основным признакам.

1. *Несопоставимость различных классов.* Семантические и форматные различия меду классами медицинских изображений обычно столь велики, что их прямое сопоставление на растровом уровне не представляется возможным. При этом под «классами» здесь понимаются разнородные группы, отличающиеся друг от друга как диагностируемыми органами (например, снимки печени и снимки мозга), так и типом используемой модальности (например, ультразвуковые и компьютерно-томографические изображения).

2. *Небольшие внутриклассовые различия.* По сравнению со снимками общего характера, медицинские изображения, как правило, имеют гораздо более ограниченный контекст, меньшую вариабельность в композиции отображаемой сцены и, с точки зрения непрофессионала, могут выглядеть почти недифференцируемыми. Поэтому любые признаки с невысокой чувствительностью, предложенные для описания обычных изображений со значительными визуальными различиями, оказываются здесь неэффективными.

3. *Специфичность классообразующих признаков.* В отличие от обычных изображений, при поиске и распознавании которых важную роль играет их общий вид и явные визуальные отличия, ключевые признаки медицинских изображений могут носить неявный и даже скрытый характер. Например, различия между МРТ изображениями головного мозга в норме, при шизофрении и при болезни Альцгеймера определяются очень тонкими и специфическими признаками, характерными для данных видов патологии. Указанные различия никак не связаны с различиями в размерах, форме или особенностями строения извилин мозга пациентов, которые в первую очередь бросаются в глаза при визуальном анализе.

2. Многосортные матрицы совместной встречаемости как универсальная основа систем поиска изображений

Поскольку «понимание» изображений, как основная проблема машинного зрения все еще весьма далека от своего решения, наиболее реалистичной стратегией решения практических задач поиска изображений остается стратегия прямого поиска изображений по их содержанию, рассматриваемому на уровне растровых, а не семантических представлений. При этом отправной точкой является выбор подходящих дескрипторов, обеспечивающих компактное представление характерных признаков изображений и уникально характеризующих каждое изображение среди всех остальных.

В данной работе рассматривается универсальный подход к организации систем поиска медицинских изображений по образцу, основанный на расширенных матрицах совместной встречаемости различных видов. Данный тип дескрипторов является достаточно гибким и аккумулирует все основные признаки изображений - текстуру, цвет и форму. Поэтому он одинаково хорошо подходит для описания изображений самых различных типов, начиная от плоских силуэтов объектов (двумерная форма), представленных в растровом виде замкнутыми контурами пикселов, до трехмерных томографических изображений головного мозга, хранящихся в базах данных современных неврологических клиник и исследовательских центров. Поиск изображений по образцу включает в себя две следующие основные стадии.

1. Включение изображений в базу данных и генерация их дескрипторов, представляющих собой список ненулевых элементов матриц совместной встречаемости подходящего типа.

2. Собственно поиск, включающий задание изображения-образца, генерацию его дескриптора либо загрузку готового дескриптора из базы данных, сравнение дескриптора образца с дескрипторами, хранящимися в БД, выбор первых N (обычно 10-20) изображений, наиболее близких к образцу и выдача их пользователю. На первой стадии предварительно накопленные изображения обрабатываются и для каждого из них генерируется дескриптор, представляющий в сжатой форме его содержание. После этого он запоминается в базе данных и устанавливается перекрестная ссылка на исходное изображение, используемая в дальнейшем в целях визуализации изображений, найденных по их дескрипторам.

   На стадии поиска, дескриптор заданного изображения-образца сравнивается с дескриптором каждого изображения БД, вычисляется соответствующая мера близости (как правило, это $L1$ расстояние или Евклидова метрика) и пользователю предъявляются изображения, наиболее похожие на заданный образец. При этом дескриптор изображения-образца либо генерируется для каждого нового изображения, либо просто извлекается из БД в случае, когда образцом является одно из существующих изображений БД. Легко представить, как данная последовательность действий может быть использована циклически для обеспечения итеративных схем поиска, когда одно из найденных изображений

вновь подается пользователем на вход системы в виде образца и так далее. Во всех случаях необходимо помнить, что основной задачей систем поиска медицинских изображений как правило является лишь предварительный отбор изображений, близких к заданному пользователем образцу. Окончательное же решение о их релевантности решаемой задаче принимается самим пользователем на основе глубоких профессиональных знаний, личного опыта и зачастую с использованием неких неформализуемых критериев.

3. Поиск изображений объектов по 2D форме

Предметом данного параграфа является рассмотрение задачи поиска изображений объектов по их двумерной (планарной) форме, т.е. по изображениям их силуэтов. Среди наиболее известных методов, используемых для анализа и распознавания формы, следует отметить метод частотного анализа, базирующийся на сферических гармониках, метод моментов, метод прокрустова анализа, морфологические методы, а так же методы, основанные гистограммных признаках формы. К последней группе методов в частности относится рассматриваемый ниже подход к поиску объектов по их форме, основанный на матрицах совместной встречаемости [1]. Целью данного параграфа является адаптация общего подхода к описанию структуры изображений на основе расширенных матриц совместной встречаемости и его применение к задаче поиска изображений биологических объектов в базах данных по их форме.

Предполагается, что форма рассматриваемого объекта изображения представлена простым контуром, т.е. упорядоченной цепочкой пикселов на плоскости, которая не касается и не пересекает саму себя. Следует подчеркнуть, что цепочка пикселов является направленной. Данный факт имеет значение, поскольку при определении углов (см. ниже), учет свойства направленности приводит к появлению положительных и отрицательных величин. Для описания формы замкнутого контура будем использовать элементарные структуры, образуемые всевозможными парами его пикселов. Очевидно, что любая точка контура однозначно определяется ее индексом, т.е. порядковым номером в цепочке, имеющей начальную и конечную точки. В соответствии с общей методикой, введем следующие параметры, описывающие произвольную пару пикселов контура (i, k).

$h(i, k)$ – кратчайшее расстояние между точками контура, т.е. расстояние по хорде.

$c(i, k)$ – кратчайшее расстояние между точками вдоль контура.

Очевидно, что $c(i, k) \geq h(i, k)$ и равенство достигается только для отрезка контура, являющегося прямой линией. В случае, когда при решении задачи необходимо обеспечить независимость от абсолютных размеров сравниваемых объектов, расстояние $c(i, k)$ нормируется на его максимальное значение. Максимальным расстоянием в контуре является половина его длины, поскольку всегда рассматривается кратчайший путь между любой парой точек.

$a(i)$, $a(k)$ – углы наклона касательных к контуру в точках i и k соответственно. Для некоторой точки j дискретного контура n с координатами $x(n[j])$ и $y(n[j])$ на плоскости угол касательной определяется как угол прямой, проходящей через две

точки контура, находящихся на расстоянии R до и после текущей, т.е. как:

$$a(j) = \arctan\left[\frac{y(n[j+R]) - y(n[j-R])}{x(n[j+R]) - x(n[j-R])}\right],$$

где R является управляющим параметром, который называется радиусом касательной и фактически задает отступ от рассматриваемой точки вдоль контура в обе стороны. Радиус касательной выбирается в соответствии с характерной величиной локальной кривизны, т.е. сообразно со степенью извилистости анализируемых контуров и обычно изменяется в пределах от 2 до 5 пикселов контура.

$A(i,k)$ – угол между касательными в точках i и k, опирающийся на хорду и изменяющийся в пределах от 0 (пара точек на отрезке прямой) до 180 градусов (пара диаметрально противоположных точек контура).

$t(i,k) = c(i,k)/h(i,k)$ – отношение расстояния по контуру к расстоянию по хорде для рассматриваемой пары пикселов (i,k). Данный параметр измеряет величину отклонения участка контура от прямой и в случае, когда абсолютные размеры сравниваемых контуров не важны, может использоваться в матрицах совместной встречаемости вместо параметра $c(i,k)$, либо вместо обоих параметров, через которые он определен.

Компактной формой частотного представления введенных выше параметров контура является трехмерная матрица совместной встречаемости вида:

$$W_S = \|w_S(c(i,k), h(i,k), A(i,k))\|,$$

где $w_S(c,h,A)$ есть количество пар точек контура, относительное расстояние между которыми по контуру равно c, по хорде – h, а угол между касательными в этих точках равен A. На практике, для сокращения размерности матриц и уменьшения количества пустых (нулевых) элементов, наличие которых обусловлено дискретностью контура, все перечисленные выше параметры пар пикселов квантуются. Общие рекомендации по выбору размеров интервалов квантования даны в [1]. Конкретные значения обычно выбираются экспериментально в соответствии с целями решаемой задачи и особенностям формы анализируемых объектов (степень гладкости или извилистости, различия контуров в максимальных размерах и их важность для решаемой задачи, пространственное разрешение при оцифровке исходного физического объекта и т.п.).

Для формального определения описанных выше матриц совместной встречаемости, обозначим номера интервалов расстояния между точками по контуру через $b_C = 1, ..., B_C$, интервалы расстояния по хорде – через $b_H = 1, ..., B_H$, интервалы углов между касательными – через $b_A = 1, ..., B_A$, а интервалы величины отношения $t(i,k)$ – через $b_T = 1, ..., B_T$. Тогда элементы содержательно описанной выше трехмерной матрицы совместной встречаемости W_S, обозначаемой в тексте по первым буквам ее параметров через СНА, могут быть формально определены как:

$$\textbf{СНА}: \; w_S(b_C, b_H, b_A) = card\{i,k \in R^2 \mid \; i \neq k, c(i,k) \in b_C, h(i,k) \in b_H, A(i,k) \in b_A\},$$

где $card$, как обычно означает количество, а выражения типа $c(i,k) \in b_C$ означают, что величина соответствующего параметра находится в пределах интервала с

указанным номером. В ситуациях, когда при анализе формы объектов необходимо обеспечить как можно большую независимость от линейных размеров, более предпочтительным оказывается использование матриц типа CTA с элементами следующего вида:

$$\mathbf{CTA}: \; w_S(b_C, b_T, b_A) = card\{i, k \in R^2 \mid \; i \neq k, c(i,k) \in b_C, t(i,k) \in b_T, A(i,k) \in b_A\}.$$

Наконец, при проведении анализа может оказаться, что помимо глобальной формы, важную роль играют также особенности локального строения сравниваемых объектов, т.е. форма сравнительно небольших «извилин», петель и углов. В данной ситуации более эффективным может оказаться использование двумерных матриц совместной встречаемости типа CA, являющихся редуцированной версией матриц CHA с элементами вида:

$$\mathbf{CA}: \; w_S(b_C, b_A) = card\{i, k \in R^2 \mid \; i \neq k, c(i,k) \in b_C, A(i,k) \in b_A\}.$$

Предложенный подход исследовался экспериментально на примере БД морских животных [4], включающей 958 изображений. Длина контуров объектов варьировала от 256 до 1653 пикселов. Размеры охватывающего прямоугольника при этом изменялись от 16 до 380 пикселов по оси X и от 23 до 526 пикселов по оси Y. Основными причинами выбора для тестирования именно этой БД являлось разнообразие, а порой и чрезвычайная замысловатость формы представленных в ней природных объектов, а так же наличие достаточно большого количества интуитивно понятных и визуально дифференцируемых классов. Предварительные эксперименты, проведенные на ограниченной подвыборке из 116 контуров, показали, что качество поиска изображений при использовании каждого из трех типов матриц, описанных выше, достаточно близко. Поскольку вычислительные затраты и размер получаемых дескрипторов были существенно ниже в случае использования матриц типа CA, то основная часть экспериментов на полной БД проводилась с использованием матриц данного типа.

Размеры матриц CA принимались равными 15 интервалам по 1/15 половины длины контура каждый для расстояния $c(i,k)$ и 10 интервалам по 18 градусов каждый для угла $A(i,k)$. Таким образом, полный размер дескрипторов составлял $15 \times 10 \times 4 = 600$ байт. Матрицы нормировались построчно, т.е. по каждому расстоянию отдельно. При осуществлении поиска матрица образца и матрицы (дескрипторы) изображений БД сравнивались поэлементно. В качестве расстояния в пространстве признаков использовалась $L1$ норма. Детальное тестирование данного способа показало, что результаты поиска хорошо согласуются с экспертной оценкой близости найденных изображений к заданному образцу в подавляющем большинстве случаев. Примеры запросов-образцов и соответствующих результатов поиска в БД морских животных приведены на рис. 1.

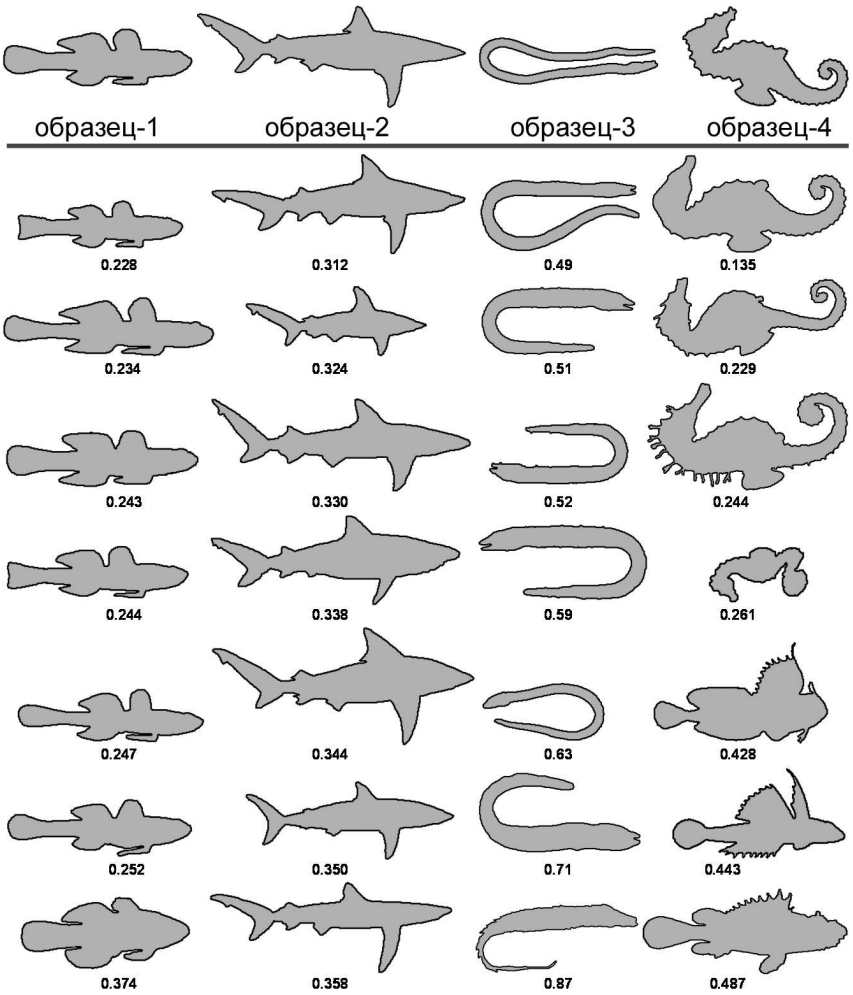


Рис. 1. Изображения-образцы (верхний ряд) и результаты поиска в базе данных морских животных, ранжированные по степени близости к образцу (под ними).

Проведенные экспериментальные исследования предложенного метода поиска изображений объектов по их 2D форме на основе специализированных матриц совместной встречаемости позволяют сделать вывод, что предлагаемый подход обеспечивает высокое качество поиска биологических объектов, имеющих нетривиальную форму. Вопреки достаточно распространенному мнению о необходимости вычисления признаков матриц совместной встречаемости как способа избавиться от их «громоздкости» и сокращения размеров памяти, требуемой для их хранения, проведенные экспериментальные оценки показали высокую компактность используемых дескрипторов. Так, например, при размере БД в 10 тысяч изображений, потребуется всего лишь около 6 мегабайт памяти для хранения дескрипторов. Что касается времени поиска, то проведенные вычислительные эксперименты показали, что для

указанного размера БД в 10 тысяч изображений, общее время поиска по образцу составляет и не более 1-3 секунд на рядовом настольном компьютере.

4. Поиск цветных двумерных изображений

Следуя общей методике конструирования матриц совместной встречаемости [5, 6], а также учитывая необходимость квантизации большого количества цветов, трехмерные матрицы совместной встречаемости цветов типа цвет-цвет-расстояние CCD могут быть определены как:

$$\textbf{CCD}: \; W_C = ||w_C(V(i), V(k), d(i,k))||,$$

$$w_C(b_{Vi}, b_{Vk}, b_d) = card\{i,k \in R^2| \;\; i \neq k, V(i) \in b_{Vi}, V(k) \in b_{Vk}, b_d = round(d(i,k)),$$

$$x_k = (x_i + \Delta x), y_k = (y_i + \Delta y), -D \leq \Delta x \leq D, -D \leq \Delta y \leq D, \Delta y S + \Delta x > 0, S = 2D+1\}.$$

В приведенном определении через (i,k) обозначена произвольная пара пикселов, расположенная на расстоянии $d(i,k)$ в плоскости изображения, а $V(i)$ и $V(k)$ соответствуют цветам этих пикселов. Весь диапазон RGB цветов квантуется путем равномерного разбиения на $b_V = 1, ..., B_C$ интервалов, а межпиксельное расстояние изменяется в пределах от 1 до D пикселов. Следует отметить, что в отличии от большинства других типов матриц, представленных в данной работе, матрицы совместной встречаемости цветов типа CCD имеют несколько другую историю. Они были предложены независимо в 1997 году группой американских исследователей под руководством профессора Хуанга [7] и автором данной работы во время его работы с доктором Волмером в Фраугоферовском институте компьютерной графики в Дармштадте [8]. Однако, поскольку публикация Хуанга и коллег вышла в том же 1997 году, а работа [8] – только в следующем, то приоритет принадлежит американским коллегам и именно их работа цитируется в большом количестве последующих публикаций в области поиска цветных изображений общего характера. В качестве некоторой «компенсации» авторам может рассматриваться лишь то, что работа [8] вошла в число трех лучших работ первой Международной конференции по системам мультимедиа в Лозанне.

Целью экспериментов, описанных ниже являлась оценка качества поиска 2D слоев цветокодированных SPECT изображений головного мозга, а так же оценка принципиальной возможности использования технологии поиска изображений в качестве одного из инструментов диагностики заболеваний на примере болезни Алзгеймера.

В качестве исходных данных использовались 79 функциональных SPECT изображений головного мозга, включающих 45 изображений пациентов с диагнозом болезни Алзгеймера на различных стадиях развития заболевания и 34 контрольных изображения здоровых добровольцев. Значительная часть указанных исходных материалов представляла собой те же самые SPECT изображения, которые использовались для оценки методов детектирования нейро-дегенеративных изменений головного мозга, описанных в [9]. Соответственно, за более детальной информацией касающейся пациентов, используемого томографического оборудования, а так же содержания процедур подготовки SPECT изображений к анализу следует обратиться к работе [9].

Для создания БД трехмерные изображения были вначале сконвертированы в цветокодированное представление с использованием одной из стандартных нейрологических цветовых палитр, а затем экспортированы в виде набора цветных двумерных изображений, каждое из которых представляло отдельный аксиальный слой с помощью известного пакета программ MRIcro. В результате данного преобразования была получена БД, состоящая из 1580 изображений, включая 900 изображений мозга пациентов, страдающих болезнью Алзгеймера и 680 контрольных изображений.

Тестирование проводилось путем последовательной автоматической выборки из БД и использования каждого из 1580 изображений в качестве изображения-образца, подаваемого на вход системы. Как и в предыдущих примерах, качество поиска формально оценивалось путем подсчета количества изображений, принадлежащих тому же классу, что и изображение-образец. Типичные примеры результатов поиска по изображениям-образцам, представляющим норму и болезнь Алзгеймера показаны на рис. 2. В приведенных примерах среди двадцати трех изображений, наиболее близких к образцу нормы, два принадлежат к противоположному классу, т.е. болезни Алзгеймера (предпоследнее изображение во втором ряду и третье изображение в последнем ряду на верхней панели рис. 2). При поиске по образцу нормы (нижняя панель) ошибочное отнесение к противоположному классу отсутствует.

В целом, анализ результатов поиска по всем 1580 запросам показал, что среди найденных изображений 91.3% принадлежат к правильному классу, т.е. тому же, что и изображение-образец. Не вызывает сомнения, что указанный процент корректного отнесения представляет определенную информацию о качестве процедуры поиска. Однако при интерпретации результатов не следует также забывать, что основным функциональным назначением средств поиска в данном случае является поиск изображений, имеющих похожий SPECT паттерн (близкую пропорцию активных и пассивных участков мозга), а не задача классификации и распознавания как таковая. Действительно, наличие, например, в изображениях болезни Алзгеймера участков (здесь – слоев), похожих на некоторые участки изображений нормы и обратная ситуация совсем не обязательно является чем-то недопустимым или ошибочным.

5. Поиск полутоновых 2D изображений (на примере шизофрении)

В качестве основы для проведения исследования использовалось 40 трехмерных изображений головного мозга испытуемых, включающих группу из 19 пациентов, больных шизофренией и группу из 21 контрольных изображений мозга здоровых добровольцев. Все исходные изображения были получены на томографе GE Signa (General Electric Medical Systems, Milwaukee) с напряженностью магнитного поля 1.5 Тесла в одной из психиатрических клиник Великобритании. При проведении магнитно-резонансного сканирования в качестве управляющих параметров последовательности использовались следующие значения: TR=4000 мс, TE=20.85 мс. Исходные изображения имели пространственное разрешение внутри слоев 0.856×0.856 мм при толщине слоя 3 мм. Аксиальные слои были ориентированы параллельно плоскости передней и задней комиссур.

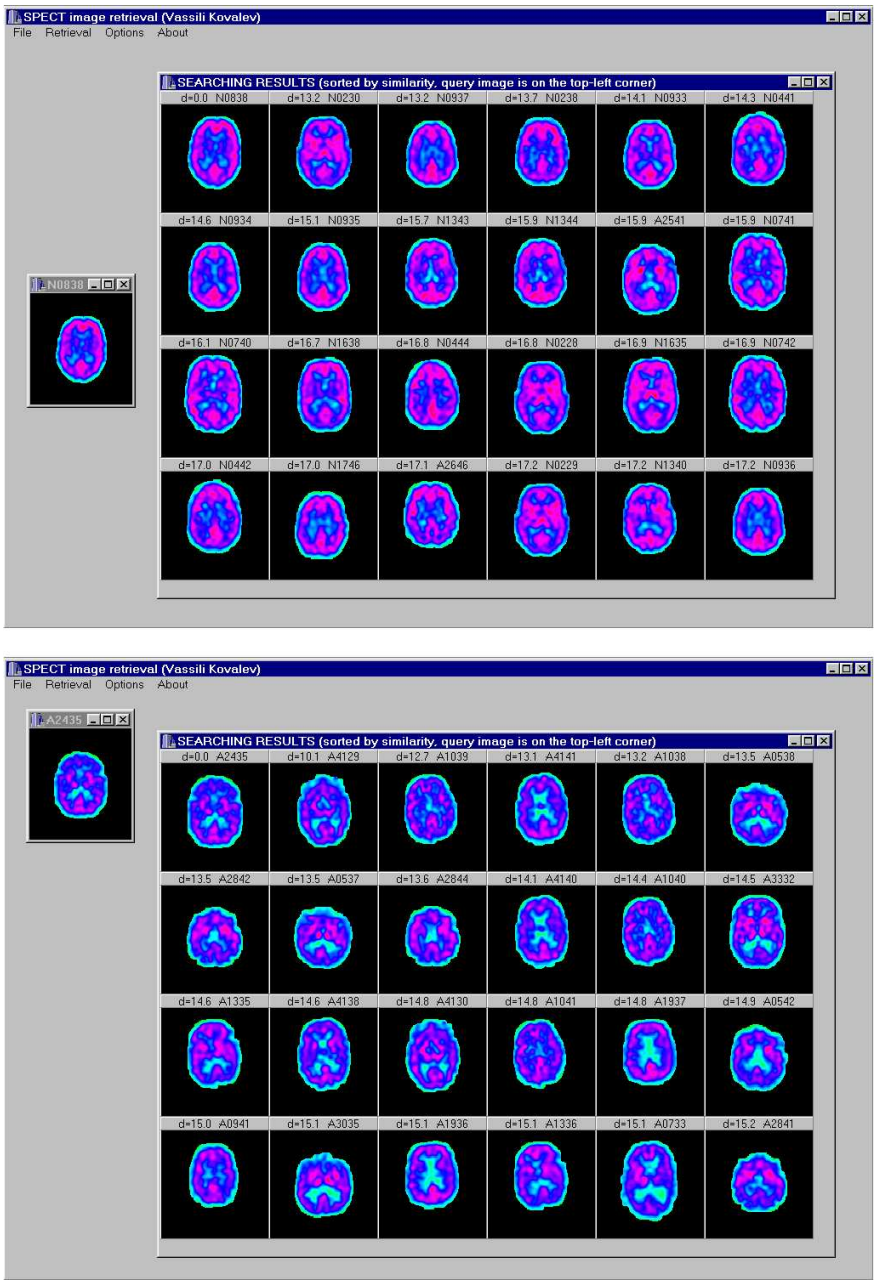


Рис. 2. Примеры результатов поиска цветокодированных SPECT изображений головного мозга по заданному изображению-образцу здорового добровольца (верхняя панель) и пациента, страдающего болезнью Алзгеймера (нижняя панель).

Для подготовки тестовой БД, описанные выше трехмерные изображения мозга сорока испытуемых были экспортированы в качестве отдельных файлов, каждый из которых представлял собой один аксиальный слой. Учитывая, что каждое исходное

изображение состояло из 32 слоев, общее число изображений тестовой БД насчитывало $32 \times 40 = 1280$, в том числе 608 изображений больных шизофренией и 672 контрольных изображения. Следует подчеркнуть, что исследуемые группы изображений имеют очень незначительные визуальные отличия (рис. 3). Поэтому задача поиска похожих изображений в случае трактовки понятия «похожести» как принадлежности к одному и тому же классу, является чрезвычайно сложной.

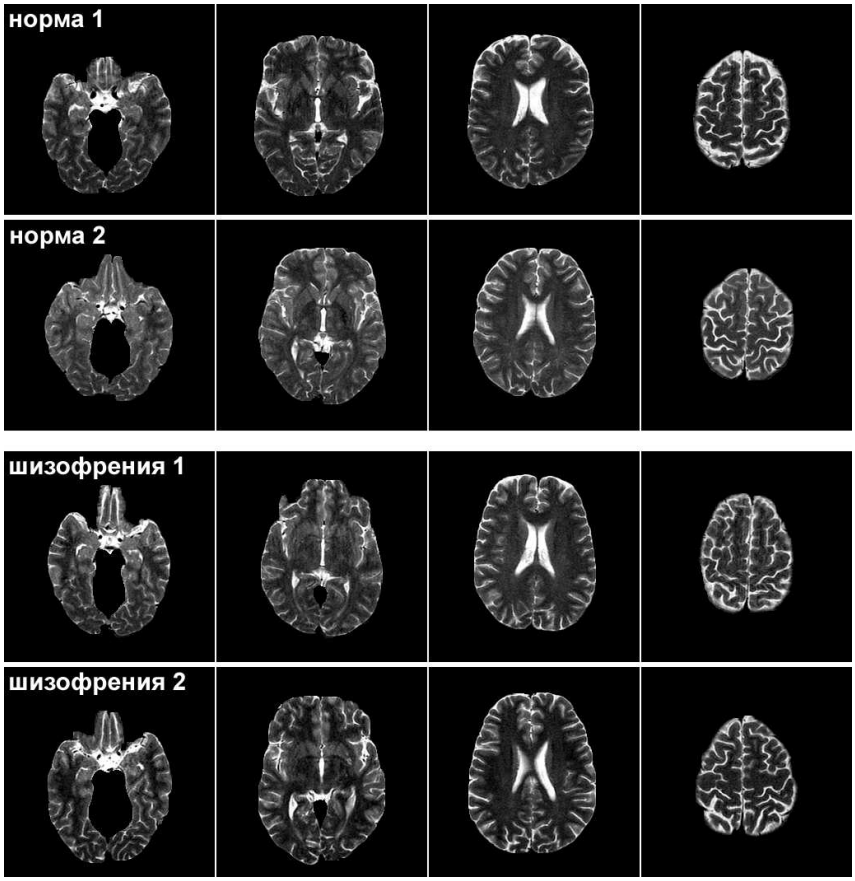


Рис. 3. Примеры аксиальных слоев МРТ изображений мозга здоровых добровольцев (два верхних ряда) и больных шизофренией (два нижних ряда).

В соответствии с результатами сравнительного исследования текстурных свойств мозга в норме и при шизофрении, описанными в [10], наибольшая степень дифференциации классов достигается не в случае описания изображений с помощью матриц общего вида типа IIGGAD (интенсивность, градиент, угол, расстояние, см. [1]), а при использовании их редуцированных версий. В частности, хорошо проявили себя дескрипторы GGAD, являющиеся матрицами совместной встречаемости градиентов, которые отличаются от матриц общего вида тем, что измерения, соответствующие интенсивностям вокселов, исключены. Это объясняется тем, что при распознавании

данного типа изображений более важной оказалась не непосредственно яркость вокселов как таковая, а скорость ее пространственного изменения, т.е. градиент. При выполнении тестирования оценивалось качество поиска для каждого из изображений БД, подаваемого на вход системы в качестве образца. Как и ранее, качество оценивалось путем подсчета процента найденных изображений, принадлежащих «правильному» классу. На рисунке 4 приведен пример изображения-образца, представляющего один из аксиальных слоев изображения мозга пациента, больного шизофренией, а так же 15 ближайших к нему изображения, найденных в БД общим объемом 1280 изображений. В условиях данного примера оказалось, что все 15 изображений-результатов также принадлежат различным больным, страдающим шизофренией.

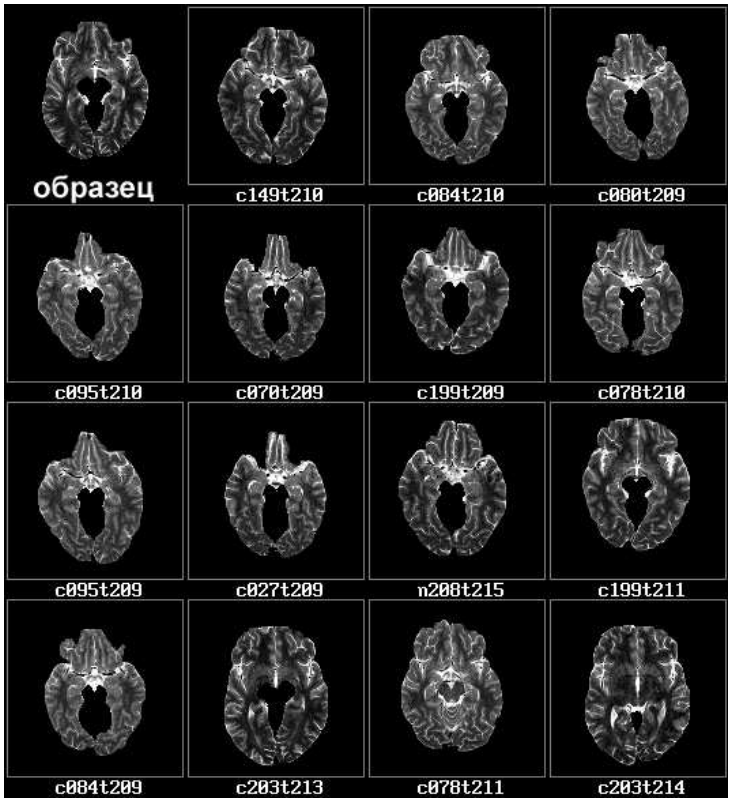


Рис. 4. Пример поиска изображений в базе данных двумерных МРТ изображений мозга в норме и при шизофрении.

Статистический анализ результатов 1280 запросов показал, что среди 15 ближайших результатов поиска, процент изображений, принадлежащих к ожидаемому классу, составляет 82.6%. Надо отметить, что такая точность является достаточно высоким показателем для столь трудно различимых МРТ изображений как норма и шизофрения. Данный результат согласуется с точностью классификации для различных частей мозга, приведенной в [10].

6. Поиск полутоновых 3D изображений мозга по возрасту и по полу

Ранее нами было показано [1], что возрастные изменения мозга проявляются анатомически весьма незначительно для широкого диапазона возрастов, вплоть до вступления в период преклонного возраста порядка 80 лет и более. Поэтому проверка средств поиска медицинских изображений на базе данных МРТ изображений головного мозга здоровых людей молодого и раннего пожилого возраста могла бы служить очень хорошим тестом чувствительности разработанного метода.

Для проведения подобного тестирования было отобрано 55 трехмерных анатомических T_1-взвешенных МРТ изображений мозга высокого разрешения, полученных путем сканирования здоровых добровольцев в возрастном диапазоне от 16 до 70 лет. Тестовая выборка была разбита на две группы. В первую группу, условно называемую молодыми людьми, входили изображения 33 испытуемых в возрасте от 16 до 25 лет, 17 из которых были мужчины, а остальные 16 – женщины. Во вторую группу, условно называемую людьми зрелого возраста, входило 22 изображения мозга испытуемых в возрасте от 50 до 70 лет, в том числе 11 мужчин и 11 женщин. В качестве дескрипторов изображений использовались обобщенные шестимерные матрицы типа IIGGAD со стандартными значениями управляющих параметров квантизации всех входящих в них величин [1]. Поиск изображений осуществлялся по обычной схеме с последовательной подачей на вход системы в виде изображения-образца каждого из 55 изображений БД. После выполнения автоматического поиска проводился анализ N наиболее близких к образцу изображений. Анализ проводился раздельно для $N = 20$ (двадцать ближайших изображений), $N = 15$, $N = 10$, $N = 5$ и $N = 1$ (изображение БД, самое близкое к заданному образцу). При этом подсчитывалось количество правильно найденных изображений, т.е. изображений, принадлежащих той же самой возрастной группе, что и заданный образец.

На рис. 5 показаны слои трехмерных МРТ изображений, иллюстрирующие достаточно типичные образцы поиска и $N = 4$ наиболее близких к ним изображений, найденных в БД. В данном примере все изображения, полученные в результате поиска, принадлежат к тем же возрастным группам, что и образцы, т.е. в условиях данного теста являются корректными. Статистические данные по результатам выполнения всех 55 запросов для каждого из рассмотренных выше значений N наиболее близких изображений представлены в первой строке таблицы 1.

Таблица 1. Точность поиска изображений мозга по возрасту и по полу

% корректных среди N ближайших	$N = 1$	$N = 5$	$N = 10$	$N = 15$	$N = 20$
молодой/зрелый возраст	96.4	93.8	90.6	86.3	82.2
мужчины/женщины	67.6	65.1	61.9	61.1	60.5

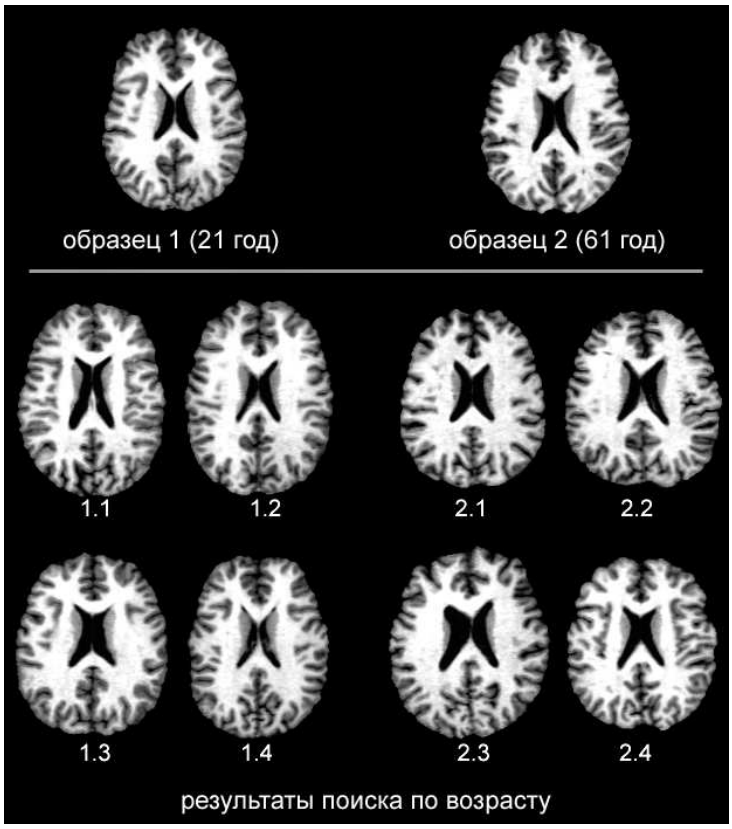


Рис. 5. Примеры изображений-образцов мозга здоровых добровольцев, входящих в группу молодых людей (слева) и людей зрелого возраста (справа) и четыре наиболее близких изображения-результата поиска, принадлежащие испытуемым из тех же возрастных групп (под ними).

Кроме поиска по возрасту, были проведены аналогичные эксперименты по поискау изображений мозга мужчин и женщин с использованием 210 молодых добровольцев (103 мужчины и 107 женщин), количественные результаты которых обобщены во второй строке таблицы. Как видно из таблицы, общее качество поиска по возрасту является достаточно высоким и достигает 96.4% корректных случаев при рассмотрении $N = 20$ ближайших к образцу изображений. Учитывая незначительность различий классов изображений (ср. левую и правую половины рис. 5) и довольно высокую точность поиска (первая строка таблицы), можно сделать вывод о высокой чувствительности и надежности рассматриваемого метода. Следует так же подчеркнуть, что задача поиска изображений по полу является экстремально сложной, поскольку изображения мозга мужчин и женщин визуально практически неразличимы. Как следует из данных, приведенных в таблице, количество изображений корректного класса заметно превышает вероятность случайного угадывания и изменяется в пределах от 60.5% для единственного изображения до 67.6% для двадцати наиболее похожих изображений. Следует заметить, что приведенные статистические

данные являются достаточно надежным основанием для оценки качества поиска, поскольку они базируются на результатах обработки 210 запросов к базе данных. Соответственно, для $N = 1$ среднее значение было подсчитано по 210 изображениям, в то время как для $N = 20$ общее количество оцененных изображений в 210 сериях результатов поиска составляло $210 \times 20 = 4200$.

Выводы

В данной работе был рассмотрен новый подход к организации систем поиска биомедицинских изображений по образцу, основанный на расширенных матрицах совместной встречаемости различных видов. По результатам проведенных исследований и экспериментов можно сделать вывод, что предложенный метод описания содержания биомедицинских изображений является универсальным и высоко эффективным. Он одинаково хорошо подходит для описания изображений самых различных типов, начиная от плоских силуэтов объектов (двумерная форма) до трехмерных МРТ изображений головного мозга высокого разрешения.

Список литературы

1. *Ковалев В.А.* Анализ текстуры трехмерных медицинских изображений. Минск: Белорусская наука, 2008, ISBN 978-985-08-0905-6. – 264 с.
2. *Huang H.K.* PACS and imaging Informatics: Basic principles and applications. Wiley, Hoboken, New Jersey, USA, ISBN 0-471-25123-2, 2004. – 704 p.
3. *Osborne I.* The next information infrastructure: towards 2012. Roadmap version 1.0. – Grid Computing Now! Knowledge Transfer Network. UK Department of Trade and Industry, London, September 2007. – 26 p.
4. *Mokhtarian F., Abbasi F, Kittler J.* Robust and efficient shape indexing through curvature scale space. In: 6th British Machine Vision Conference (BMVC-96), Edinburgh, 10-12 September, British Machine Vision Association, 1996, pp. 53-62.
5. *Kovalev V., Petrou M.* Multidimensional co-occurrence matrices for object recognition and matching. Graphical Models and Image Processing, vol. 58. No 3, 1996, pp. 187-197.
6. *Kovalev, V.A., Kruggel, F., Gertz, H.-J. and von Cramon D.Y.* Three-dimensional texture analysis of MRI brain datasets. IEEE Transactions on Medical Imaging, vol. 20, No 5, 2001, pp. 424-433.
7. *Huang J., Kumar S.R., Mitra M., Zhu W.-J., Zabih R.* Image indexing using color correlograms. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, 1997, pp. 762-768.
8. *Kovalev V., Volmer S.* Color co-occurrence descriptors for querying-by-example. International Conference on Multimedia Modelling. Oct. 12-15. Lausanne, Switzerland, IEEE Computer Society Press, 1998, pp. 32-38.
9. *Kovalev V.A., Thurfjell L., Lundqvist R. and Pagani M.* Asymmetry of SPECT Perfusion Image Patterns as a Diagnostic Feature for Alzheimer's Disease. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI 2006), Copenhagen, Denmark, 1-6 October, Larsen R., Nielsen M. and Sporring J. (Eds), Springer Verlag, LNCS, vol. 4191, 2006, pp. 421-428.
10. *Kovalev V.A., Petrou M., Suckling J.* Detection of structural differences between the brains of schizophrenic patients and controls. Psychiatry Research: Neuroimaging, vol.124, 2003. pp. 177-189.

Статья поступила в редакцию 18.04.2008

УДК 004.932

ИТЕРАЦИОННЫЕ ПРОЦЕДУРЫ ОБРАБОТКИ ИЗОБРАЖЕНИЙ НА ОСНОВЕ АЦИКЛИЧЕСКОГО ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

© Копылов А.В., Мельников П.А.

Тульский государственный университет, Россия
факультет кибернетики
300600, Россия, г. Тула, пр. Ленина 92,

e-mail: *kopylov@uic.tula.ru, petrmelnikov@mail.ru*

Abstract. A lot of image analysis problems lend themselves to a unified mathematical formulation as optimization problems, namely (min, +)labelling problems. It is known that these problems are NP-hard in the general formulation, but when the adjacency graph is a tree, they can be effectively solved by the tree-serial dynamic programming procedure. In this paper, it is considered an optimization method on the basis of Gauss-Seidel principle, which uses iterative reevaluation of groups of variables with tree-like neighborhood relation. We change a way of such variable aggregation from iteration to iteration, as a mean for increasing the robustness of the algorithm with respect to local extremes.

Введение

Несмотря на большое разнообразие задач и множество существующих алгоритмов низкоуровневой обработки изображений, каждый из которых использует свой математический аппарат и особые эвристические приемы, оказывается возможным выделить достаточно большие подклассы задач, допускающие единую математическую постановку.

При компьютерном анализе исходное изображение, в большинстве случаев, представляется как числовой массив, $Y = (y_{\mathbf{t}},\ \mathbf{t} \in T)$, который обычно принимает значения на непрерывной или дискретной оси уровня яркости $y_{\mathbf{t}} \in \mathscr{Y}$ и определен на дискретном множестве элементов $T = \{\mathbf{t} = (t_1, t_2) :\ t_1 = 1, ..., N_1,\ t_2 = 1, ..., N_2\}$. Результатом анализа данных является вторичный массив $X = (x_t,\ \mathbf{t} \in T)$, определенный на том же множестве элементов $\mathbf{t} \in T$ и принимающий значения $x_t \in \mathscr{X}$ из множества $\mathscr{X}$, специфичного для каждой конкретной задачи.

В этом случае задачу обработки изображения можно понимать как задачу выбора наиболее «подходящей» оценки $\hat{X}$ вторичного массива X из множества всех его мыслимых значений $\mathbb{X}$. Очевидно, что всякое правило оценивания представляет собой в данном случае некоторый оператор $\hat{X}(Y)$ и с математической точки зрения проблема обработки изображения сводится к выбору этого оператора.

Выбор оценки всегда производится из соображений компромисса между двумя обычно противоречащими друг другу требованиями. С одной стороны, знание, по

[1]Исследование выполняется при поддержке грантов 08-01-99003, 06-07-89249, 06-01-00412, 08-01-00695 Российского фонда фундаментальных исследований

крайней мере, основных механизмов изучаемого явления, как правило, дает представление о том, какие значения результата обработки более естественны и, следовательно, более ожидаемы, чем другие. Соответствующую информацию, имеющуюся еще до того, как подлежащее анализу изображение стало доступно наблюдателю,будем называть априорной информацией. С другой стороны, это же знание позволяет количественно судить, какие значения вторичного массива данных лучше согласуются с анализируемым изображением. Соответствующую информацию, непосредственно основанную на анализе зарегистрированного изображения, будем называть информацией наблюдения.

При использовании оптимизационного подхода, вторичный массив должен играть роль аргумента целевой функции $J(X|Y)$, оценивающей несоответствие между каждой допустимой версией результата X и исходным массивом данных Y. Важно лишь так выбрать класс целевых функций, чтобы было гарантировано существование достаточно эффективного алгоритма поиска точки минимума

$$\hat{X}(Y) = \arg\min_{x_{\mathbf{t}} \in \mathscr{X}} J(X|Y), \tag{1}$$

выступающей в качестве результата обработки. Алгоритм минимизации целевых функций из выбранного класса и будет играть роль универсального обобщенного алгоритма обработки изображений.

Существенной особенностью изображения как массива данных является упорядоченность вдоль осей двух, либо даже трех переменных, если анализу подлежит изображение, изменяющееся во времени, как это, вообще говоря, и имеет место в зрительных системах человека и животных. Каждый элемент $\mathbf{t} \in T$ такого массива данных естественным образом связан с рядом соседей. Удобно выражать такое отношение соседства в виде неориентированного графа G, который понимается как множество пар соседних элементов $G \subset T \times T$.

В качестве универсального носителя всей используемой в дальнейшем информации наблюдения об исходном массиве данных Y в окрестности каждой его точки $\mathbf{t} \in T$ удобно использовать локальные оценочные функции $\psi_t(x_t) = \psi_t(x_t|Y)$, каждая из которых определена на множестве значений целевой переменной и принимает тем меньшие значения, чем более правдоподобной представляется предположение, что $x_{\mathbf{t}}$ как раз и есть искомое значение этой переменной. Мы будем называть эти функции узловыми функциями, поскольку каждая из них связывается с одним элементом массива данных, играющим роль одной из вершин (узлов) графа смежности элементов массива.

Для выражения априорных предположений о результате обработки кроме оценочных функций, зависящих от данных, используются также так называемые функции связи, определяемые модельными представлениями о предпочтительных сочетаниях значений пар соседних целевых переменных. Каждая функция связи задается как функция соответствующих двух переменных $\gamma_{\mathbf{t}',\mathbf{t}''}(x_{\mathbf{t}'}, x_{\mathbf{t}''})$, $(\mathbf{t}', \mathbf{t}'') \in G$, возрастающая при увеличении взаимного рассогласования их значений в том смысле, который диктуется имеющимися модельными представлениями о желаемом результате обработки.

Каждой вершине графа соседства G элементов изображения соответствует одна целевая переменная $x_{\mathbf{t}}$ и, следовательно, одна локальная оценочная функция $\psi_t(x_t) = \psi_t(x_t|Y)$, а совокупность его ребер отображает представление о непосредственных связях между переменными, которые и должны быть учтены в процессе согласования локальных решений. Именно с ребрами графа соседства $(\mathbf{t}', \mathbf{t}'') \in G$ и отождествляются функции связи $\gamma_{\mathbf{t}', \mathbf{t}''}(x_{\mathbf{t}'}, x_{\mathbf{t}''})$.

Таким образом, для заданного изображения Y решение о вторичном массиве X в целом следует искать минимизируя как узловых функции,так и функции связи. Естественно принять их сумму

$$J(X|Y) = \sum_{\mathbf{t} \in T} \psi_{\mathbf{t}}(x_{\mathbf{t}}) + \sum_{(\mathbf{t}', \mathbf{t}'') \in G} \gamma_{\mathbf{t}', \mathbf{t}''}(x_{\mathbf{t}'}, x_{\mathbf{t}''}) \qquad (2)$$

в качестве комбинированного критерия принятия решения (1). Заметим, что целевая функция (2), подлежащая минимизации, представляет собой сумму некоторого числа элементарных функций, каждая из которых зависит не более, чем от двух переменных, причем структура смежности переменных задается графом соседства элементов массива. Мы будем называть такие функции парно-сепарабельными с графом смежности переменных G. Число переменных такой целевой функции равно числу вершин графа, т.е. числу элементов массива $|T|$.

Известные алгоритмические средства решения подобных задач общего вида исчерпываются процедурами итерационного случайного поиска, такими как стохастическая релаксация и моделируемый отжиг (Simulated Annealing) [1]. Итерационные процедуры такого рода сходятся особенно медленно, поскольку большое число случайных шагов является как раз тем вычислительным приемом, который не позволяет процессу поиска «залипать» в локальных экстремумах. Это значительно ограничивает класс практических задач, для решения которых такие алгоритмы могут быть использованы.

В случае, когда целевые переменные принимают значения из конечного множества $x \in X = \{1, ..., m\}$, данная оптимизационная задача является одним из частных случаев так называемых задач разметки, который известен как (max, $+$) или (min,$+$) задачи. Для произвольного графа соседства G задача оптимизации парно-сепарабельной целевой функции (2) является NP-полной. Тем не менее, для некоторых частных случаев возможно построение достаточно эффективных итерационных алгоритмов оптимизации, имеющих полиномиальную вычислительную сложность [2]. Это так называемые субмодулярные задачи, которые можно свести к задачам поиска оптимального сечения графа [3], задачи в которых граф соседства не имеет циклов, то есть является деревом, и которые решаются на основе принципа динамического программирования [4]. Третий класс задач, включающий и первые два, объединяет задачи, разрешимые с помощью их эквивалентных преобразований [5]. Алгоритм решения задач третьего типа сводится к решению специфической задачи линейного программирования.

Вычислительная эффективность процедуры динамического программирования делает особенно интересными задачи второго типа, и алгоритмы, использующие для оптимизации ациклические графы, такие как Loopy Belief Propagation (LBP) [6] и

Tree-Reweighed Message Passing (TRW) [7] занимают в настоящее время лидирующие позиции с точки зрения точности получаемого решения и скорости работы, среди алгоритмов оптимизации парно-сепарабельных целевых функций вида (2). Однако скорость работы даже этих алгоритмов далека от реального времени, что делает проблематичным их использование, например, для задач построения функций сходства изображений в системах распознавания образов.

В данной статье рассматривается способ использования ациклического динамического программирования для создания быстрых итерационных алгоритмов оптимизации на основе группировки переменных так, что соответствующий каждой группе подграф графа соседства является деревом. Существенной особенностью метода является изменения способа группировки переменных в процессе итерационного пересчета.

1. Итерационная парно-сепарабельная оптимизация на основе принципа Гаусса-Зайделя

К сожалению, поиски алгоритма оптимизации для парно-сепарабельной целевой функции с произвольным графом смежности переменных неизбежно приводят к вычислительным трудностям. Природа поставила некий фундаментальный барьер надеждам построить неитерационную процедуру, которая позволяла бы находить глобальный минимум парно-сепарабельной целевой функции с графом смежности переменных произвольного вида.

Тем не менее, парная-сепарабельность целевой функции (2), дает возможность поочередной минимизации по каждой целевой переменной на основе принципа Гаусса-Зайделя. Общий принцип, подобных процедур заключается в поочередном итерационном исправлении значения оценки вторичного массива данных в каждой точке растра $x_{\mathbf{t}}$, начиная с некоторого исходного приближения. Каждое исправление строится, как правило, так, чтобы значение целевой функции, по крайней мере, не уменьшалось.

Пусть $X^k = (x^k_{\mathbf{t}},\ \mathbf{t} \in T)$ – очередное приближение к точке максимума. Выберем некоторый элемент массива $x_{\mathbf{t}^*}$. Тогда целевая функция (2) может быть представлена в следующем виде:

$$J(X|Y) = \psi_{\mathbf{t}^*}(x_{\mathbf{t}^*}) + \sum_{(\mathbf{t}^*,\mathbf{t}')\in G} \gamma_{\mathbf{t}^*,\mathbf{t}'}(x_{\mathbf{t}^*}, x_{\mathbf{t}'}) + \sum_{\mathbf{t}\in T;\,\mathbf{t}\neq\mathbf{t}^*} \psi_{\mathbf{t}}(x_{\mathbf{t}}) + \sum_{(\mathbf{t}',\mathbf{t}'')\in G;\,\mathbf{t}',\mathbf{t}''\neq\mathbf{t}^*} \gamma_{\mathbf{t}',\mathbf{t}''}(x_{\mathbf{t}'}, x_{\mathbf{t}''})$$

Решим экстремальную задачу:

$$x^{k+1}_{\mathbf{t}} = \arg\min_{x_{\mathbf{t}^*}\in\mathscr{X}} J(X|Y) = \arg\min_{x_{\mathbf{t}^*}\in\mathscr{X}} \left[\psi_{\mathbf{t}^*}(x_{\mathbf{t}^*}) + \sum_{(\mathbf{t}^*,\mathbf{t}')\in G} \gamma_{\mathbf{t}^*,\mathbf{t}'}(x_{\mathbf{t}^*}, x_{\mathbf{t}'})\right] \tag{3}$$

Из способа выбора значения $x^{k+1}_{\mathbf{t}^*}$ следует, что $J(x^{k+1}_{\mathbf{t}^*}, X^k_{(\mathbf{t}^*)}|Y) \le J(x^k_{\mathbf{t}^*}, X^k_{(\mathbf{t}^*)}|Y)$, где $X_{(\mathbf{t}^*)} = (x_{\mathbf{t}},\ \mathbf{t} \in T, \mathbf{t} \neq \mathbf{t}^*)$. Повторим операцию, аналогичную (3), последовательно для всех элементов массива, всякий раз заменяя значение переменной $x^k_{\mathbf{t}}$ найденным значением $x^{k+1}_{\mathbf{t}}$. Будем рассматривать совокупность новых значений

$X^{k+1} = (x_{\mathbf{t}}^{k+1},\ \mathbf{t} \in T)$ как очередную итерацию процесса минимизации. Очевидно, что

$$J(X^{k+1}|Y) \leq J(X^k|Y). \tag{4}$$

Процесс останавливается, когда на очередной итерации значения всех элементов поля останутся прежними $X^{k+1} = X^k$. Та же идея выраженная в вероятностных терминах, когда задача сводится к поиску максимума апостериорной плотности вероятности марковского случайного поля, приводит к алгоритму, получившему название Iterated Conditional Modes (ICM) [8].

Надо иметь в виду, что эта процедура гарантирует нахождение глобального минимума только в условиях весьма обременительных и трудно проверяемых предположений о целевой функции. Выполнение равенства $X^{k+1} = X^k$ еще не является гарантией его достижения. Таких точек, как правило, очень много, они являются точками лишь локального минимума, что категорически неприемлемо для подавляющего большинства прикладных задач анализа изображений. Алгоритмы типа ICM быстро сходятся, обычно за пять-шесть просмотров всего массива, однако платой за быструю сходимость является их склонность «залипать» в ближайшем локальном минимуме, так что результат оказывается очень сильно зависящим от исходного приближения, часто лишь немного «подправляя» его.

Ниже, в разделе 2, мы рассмотрим некоторые модификации данного метода, основанные на сочетании итерационного поиска по Гауссу-Зайделю с разбиением массива с произвольным графом смежности на два или более подмассивов с древовидной смежностью элементов. Такой прием позволит в существенной степени преодолеть проблему локальности итерационной процедуры.

2. Древовидная декомпозиция графа соседства в виде решетки

Поскольку граф соседства в виде решетки, являющийся естественным для изображений, деревом не является, то непосредственно применить процедуру динамического программирования для задач обработки изображений невозможно. Тем не менее, возможно организовать процедуру итерационной оптимизации с поочередным пересчетом значений целых групп соседних переменных связанных древовидным отношением соседства.

Основная идея такого итерационного метода состоит в разбиении исходного графа соседства G в виде решетки на два или большее число поддеревьев $G^1,\ G^2, \ldots,\ G^K$ так, что $G = \bigcup\limits_{j=1}^{K} G^j,\ G_i \cap G_j = \emptyset,\ i \neq j.$

Тогда, можно провести оптимизацию на одном из поддеревьев G_j, зафиксировав значения целевых переменных в остальных поддеревьях $G^i;\ i = 1, \ldots, K,\ i \neq j$, и получить новые значения переменных в этом дереве. Данный процесс повторяется для каждого поддерева $G^i;\ i = 1, \ldots,\ K$. После завершения итерации, процедура оптимизации по всем поддеревьям может быть проведена вновь до тех пор, пока не будет найден локальный минимум исходного критерия (2), или не будет достигнута требуемая точность. Очевидно, что неравенство (4) остается справедливым и в случае итерационной оптимизации с поочередным пересчетом значений групп переменных.

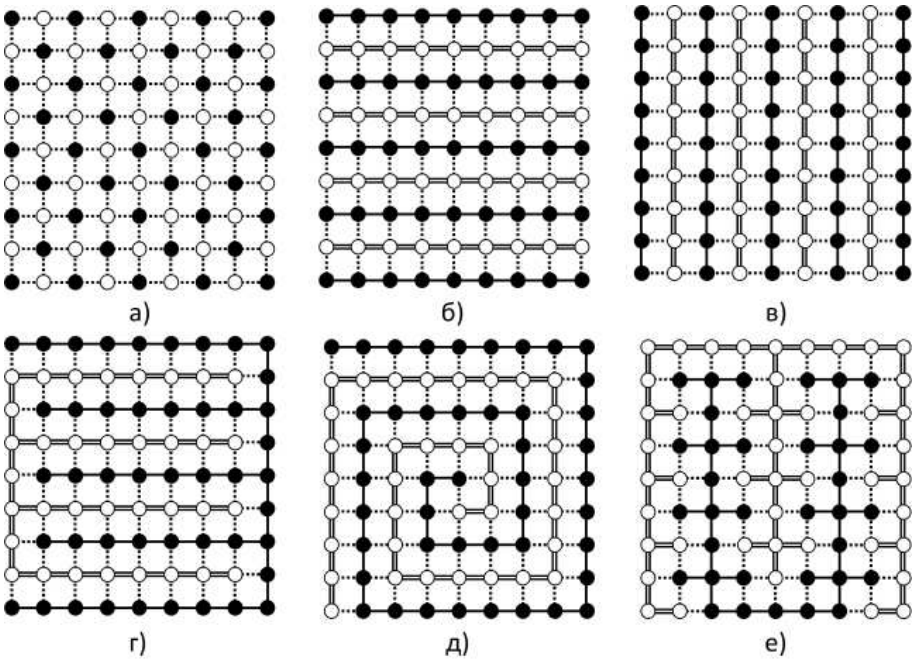


Рис. 1. Примеры разбиения графа соседства целевых переменных на совокупность поддеревьев.

Тривиальным случаем разбиения является независимая оптимизация по каждой из переменных в отдельности (рис. 1, а). Структура графа соседства позволяет проводить такую оптимизацию сразу для половины всех переменных, используя разбиение наподобие шахматной доски. Следующим примером является разбиение на четные и нечетные столбцы (рис. 1, б) или строки (рис. 1, в). И, наконец, на собственно поддеревья различной конфигурации (рис. 1, г, д, е). Очевидно, что как и сама процедура , все эти методы способны находить лишь локальный минимум, и как показывают эксперименты, часто довольно далекий от глобального. Преодолеть этот недостаток отчасти помогает изменение способа разбиения графа соседства переменных при достижении очередного локального минимума, или даже на каждом шаге процедуры. Алгоритмы, основанные на итерационном пересчете групп переменных заметно более устойчивы к локальным экстремумам.

Вторым фактором, определяющим точность процедуры, является выбор начального приближения.

3. Выбор начального приближения для итерационного алгоритма на основе принципа Гаусса-Зайделя

Для выбора начального приближения итерационного алгоритма можно воспользоваться независимой оптимизацией по каждой переменной, используя только информацию наблюдения и не принимая во внимание априорную информацию о гладкости желаемого результата обработки, то есть считая что функции связи

$\gamma_{\mathbf{t}',\mathbf{t}''}(x_{\mathbf{t}'}, x_{\mathbf{t}''}) \equiv 0$, $(\mathbf{t}', \mathbf{t}'') \in G$. В этом случае оператор (1) приобретает вид:

$$\hat{x}_{\mathbf{t}}(Y) = \arg\min_{x_{\mathbf{t}} \in \mathscr{X}} \psi_t(x_t|Y_t), \ \mathbf{t} \in T.$$

Однако такое решение находится, как правило, слишком далеко от глобального минимума, чтобы результат итерационной оптимизации по принципу Гаусса-Зайделя был приемлемым.

В то же время существуют неитерационные процедуры оптимизации на основе ациклического динамического программирования [4] , которые имеют линейную вычислительную сложность относительно количества целевых переменных. Очевидно, что решетчатый граф смежности невозможно заменить деревом без потери основного свойства графа нести информацию о положении каждого элемента массива по отношению к другим элементам. Это препятствие можно обойти, используя при поиске значений искомого массива в каждом столбце изображения индивидуальное дерево соседства элементов, представляющее собой вершины исходного графа, соединенные связями по горизонтали и вертикальными связями лишь в единственном столбце. Процедура обработки изображения [4] построена так, что она направлена на поиск оптимальных значений целевых переменных только в узлах ствола каждого дерева, из которых затем и «собирается» весь результат обработки. Мы будем называть данный алгоритм алгоритмом на основе древовидной аппроксимации графа соседства.

Но так как решение об оптимальном значении целевых переменных в различных столбцах изображения принимается на основе индивидуального графа соседства, решение часто оказывается несогласованным, что особенно ярко проявляется при использовании функций связи вида $\gamma_{t',t''}(x_{t'}, x_{t''}) = \begin{cases} 0, & |x_{t'} - x_{t''}| \leq 1 \\ \infty, & |x_{t'} - x_{t''}| > 1 \end{cases}$, довольно широко применяющимися в задачах построения стереосоответствия. В этом случае полученное в результате оптимизации значение целевой функции может быть равно бесконечности.

Применение итерационной процедуры на основе принципа Гаусса-Зайделя и древовидной декомпозиции графа соседства для уточнения полученного решения позволяет резко улучшить ситуацию. Общая алгоритмическая схема решения задач парносепарабельной оптимизации в этом случае состоит из двух этапов. Сначала при помощи быстрого неитерационного алгоритма оптимизации, определяется начальное приближение, затем при помощи итерационной процедуры на основе древовидной декомпозиции, с изменением способа разбиения графа соседства на поддеревья, находится окончательное решение.

4. Эксперименты

Для экспериментальных исследований алгоритмической схемы, предложенной в предыдущем разделе, мы использовали базу стандартных тестов Middlebury [9] для оптимизационных алгоритмов обработки изображений. В качестве начального приближения для нашего итерационного алгоритма использовалось решение, найденное алгоритмом на основе древовидной аппроксимации графа соседства. Итерационный поиск производился при разбиении графа соседства сначала на отдельный строки

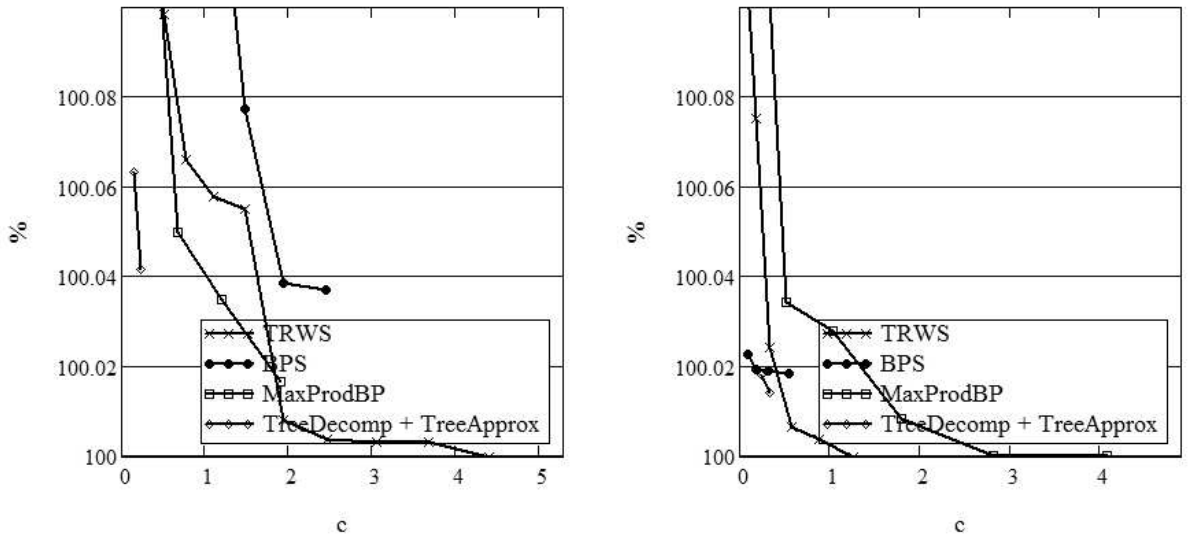


Рис. 2. Результат сравнения алгоритмов для решения задачи сегментации. По вертикали изображены значения целевой функции в процентах от значения, соответствующего глобальному минимуму, по горизонтали – время в секундах.

(рис. 1, в), которое при достижении очередного локального минимума менялось на разбиение на отдельные столбцы (рис. 1, б), затем снова на отдельные строки и т.д. (Tree-Decomp+TreeApprox).

Для сравнения быстродействия и точности работы были выбраны алгоритмы использующие древовидные структуры для оптимизации парно-сепарабельных целевых функций с графом соседства в виде решетки, а именно: последовательный алгоритм Tree-reweighted message passing (TWRS) [7], алгоритм циклического распространении доверия – Max Product Belief Propagation (MaxProdBP) и его модифицированную версию – последовательный алгоритм циклического распространении доверия (BPS) [6]. Результаты работы алгоритма ICM не приводятся в данной работе из за слишком низкой точности.

На рисунке 4 показаны результаты сравнения алгоритмов на примере решения задач интерактивной сегментации изображений на объект и фон [10], для изображений «Person» (600 x 450) (рис. 4, а) и «Sponge» (640 x 480) (рис. 4, б) Как видно из рисунка, для данных задач достигнутое в результате оптимизации значение целевой функции довольно мало отличается от более точного решения, найденного алгоритмами TRWS и BPS, но получено оно гораздо быстрее.

Заключение

Основная идея данной работы заключается в применении принципа динамического программирования для решения оптимизационных задач на графах соседства целевых переменных в виде решетки и создании на его основе эффективных с вычислительной точки алгоритмов.

Как показали эксперименты, использование процедуры оптимизации на основе древовидной декомпозиции решетчатого графа соседства позволяет существенно улучшить точность работы эвристического неитерационного алгоритма древовидной аппроксимации [4], а быстрая сходимость (для задач, сегментации из базы Middlebury потребовались всего две итерации) приводит лишь к незначительному увеличению времени, необходимого для поиска решения.

Список литературы

1. *S. Geman and D. Geman* Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. // IEEE Trans. on PAMI, Vol. 6, No. 6, pp. 721-741, Nov. 1984.
2. *Schlesinger M. I., Flach B.* Some solvable subclasses of structural recognition problems. // Proceedings of Czech Pattern Recognition Workshop,Praha, 2000. – P. 55–62.
3. *Boykov Yu., Veksler O., Zabih R.* Fast approximate energy minimization via graph cuts. // PAMI, 23(11), November 2001. – P. 1222-1239.
4. *Mottl V., Blinov A., Kopylov A., Kostin A.* Optimization techniques on pixel neighborhood graphs for image processing. // Graph-Based Representations in Pattern Recognition (J.-M. Jolion and W.G. Kropatsch, ed.). Computing, Supplement 12. Springer–Verlag/Wien, 1998. P. 135–45.
5. *Шлезингер М.И., Гигиняк В.В.* Решение (max,+)-задач структурного распознавания с помощью их эквивалентных преобразований. // УСиМ.-2007. - № 1. - С.
6. *Felzenszwalb, P.F., Huttenlocher, D.P.* Efficient belief propagation for early vision. In: CVPR. (2004) 261–268
7. *Kolmogorov, V.* Convergent tree-reweighted message passing for energy minimization. In: AISTATS. (2005)
8. *Besag J.E.* On the statistical analysis of dirty pictures (with discussion). Journ. Royal Statist. Soc., B 48, 1986, pp. 259-302.
9. *R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother.* A Comparative Study of Energy Minimization Methods for Markov Random Fields. //ECCV 2006, volume 2, pages 16-29, Graz, Austria, May 2006.
10. *Rother, C., Kolmogorov, V., Blake, A.* «GrabCut» – interactive foreground extraction using iterated graph cuts. // SIGGRAPH 23 (2004) 309–314

Статья поступила в редакцию 25.04.2008

УДК 004.932

СКЕЛЕТИЗАЦИЯ ПОЛУТОНОВОГО ИЗОБРАЖЕНИЯ НА ПРИМЕРЕ ИЗОБРАЖЕНИЙ ОТПЕЧАТКОВ ПАЛЬЦЕВ

© Котик С.В.

Вычислительный центр российской академии наук им. А.А. Дородницына
ул. Вавилова, 40, г. Москва, 119333, Россия

e-mail: *kot_serg@inbox.ru*

Abstract. In this article the grayscale image skeletonization problem is shown by the example of images of fingertips prints. The main idea: the grayscale image skeleton is constructed using some parts of binary images skeletons. One can take these binary images after initial grayscale picture binarization with various brightness levels. Such binary images skeletons fragments give us the opportunity constructing clear binary image of initial grayscale picture. By this clear binary image one can construct the final skeleton; this skeleton allows comparing of particular points of a finger papillary pattern. This final skeleton is concerned as a skeleton of a grayscale image.

Введение

На сегодняшний день в мире существует множество систем распознавания отпечатков пальцев, однако, главным камнем преткновения остаются смазанные, нечеткие, размытые, одним словом, «некачественные» изображения. Для наиболее распространенного метода сравнения отпечатков по особым точкам, которыми являются конечные точки и точки ветвления папиллярного узора, качество полученного изображения особенно актуально. Скелетизация полутонового отпечатка пальца позволяет получать четкое бинарное изображение даже на основе некачественно отсканированной картинки. Скелетизация бинарных изображений исследуется, начиная с начала шестидесятых. В случае же скелета серого изображения очень много неоднозначностей, и не существует четкого определения данного понятия. Подходы к прболеме скелетизации серого изображения можно разделить на две группы. В первой группе изображение рассматривается как трехмерная поверхность, где два измерения – координаты пикселя, а третье – его яркость. Во второй группе алгоритмов производится итерационное удаление точек изображения, пока не останутся срединные оси толщиной в один пиксель (они и будут являться скелетом). Разные алгоритмы основываются на разных определениях свзяности серых изображений. Вот краткий обзор некоторых из них по именам авторов [4].

1. Levi, Montanary. Алгоритм основан на концепции gray weighted distance, определенной авторами.
2. Dyer, Rosenfeld. Авторы определяют скелет так, что он расположен не в областях наибольшей яркости, а получается по центру изображения относительно его границ.
3. Salary, Siy. В данном алгоритме выделяются «гребни» трехмерного изображения, которые и являются скелетом.
4. Maragos, Ziff. Алгоритм, в котором скелет серого складывается из частей скелетов бинарных изображений.

5. Pal, Ghosh. В данном алгоритме определяется функция, принимающая решения, принадлежит ли пиксель скелету, или нет, основанная на функциях от координат и яркости пикселя.
6. Arcelli, Ramella. Алгоритм «утоньшения» изображения, в котором итерационно удаляются пиксели со всех четырех сторон изображения.

Предлагаемый подход наиболее близок работам, описанным в пунктах 3 и 4. Рассматривается алгоритм построения скелета серого изображения на основе бинаризации с различными порогами исходного изображения для получения четкой бинарной картинки, позволяющей проводить распознавание отпечатков пальцев по особым точкам

1. Скелетное представление формы

Скелетом плоской области называется множество ее внутренних точек, имеющих не менее двух ближайших граничных точек. Другое название скелета – серединные оси или симметрические оси области. С каждой точкой скелета связана радиальная функция или ширина, которая задает для этой точки ее расстояние от границы области. На основании скелета и ширины можно однозначно реконструировать саму замкнутую область как объединение всех кругов с центрами в точках скелета и радиусами, задаваемыми радиальной функцией. Поэтому скелет вместе с радиальной функцией является, фактически, одним из способов представления замкнутой области. На рисунке 1 виден скелет фигуры быка.

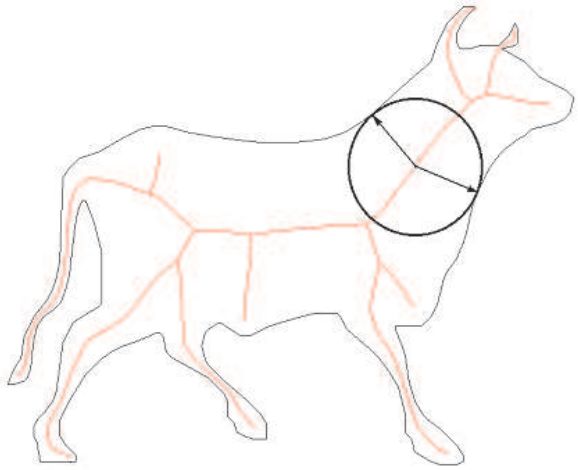


Рис. 1. Скелет фигуры

Скелетом бинарного изображения будем называть объединение всех скелетов компонент изображения. Та часть скелета изображения, которая относится к черным компонентам, составляющим фигуру, называется внутренним скелетом, а часть, относящаяся к белым фоновым компонентам – внешним скелетом бинарного изображения.

Пустым кругом будем называть круг, не имеющий внутренних точек, которые являются граничными для непрерывной фигуры. Максимальным пустым кругом называется пустой круг, не содержащийся целиком ни в каком другом пустом круге. Каждый круг с центром в точке скелета и радиусом, равным радиальной функции в этой точке, является пустым. Такой круг имеет по крайней мере две граничные точки, общие с границей фигуры, поэтому он является максимальным. Это свойство является характеристическим для точек, составляющих скелет изображения: скелет является геометрическим местом точек – центров максимальных пустых кругов.

В качестве наиболее простого случая рассмотрим пример фигуры, границей которой является простой многоугольник (рис. 2).

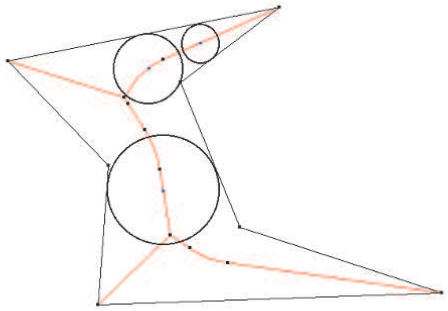


Рис. 2. Скелет многоугольника

Максимальные пустые круги касаются границы по крайней мере в двух точках. Точка касания может быть вершиной многоугольника, либо быть внутренней точкой какого-либо отрезка прямой, являющегося стороной многоугольника. От типа точек касания (вершины или нет) зависит форма серединных осей, составляющих скелет. Представим многоугольную границу, как объединение непересекающихся множеств точек – так называемых сайтов. Сайтами будем считать вершины многоугольников и связные множества остальных точек, т.е. стороны многоугольников без вершин (открытые сегменты). Таким образом, имеет сайты-точки и сайты-сегменты. Пусть максимальный пустой круг касается границы ровно в двух точках. Эти точки принадлежат соответствующим сайтам. Возможны три комбинации типов сайтов: два сайта-сегмента, два сайта-точки, сайт-точка и сайт-сегмент. В окрестности точки – центра пустого круга найдутся другие точки, которые также равноудалены от пары этих же сайтов. Следовательно, в окрестности рассматриваемой точки – центра пустого круга – серединная ось представляет собой линию, равноудаленную от пары сайтов. В зависимости от типажа этих сайтов форма линии будет либо прямая, либо парабола. В скелете также выделяются точки соединения трех или большего числа серединных осей. Каждая такая точка является центром пустого круга, касающегося границы в трех или более точках. Кроме того, существуют еще такие точки скелета,

в которых соединяются всего две оси, одна из которых является параболой, а другая – либо прямая, либо тоже парабола. У кругов с центрами на таких линиях есть общий сайт-точка. Вершинами степени 1 являются концевые точки скелета, из которых выходит только одно ребро, вершинами степени 2 – такие, из которых выходят два ребра и вершинами степени 3 – такие, из которых выходят три ребра.

Из всего вышесказанного следует, что скелет многоугольника представляет собой плоский граф [1], [2].

2. Общий подход

Если рассматривать предлагаемый подход построения скелета серого изображения относительно разбиения всех алгоритмов на два класса, упомянутых во вступлении, то он относится к тем, где изображение рассматривается как трехмерная поверхность. Основная идея построения скелета серого изображения состоит в том, что он собирается на основе частей скелетов бинарных «срезов», полученных при бинаризации исходной картинки по всем 256 уровням яркости. Идея обработки полутонового изображения на основе полученных бинарных «срезов» уже предлагалась ранее в работе [3].

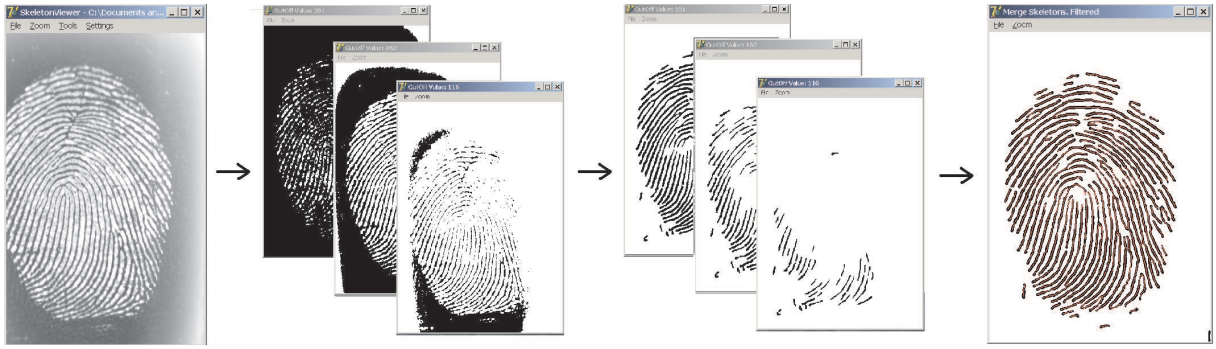


Рис. 3. Серое изображение – бинарные «срезы» – фрагменты бинарных изображений, восстановленные на основе «почищенных» скелетов – итоговый скелет серого изображения и бинарное изображение, на основе которого он строится

На практике не требуется «нарезать» все 256 слоев, и количество слоев и пороговые значения варьируются для достижения наилучшего результата с минимальными временными затратами. Для каждого бинарного «среза» производится «чистка» получившегося скелета по алгоритму, описанному ниже, после чего по оставшимся ветвям и вершинам производится восстановление фрагмента бинарного изображения. Затем все полученные фрагменты накладываются друг на друга и образуют четкое бинарное изображение отпечатка, по которому строится окончательный скелет, позволяющий проводить сравнение особых точек папиллярного узора. Можно провести аналогию с горными массивами, если рассматривать яркость пикселя как координату

высоты – тогда ветви скелета серого изображения будут являться хребтами данного горного массива. На рисунке 3 видны основные шаги алгоритма. Исходное серое изображение отпечатка пальца берется инвертированным для удобства обработки.

3. «Чистка» скелетов бинарных изображений

«Чистка» скелета бинарного среза производится в два этапа. На первом этапе отбрасываются все ветви, толщина которых превышает заданный максимальный порог и все ветви, толщина которых меньше заданного минимального порога. Таким образом, мы избавляемся от всех фрагментов изображения, заведомо не являющихся частью папиллярного узора. Так как для каждой вершины скелета мы знаем радиус максимального пустого круга, описанного вокруг нее, мы легко можем произвести подобную «чистку» (оставляем только те ветви скелета, обе вершины которых удовлетворяют заданным условиям). На втором этапе из получившегося скелета оставляются только вершины и ветви, входящие в цепи с длиной, превышающей порог – таким образом, отбрасываются шумы, появляющиеся при бинаризации и не относящиеся к папиллярному узору. Цепь – все последовательные ветви и вершины скелета, конечной вершиной цепи может служить вершина скелета степени 1, либо степени 3.

Рис. 4. Бинарный «срез» – то же изображение после первого этапа – то же изображение после второго этапа

На обоих этапах «чистки» принимаются во внимание особенности изображений отпечатков пальцев, что позволяет оставлять только те фрагменты скелета, которые имеют отношение к значимой части изображения (рис. 4).

4. Примеры работы алгоритма

На рисунке 5 демонстрируются исходные серые изображения и соответствующие им конечные бинарные изображения, по которым строятся итоговые серые скелеты.

Как видно, по некачественному изображению не удалось восстановить полностью четкую бинарную картинку, однако достаточно много особых точек на картинке присутствуют. На более четких исходных изображениях строится почти идеальная картинка.

Рис. 5. Исходные серые изображения и получившиеся итоговые бинарные изображения, по которым строится скелет

Заключение

Предлагаемый метод скелетизации полутонового изображения позволяет на основе даже не очень качественной серой картинки отпечатка пальца получить пригодное для дальнейшего распознавания бинарное изображение. В дальнейшем планируется уменьшить время работы алгоритма, а также улучшить характеристики работы алгоритма при выделении цепей на этапе «чистки» скелетов бинарных «срезов» исходного изображения. Также планируется рассмотрение приложений построения скелета серого изображения относительно других прикладных задач. Работа поддержана грантами РФФИ 08-01-00670, 08-07-00338.

Список литературы

1. *Местецкий Л.М.* Векторизация бинарных растровых изображений на основе аппроксимации // Доклады X Всероссийской конференции «Математические методы распознавания» (ММРО-10). Москва, 2001
2. *Местецкий Л.М.* Непрерывный скелет бинарного растрового изображения // Труды международной конференции «Графикон-98». Москва, 1998
3. *С.В. Котик, Л.М. Местецкий* Сжатие полутоновых изображений рукописных документов на основе кодирования по изолиниям яркости // Доклады XII Всероссийской конференции «Математические методы распознавания» (ММРО-12). Москва, 2005
4. *4. Samira S. Mersal, Ahmed M. Darwish* . A New Parallel Thinning Algorithm For Gray Scale Images //Works of IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing – NSIP'99. Antalya, 1999

Статья поступила в редакцию 29.04.2008

УДК 004.912

СИСТЕМА АВТОМАТИЧЕСКОГО ИНДЕКСИРОВАНИЯ И РЕФЕРИРОВАНИЯ ТЕКСТОВЫХ ДОКУМЕНТОВ

© Кравцов А.А., Липницкий С.Ф., Степура Л.В.

Объединенный институт проблем информатики НАН Беларуси
ул. Сурганова, 6, г. Минск, 220012, Беларусь

e-mail: *kravtsov@newman.bas-net.by, lipn@newman.bas-net.by, stepura@newman.bas-net.by*

Abstract. Theoretical preconditions and program realization of computer system of indexing and abstracting of text documents on the basis of knowledge base use about a subject domain in the form of an is situational-syntagmatic network are considered. The approach offered by authors to calculation informative lexemes and the offers, based on comparison of statistical characteristics of word forms in the reviewed text and a full corpus is used in this system. The computer system can be used in scientific and technical libraries for the automated indexing and annotation of scientific and technical articles and books, and also Internet users for familiarity with the content of text documents.

Введение

Постановка проблемы. Эффективность процессов автоматического индексирования и реферирования текстовых документов существенным образом зависит от интеллектуальности программной системы, т. е. ее способности накапливать и использовать знания с целью «компьютерного понимания» текстов и их фрагментов. Известно, что понимание текста человеком связано со знанием языка, с одной стороны, и распознаванием ситуативного контекста, с другой. При отсутствии ситуативных знаний восприятие текста возможно только на лингвистическом уровне. В связи с этим возникает проблема построения такой модели базы знаний, которая обеспечила бы реализацию эффективных алгоритмов индексирования и реферирования текстов с учетом ситуативных связей между понятиями предметной области.

Состояние проблемы. Основной задачей, возникающей при автоматическом индексировании и реферировании текстовых документов, является вычисление информативности слов и предложений. Существующие подходы к решению этой задачи основаны главным образом на анализе (статистическом, лингвистическом, семантическом) самого исходного текста без привлечения знаний о предметной области. Использование таких знаний позволяет существенно повысить эффективность функционирования информационных систем различного назначения за счет их интеллектуализации [1, 2, 3, 4].

Нерешенные задачи, цель и задачи статьи. В рамках рассматриваемой проблемы нерешенными являются следующие задачи:

- вычисление информативности слов и предложений текста с использованием накопленных знаний в виде корпусов текстов по различной тематике;
- создание словарей базы знаний на основе моделирования ситуативных связей между понятиями предметной области.

Решение этих задач является основной *целью* данной статьи.

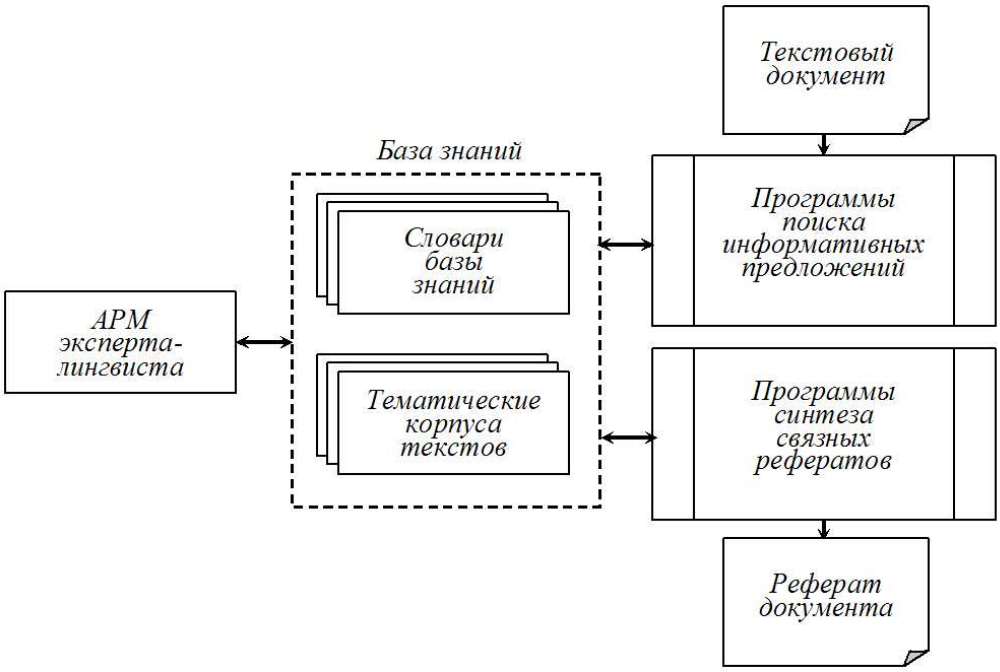


Рис. 1. Структурная схема системы индексирования и реферирования текстовых документов

1. Архитектура системы

В состав системы автоматического индексирования и реферирования текстовых документов входят следующие основные структурные компоненты (рис. 1):

– автоматизированное рабочее место (АРМ) эксперта-лингвиста;
– база знаний, включающая систему словарей;
– программы поиска информативных слов и предложений в текстовых документах;
– программы синтеза рефератов.

АРМ эксперта-лингвиста – это комплекс программ, предназначенный для автоматизированного формирования и актуализации баз данных и знаний.

В базе данных системы накапливаются и хранятся текстовые документы, на основе которых формируются тематические корпуса текстов для всех разделов предметной области. (Тематический корпус – это совокупность текстов по конкретной тематике; множество всех тематических корпусов – это полный корпус текстов). Если в полном корпусе текстов представлен только один тематический, то полный корпус дополняется корпусом текстов с общеупотребительной лексикой, т. е. полный корпус текстов должен содержать, как минимум, два тематических. На основе статистической обработки корпусов текстов формируются словари базы знаний.

В состав базы знаний системы входят частотный словарь словоформ, словарь словоизменительных парадигм, словарь синонимов и словарь ситуативных связей.

В словаре словоформ каждой словоформе поставлены в соответствие:

– абсолютная частота словоформы в полном корпусе текстов;
– абсолютные частоты словоформы во всех тематических корпусах текстов;
– номер (код) парадигмы.

В первоначальном состоянии каждая словоформа словаря образует отдельную парадигму. После объединения некоторых (или всех) словоформ в словоизменительные парадигмы словоформам присваивается номер парадигмы, элементом которой эта словоформа является.

Словарь словоформ используется при определении абсолютной частоты словоформы в реферируемом документе. При этом предусмотрены два варианта. В первом варианте (при построении общего реферата текстового документа) частота словоформы подсчитывается непосредственно в документе. Во втором варианте (в случае создания тематически ориентированного реферата) словоформам документа приписываются частоты из соответствующего тематического корпуса текстов.

Словарь парадигм служит для поиска всех словоформ парадигмы после нахождения словоформы и ее кода в словаре словоформ. Частоты всех словоформ найденной парадигмы при этом суммируются, и словоформе приписывается суммарная частота. Аналогичным образом используется словарь синонимов. Процедура определения частот слов используется при вычислении их информативности.

В словаре ситуативных связей представлены упорядоченные пары слов, каждой из которых поставлена в соответствие абсолютная частота этой пары в предложениях полного корпуса текстов. Словарь используется при синтезе реферата и является реализацией модели базы знаний.

2. Моделирование знаний о предметной области

Построим модель базы знаний системы индексирования и реферирования текстовых документов в виде ситуативно-синтагматической сети. Это граф, вершинами которого являются информативные лексемы, а ребрами – ситуативные связи между лексемами. Информативность лексем в сети определяется на основе сопоставления их частотных характеристик в корпусах текстов.

Пусть имеется некоторое непустое множество текстов входного языка (набор текстов по конкретной тематике). Сформируем текст Th, объединив все множества предложений каждого из этих текстов, и назовем его *тематическим корпусом* текстов. Поскольку в информационной системе представлено, как правило, несколько таких корпусов, будем обозначать их Th_i (i – номер корпуса). Объединение $Fu = \bigcup\limits_{i=1}^{n} Th_i$ всех тематических корпусов назовем *полным корпусом* текстов.

С учетом введенных обозначений ситуативные связи формализуем в виде ситуативного отношения на множестве лексем.

Обозначим через Str множество всех лексем полного корпуса текстов Fu. Тогда отношение толерантности Θ (рефлексивное и симметричное бинарное отношение) на множестве Str назовем *ситуативным отношением* в полном корпусе текстов Fu, если любая упорядоченная пара лексем (μ, ν) из множества Str является элементом отношения Θ тогда и только тогда, когда вероятность совместной встречаемости

лексем μ и ν в корпусе текстов Fu не меньше некоторого порогового значения (*уровня* ситуативной связи).

Под совместной встречаемостью двух лексем здесь понимается наличие этих лексем (или их синонимов) в одном и том же предложении корпуса Fu. Граф $S_{\text{сит.}}$ ситуативного отношения будем называть *ситуативно-синтагматической сетью.*

3. Индексирование текстовых документов

Процесс индексирования текстовых документов включает два этапа. На первом этапе в тексте выявляются информативные словоформы, а на втором – ключевые слова (лексемы).

Информативность словоформы определим как условную вероятность того, что эта словоформа извлечена из индексируемого текста (или релевантного ему тематического корпуса текстов) Th при условии, что она уже извлечена из полного корпуса текстов Fu:

$$P(S_{Th}/S_{Fu}) = \frac{P(S_{Th} \cdot S_{Fu})}{P(S_{Fu})} = \frac{P(S_{Th}) \cdot P(S_{Fu}/S_{Th})}{P(S_{Fu})} \tag{1}$$

В формуле (1) задействованы следующие события:

- S_{Th} – словоформа извлечена случайным образом из тематического корпуса текстов (или текстового документа) Th ($Th \in Fu$);
- S_{Fu} – словоформа извлечена из полного корпуса текстов Fu.

Пусть n_{Th}, n_{Fu} – абсолютные частоты встречаемости словоформы в индексируемом тексте (или релевантном ему тематическом корпусе текстов) Th и полном корпусе текстов Fu соответственно. Тогда нетрудно установить [5, 6], что при достаточно больших объемах корпусов текстов Th и Fu формула для вычисления информативности словоформы примет вид

$$I_{Th} \approx \frac{n_{Th}}{n_{Fu}}. \tag{2}$$

При вычислении информативности лексемы в числителе и знаменателе формулы (2) находится сумма частот всех словоформ парадигмы для данной лексемы с учетом словоизменения, зафиксированного в словаре парадигм системы, и синонимии в корпусах текстов Th и Fu соответственно.

В разработанной версии программной системы результаты индексирования предъявляются пользователю в виде списка лексем (ключевых слов) с их информативностью (в процентах). Список может быть отсортирован по алфавиту или по убыванию информативности лексем.

4. Реферирование текстовых документов

При построении реферата текста формируется маршрут его информативности и семантический след. Формально понятия маршрута информативности и семантического следа текста определим следующим образом.

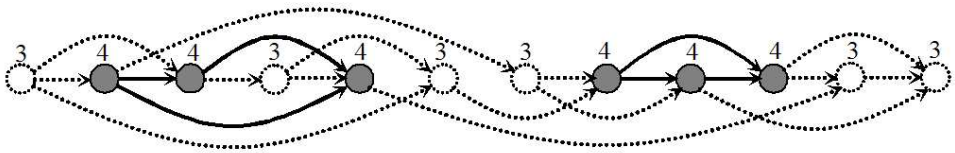


Рис. 2. Пример семантического следа текста в графе информативности

Пусть имеется текст (т. е. кортеж предложений) Te. Вычислим информативность лексем всех предложений текста Te. Поставим в соответствие каждому предложению из Te длину вектора, компонентами которого являются показатели информативности всех информативных слов этого предложения. Эту длину будем считать характеристикой информативности данного предложения. Исключим из текста Te все неинформативные предложения, т. е. предложения, показатель информативности которых меньше некоторого числа (порога информативности). В результате получим кортеж предложений (в порядке их следования в Te) $Te_{\text{инф.}} = \langle \pi_1, \pi_2, \ldots, \pi_n \rangle$. Кортеж $Te_{\text{инф.}}$ будем называть *маршрутом информативности* текста Te.

Построим орграф $Gr_{\text{инф.}}$, считая все предложения маршрута информативности $Te_{\text{инф.}}$ его вершинами. Всякую пару вершин π_i, π_j $(i < j, 1 \leq i \leq n-1, 2 \leq j \leq n)$ соединим дугой (π_i, π_j) тогда и только тогда, когда в ситуативно-синтагматической сети $S_{\text{сит.}}$ существует хотя бы одна пара вершин (подцепочек предложений π_i и π_j соответственно), соединенных ребром, которое указывает на существование ситуативной связи между этими подцепочками.

Орграф $Gr_{\text{инф.}}$, на множестве вершин которого определен линейный порядок, соответствующий порядку предложений в маршруте информативности $Te_{\text{инф.}}$, будем называть *графом информативности* текста Te.

Маршрут информативности $Te_{\text{инф.}}$ является основой для построения реферата текста Te. Для регулирования объема маршрута информативности и выявления в нем монотематических фрагментов построим семантический след текста.

Семантическим следом Tr текста Te будем называть подграф графа информативности $Gr_{\text{инф.}}$, вершинами которого являются все вершины орграфа $Gr_{\text{инф.}}$ с числом дуг, инцидентным им, не меньше некоторого n_0 (рис. 2).

На рис. 2 каждая вершина графа информативности $Gr_{\text{инф.}}$ помечена числом, обозначающим количество инцидентных ему дуг (в данном случае $n_0 = 4$). Вершины и дуги орграфа $Gr_{\text{инф.}}$, не вошедшие в состав семантического следа Tr, изображены пунктирными линиями. Связные подграфы семантического следа соответствуют двум монотематическим фрагментам текста.

Семантический след текста – это модель реферата текстового документа.

5. Программная реализация системы

Рассматриваемая версия системы индексирования и реферирования текстовых документов реализована на языке программирования C++. Система может обрабатывать тексты в форматах html, txt, rtf, doc. Обработка документов формате pdf

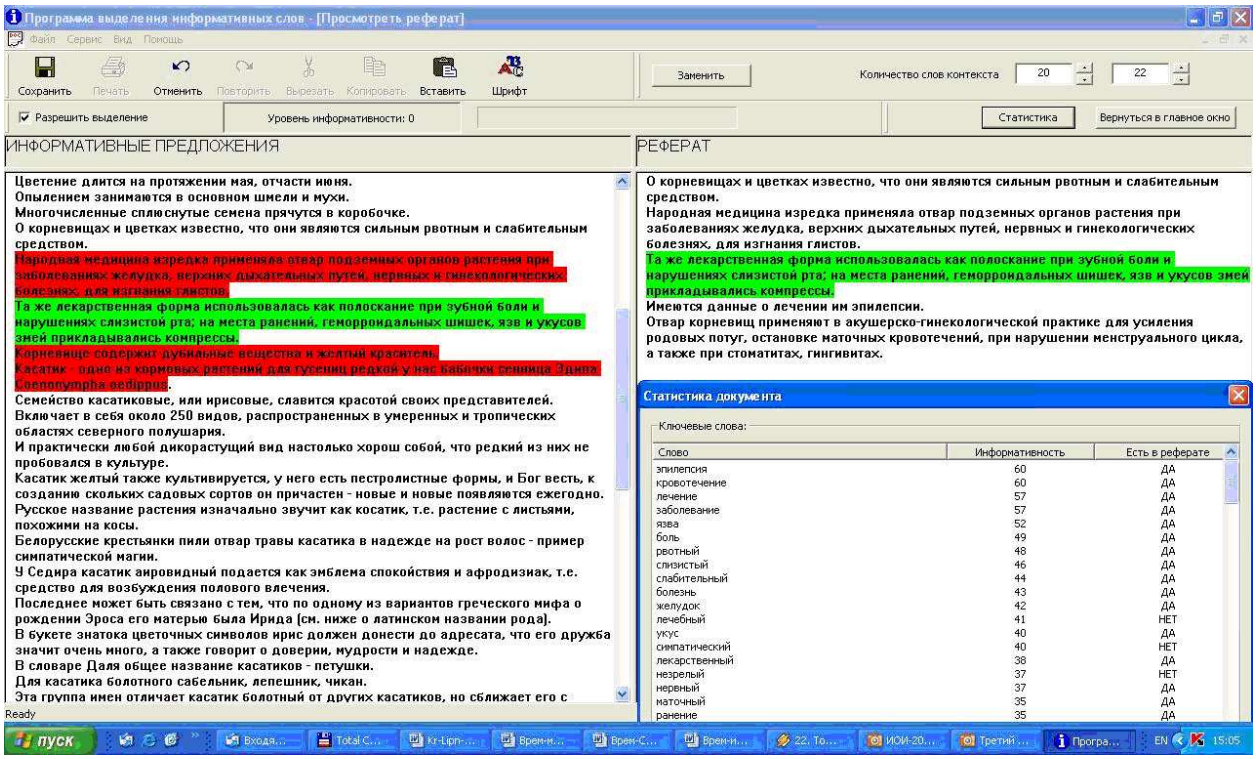


Рис. 3. Результаты индексирования и реферирования текстового документа

возможна после их конвертирования в поддерживаемые системой форматы с помощью существующих программных средств.

Программная система представляет собой исполняемое приложение Windows, в рабочей области которого отображается открытый документ и результаты его индексирования и реферирования. На панели инструментов приложения имеются элементы управления «Уровень информативности», «Порог некорректности», «Объем реферата», «Количество слов контекста». С помощью этих элементов пользователь задает необходимые ему параметры системы.

На рис. 3 представлены результаты индексирования и реферирования текста «Касатик аировидный» (о лекарственном растении). В левом окне представлены все предложения текста (при уровне информативности, равном нулю). В правом окне приведены реферат и список ключевых слов, упорядоченных по убыванию их информативности. В левом окне выделен контекст предложения из реферата, отмеченного пользователем в правом окне.

Данная версия системы обеспечивает индексирование и реферирование текстов в двух режимах. В первом режиме формируется общий реферат, представляющий основное содержание документа. Во втором режиме создается реферат, ориентированный на тематику пользователя по рубрикам: авиация и космонавтика, астрономия, биология, военные технологии, география, геология, дистанционное зондирование Земли, лингвистика, математика, общая медицина, поиск и обработка текстовой

информации, психология, социология, технология и промышленность, транспорт и связь, физика, философия, химия, энергетика и строительство. Количество рубрик может быть увеличено за счет создания новых тематических корпусов текстов.

Заключение

Полученные результаты.

1. Разработана модель базы знаний системы индексирования и реферирования текстовых документов.
2. Предложен подход к вычислению информативности словоформ и предложений текста на основе использования корпусов текстов различной тематической направленности.
3. Разработана программная система, в которой реализованы полученные авторами теоретические результаты.

Перспективы дальнейших исследований. Исследования могут быть продолжены в направлении индексирования и реферирования текстов на различных входных языках с выдачей результатов на языке, отличном от входного.

Список литературы

1. *Удо Хан, Индерджиет Мани.* Системы автоматического реферирования // Открытые системы, № 12 [Электронный ресурс], 2000 г. – Режим доступа: http://www.osp.ru/os/2000/12/178370 – Дата доступа: 3.04.2008.
2. *Hulth, A. et al* Automatic keyword extraction using domain knowledge // Lecture notes in computer science, Vol. 3930/2006, 2006, P. 633-641.
3. *Демьянков В.З.* Интерпретация, понимание и лингвистические аспекты их моделирования на ЭВМ // М.: Изд-во Моск. ун-та, 1989 г., 172с.
4. *Ильин Н, Киселев С., Рябышкин В., Танков С.* Технологии извлечения знаний из текста // Открытые системы, № 6 [Электронный ресурс], 2006 г. – Режим доступа: http://www.i-teco.ru/article104.html. – Дата доступа: 3.04.2008.
5. *Липницкий С.Ф.* Семантический анализ текста на основе ситуативно-синтагматической сети // Информатика, № 2, 2005 г., С. 102-110.
6. *Кравцов А.А., Липницкий С.Ф., Насуро Д.Р.* Синтез рефератов текстовых документов на основе ситуативно-синтагматической сети // Искусственный интеллект, № 2, 2006 г., С. 172-175.

Статья поступила в редакцию 27.04.2008

УДК 004.9

ПРИНЯТИЕ РЕШЕНИЙ В ОПЕРАТИВНЫХ ЗАДАЧАХ РЕГИОНАЛЬНОГО УПРАВЛЕНИЯ

© Краснопрошин В.В., Виссия Х., Вальвачев А.Н.

Белорусский государственный университет
220050, г.Минск, пр.Независимости, 4.

e-mail: *krasnoproshin@bsu.by*

Abstract. Problems arising up at the decision of tasks of operative regional and department management are considered in the work. Management methods are offered for thousands of remote organizations on the basis of heterogeneous information.

Введение

В процессе становления рыночных отношений в странах СНГ регионы, министерства и ведомства (далее – субъекты) получили достаточно прав и возможностей для самостоятельного развития. Соответственно на них перешла ответственность и многочисленные риски. Тем не менее, прерогативой органов государственного управления осталось анализ, прогнозирование и «мягкая» коррекция их состояния [1].

Теоретические вопросы управления субъектами в новых условиях исследуются в работах А.С. Малина, Ю.Г. Волкова, А.Л. Гапо ненко, Ю.С. Дульшикова, В.П. Кнорринга и др. Проблемы построения компьютерных систем управления рассмотрены в трудах Д. Клира, А.В. Царегородцева, В.Г. Тоценко, Г.С. Осипова, П.Ф. Иванова и др.

В целом задачи регионального управления можно разделить на три группы [2]:

- стратегические задачи, связанные с перспективами развития региона, стабильностью и преемственностью власти;
- оперативные задачи, ориентированные на текущее организационное и финансово-ресурсное обеспечение выполнения планов и проектов;
- чрезвычайные задачи, связанные с необходимостью предотвращения и устранения последствий воздействия разрушительных природных или техногенных факторов, возникновением и неконтролируемым развитием острых социально-политических проблем и конфликтов.

В динамической среде, формируемой глобализацией, изменением климата и другими процессами со слабо предсказуемыми последствиями, особую значимость приобретают задачи второй и третьей групп, которые имеют много общего. Для краткости определим их как оперативные задачи. Решение оперативных задач затруднено рядом естественных факторов, основными из которых являются:

- выработка правильного управления требует анализа состояния тысяч удаленных социально-экономических систем (далее – организаций), входящих в состав субъектов управления;
- организации удалены от центра на значительные расстояния;

- количество и семантика возможных состояний организации и диагностических показателей для их определения носят динамический характер;
- значения показателей относятся к различным типам данных;
- время выработки управления является критическим параметром и, как правило, крайне ограничено [3].

Коротко основную проблему можно сформулировать так: как выработать управление на основе разнородных данных для тысяч удаленных организаций в минимальное время, значение которого не зависит от количества организаций. Очевидно, что решить ее можно только с использованием компьютеров и современных средств связи. В данной работе предложена технология выработки управления, построенная на основе теории нечетких множеств и реализованная в рамках многоагентного подхода.

1. Постановка задачи

Пусть имеется регион, центр управления регионом С и множество входящих в его состав удаленных организаций P^1, P^2, P^n. Изменения среды порождают задачу S, которая влияет на гомеостазис С, характеризуется переменными X^1, X^2, X^m и требует выработки управления U^1, U^2, U^k для объектов P^1, P^2, P^n с целью удержания их в одном из заданного множества состояний V^1, V^2, V^r.

Требуется разработать технологию выработки управления U^i для организации P^j на основе X^1, X^2, X^n, инвариантную количеству организаций.

Условия решения:

- количество возможных состояний организаций зависит от семантики S;
- количество переменных X^1, X^2, X^n зависит от семантики S;
- значения параметров X^1, X^2, X^n разнородны $(string, integer, real, boolean)$.

Предположения:

- центр и организации позиционированы в ведомственной сети, в инфраструктуре Интернет или другой системе коммуникаций, которые в совокупности обозначим как Env;
- объекты заинтересованы в передаче центру достоверной информации;
- центр имеет возможности реализовать выработанное управление.

Основное требование к решению: технология должна быть простой, гибкой, реализуемой на типовых IBM PC под управлением OS Windows и понятной как руководству региона, так и удаленным организациям.

2. Базовые процессы решения

В [4] показана возможность построения систем обработки распределенных данных в иерархических организационных структурах на основе трех базовых процессов:

P1 : построение модели задачи $modS$ и ее доставка организациям P^1, P^2, P^n;

P2 : включение в модель $modS$ требуемой информации $< X^i >$ и возврат обогащенной модели в центр C;

P3 : выработка управления U^j для P^i на основании $< X^i >$.

Процессный подход позволяет выделить основные группы задач и связать их с соответствующими моделями и алгоритмами, а так же построить общую схему автоматизированного решения.

3. Организационно-коммуникативная схема решения

В качестве базы для реализации процессов P^1 – P^3 предлагается общая организационно-коммуникативная схема решения (ОКС). Определим ОКС как совокупность организационных процедур, участников решения (акторов), процессов и компьютерных технологий, которые обеспечивают решение общей задачи в инфраструк

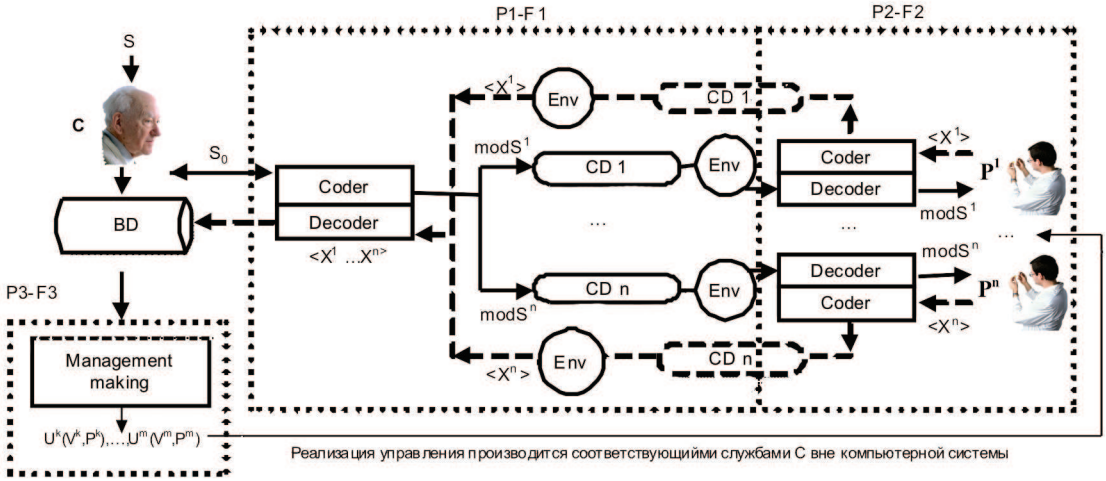


Рис. 1. Организационно-коммуникативная схема решения.

В схеме присутствуют акторы (центр, организации), базовые процессы решения ($P1 - P3$), алгоритмы для их реализации ($F1 - F3$) и соответствующие информационные потоки.

В отличие от известных моделей коммуникации (Шеннона-Уивера, Осгуда-Шрама и т.д.) [5], ОКС обеспечивает:

- цельный взгляд на разработку гибких региональных систем управления, так как процессы $P1 - P3$ реализуют полный жизненный цикл ПО;
- распараллеливание процессов получения информации от удаленных организаций;
- сопоставление процессов $P1 - P3$ и алгоритмов их реализации $F1 - F3$;
- увеличение количества организаций при постоянном времени выработки управления (за счет параллельной обработки);
- коммуникации человек-программа, программа-программа, программа-человек в рамках единообразного удаленного диалога;
- отображение в архитектуру соответствующей компьютерной системы.

Реализация ОКС в первую очередь требует построения моделей акторов и средства их удаленного общения.

4. Модели

Модели акторов построены на основе онтологического подхода и изначально ориентированы на отображение в соответствующие интерфейсы и их быструю реализацию в среде .Net.

Модель центра:

$$C = (adrC, adrP, admC, BA, BD, Bufer, S, X, V, M, U, DC) \quad (1)$$

где: $adrC$ – адрес центра Интернет; $AdrP$ – адреса управляемых объектов; $admC$ – администрация центра; BA – база алгоритмов формализации данных и принятия решений; BD – база данных; $Bufer$ – буфер; S – задача; X – диагностические переменные; V – идентификаторы возможных состояний; M – образы возможных состоний; U – управления; DC – посредник, обеспечивающий информационное взаимодействие между центром и объектами.

Модель посредника:

$$CD = (adrP^i, adrC, S, Q, X, < X^i >, Box) \quad (2)$$

где: Q – запросы для получения значений X; $< X^i >$ - значения X в i –ой организации; Box – контейнер, который на практике часто используют для передачи дополнительной информации. Например, в оффшорном программировании это может быть фрагмент разрабатываемой системы, затребованной в центр для экспертной оценки степени ее готовности. В каждой задаче выбора управления посредник CD участвует два раза для каждого объекта (см. рис.1).

Модель объекта управления:

$$Pi =< adrP^i, admP^i, < X >) \quad (3)$$

где: $adrPi$ – адрес объекта в Интернет; $admP^i$ - администрация организации.

Реализация ОКС на основе процессов $P1-P4$ в рамках моделей C, CD, P требует разработки соответствующих алгоритмов. Разнородность исходных данных говорит о возможности применения для их нормализации теории нечетких множеств [6].

5. Алгоритмы

Алгоритм $F1$ (рис.2) формирует в центре C модель задачи $modS$: значения V, U, X, Q, подмножество бифуркационных переменных $\underline{x}$, функции принадлежности μ^1 для X и μ^2 для x. Множество эталонных образов M строится автоматически на основе характеристик V и X. Затем модель $modSQ, X$ отправляется удаленной организации.

Алгоритм $F2$ (рис.3) формирует P^i нечеткий образ $< X >$, характеризующий текущее состояние удаленной организации по заданным параметрам, который затем отсылается в центр.

Алгоритм $F3$ (рис.4) сравнивает образ $< X >$ с каждым из эталонных образов M и выбирает наиболее похожий образ. По номеру этого образа выбирается состояние V. Затем корректируется номер V с помощью алгоритма μ^2 в зависимости

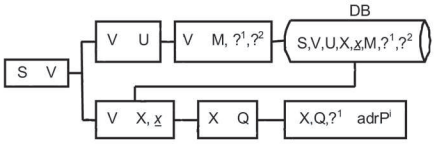


Рис. 2. Схема алгоритма построения модели.

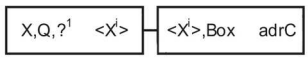


Рис. 3. Схема алгоритма обогащения модели.

от значений бифуркационных переменных x. Номер идентификатора образа соответственно увеличивается или уменьшается. Согласно уточненному номеру выбирается управление U.

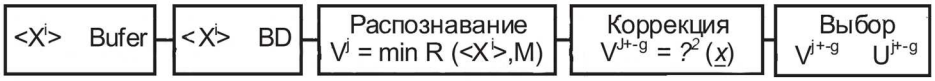


Рис. 4. Схема алгоритма распознавания и выбора управления.

Представленные алгоритмы носят открытый характер и могут быть реализованы в рамках различных подходов. Ниже приведен вариант, ориентированный на закрытые ведомственные сети.

6. Архитектура системы

Для реализации ОКС на основе разработанных моделей и алгоритмов целесообразно применить многоагентный подход, изначально ориентированный на работу с распределенными данными [7]. Соответственно на основе процессов $P1 - P3$, моделей C, CD, P и алгоритмов $F1 - F3$ построены агенты $Modeller, Miner$и$Managing$. Архитектура системы, построенной на их основе показана на рис. 5.

Данная архитектура реализована на языке $C\sharp$ и применялась для решения ряда задач ведомственного управления, в частности для решения задачи «Контрольно-наблюдательное дело» с целью оценки деятельности инспекций МЧС. Практика показала, что основным достоинством данной архитектуры является оперативность настройки на задачу, удобство ввода информации персоналом любого уровня в удаленных организациях и получение результата в реальном режиме времени. Архитектура проста и прозрачна, что делает ее понятной как для руководства, так и для персонала распределенных организаций

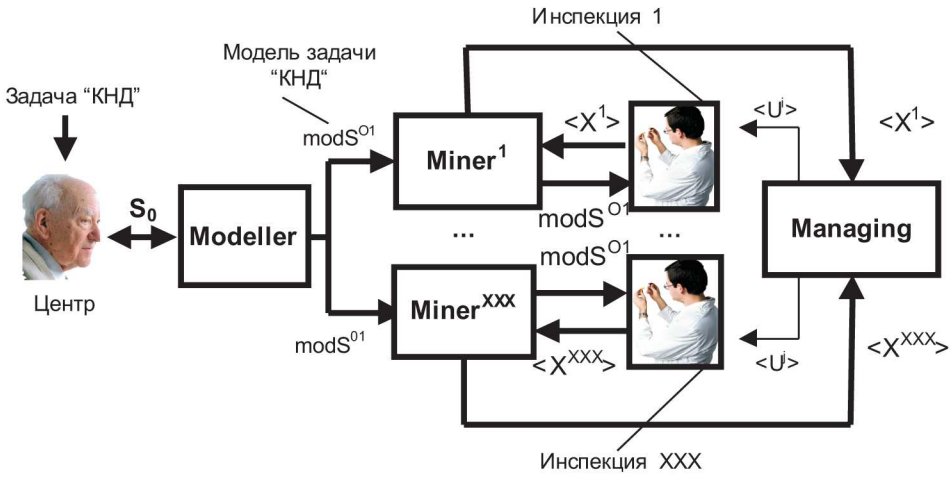


Рис. 5. Схема алгоритма распознавания и выбора управления.

Заключение

В работе рассмотрены проблемы решения оперативных задач регионального и ведомственного управления. Показана актуальность разработки технологии решения оперативных задач, инвариантной количеству распределенных объектов управления.

Предложен вариант такой технологии, обеспечивающий:

- построение необогащенной модели ситуации и ее доставку из центра в организации региона;
- включение в модель требуемой информации на местах и доставка обогащенной модели в центр управления регионом;
- интеграция локальных моделей организаций в глобальную модель региона;
- формирование цельного взгляда на состояние региона;
- выработку управления, адекватного состоянию организаций.

Первоначально сформированная организационно-коммуникативная схема решения позволила отобразить модели в интерфейсы, а алгоритмы - в функциональные программные модули, которые в целом составили гибкую технологию, ориентированную на типовые компьютеры и $OS\ WindowsXP$.

Многоагентный подход к реализации позволил использовать технологию для решения практических задач как в закрытых (ведомственных), так и в открытых (Internet) сетях.

Технология успешно использовалась при решении ряда задач оперативного управления подразделениями МЧС.

Список литературы

1. Региональная экономика и управление / Под общ. ред. А.Л. Гапоненко и Ю.С. Дульшикова – М.: РАГС, 2006. – 616 с.
2. *Иванов П.Ф.* Информационно-аналитическое обеспечение законодательной деятельности: проблемы и опыт. Ан.Вестн. Сов.Фед. ФС РФ. – 2002. – № 2 (157). – стр. 23.
3. *Кнорринг В. И.* Теория, практика и искусство управления. – М.: НОРМА, 2001. – 528 с.
4. *Краснопрошин В.В., Шаках Г., Вальвачев А.Н.* Интеграция распределенных экспертных знаний: проблемы и решения // Информатика. – Минск, 2004. – № 1, – С. 45-53.
5. *Василик М.А.* Теории коммуникаций. – М.: Гордарики, 2003. – 616 с.
6. *Кофман А.* Введение в теорию нечетких множеств. М.: Радио и связь, 1982. – 432 с.
7. *Wooldridge M.* Multiagent Systems. – John Wiley & Sons, 2002. – 340 p.

Статья поступила в редакцию 30.04.2008

УДК 004.9311

МЕТОДЫ РЕГУЛЯРИЗАЦИИ В ЗАДАЧЕ ВОССТАНОВЛЕНИЯ НЕСТАЦИОНАРНОЙ РЕГРЕССИОННОЙ ЗАВИСИМОСТИ

© Красоткина О.В.[1], Моттль В.В.[2]

[1]Тульский государственный университет
факультет Кибернетики
пр-т Ленина, 92, г. Тула, 300600, Россия

e-mail: *krasotkina@uic.tula.ru*

[2]Вычислительный центр РАН им. Дородницына
ул. Вавилова, 42, г. Москва, , Россия

e-mail: *vmottl@yandex.ru*

Abstract. The problem of finding the most appropriate subset of features or regressors is the generic challenge of Machine Learning problems like regression estimation or pattern recognition. We consider the problem of time-varying regression estimation, which implies also the inevitable necessity to choose the individual appropriate levels of model volatility in each of regressors, ranging from the full stationarity of instant models to their absolute inde-pendence in time. The problem is considered from the Bayesian point of view as that of estimating the sequence of regression coefficients associated with the hidden vector state of a stochastic linear dynamic system, whose a priori model includes parameters responsible for both the size of the subset of active regressors and the time-volatility factors of regression coefficients at them. The proposed technique of adaptive time varying regression estimation is built as that of estimating both the state and parameters of the hidden state-space model.

Введение

Задача восстановления регрессионной зависимости является одной из ключевых в области интеллектуального анализа данных.

В этой задаче подлежащий анализу сигнал состоит из нескольких компонент. Одна из них y_t, понимаемая как выходная, является зашумленной линейной комбинацией $y_t \cong \mathbf{x_t^T}\boldsymbol{\beta_t}$ остальных переменных $\mathbf{x}_t = (x_t^1, x_t^2, ..., x_t^n)$, которые называются признаками или регрессорами. Одним из ключевых аспектов этой задачи является выбор подходящего подмножества $\hat{I} \subseteq I$ из всего доступного множества признаков $\{x_t^i, i \in I\}$ [1]. Для многих приложений типичны ситуации, когда совокупность наблюдений $t \in T$ рассматривается как упорядоченная последовательность $T = \{1, ..., N\}$ и искомая линейная зависимость изменяется во времени

$$y_t = \sum\nolimits_{i=1}^{n} \beta_t^{(i)} x_t^{(i)} + e_t = \mathbf{x}_t^T \boldsymbol{\beta}_t + e_t. \qquad (1)$$

Требуется оценить неизвестные коэффициенты регрессии $\boldsymbol{\beta}_t = (\beta_t^1, \beta_t^2, ..., \beta_t^n)$ которые, вообще говоря, меняются во времени. Естественно, число переменных в такой задаче оказывается очень большим и значительно превосходит число наблюдений. Таким образом, оказывается невозможным оценить коэффициенты регрессии без дополнительной регуляризации задачи, то есть без принятия дополнительных априорных предположений о скрытой последовательности коэффициентов регрессии. Идеи

регуляризации могут быть подсказаны практикой. Во-первых, для большинства приложений типично, что все число возможных регрессоров много больше числа регрессоров, реально присутствующих в модели, так что большинство коэффициентов регрессии равно нулю. Таким образом, проблема отбора регрессоров остается актуальной и для случая нестационарной регрессии. во-вторых,предположим, что скрытые коэффициенты регрессии меняются достаточно плавно. Это предположение существенно сужает область поиска степени нестационарности коэффициентов. Наконец, для многих практических ситуаций типично, что только небольшое число коэффициентов регрессии меняется во времени, а большинство из них остается практически постоянными. Если мы знаем список регрессоров, имеющих постоянные коэффициенты, мы можем существенно уменьшить реальное число оцениваемых переменных. В этой статье мы предлагаем процедуру, которая автоматически оценивает подмножество активных регрессоров, определяет среди них регрессоры с действительно меняющимися коэффициентами и, наконец, оценивает степень нестационарности последних.

1. Flexible Least Squares критерий для задачи нестационарной регрессии

Задача нестационарной регрессии интенсивно изучалась в литературе. Общепринятым средством оценивания нестационарных моделей является метод FLS - Flexible Least Squares [2]

$$J\big(\beta_t^{(i)},\ t=1\ldots,N,\ i\in\hat{I}\mid\hat{I},\rho\big)= \\ =\sum_{t=1}^{N}\Big(y_t-\sum_{i\in\hat{I}}\beta_t^{(i)}x_t^{(i)}\Big)^2+\rho\sum_{t=2}^{N}\sum_{i\in\hat{I}}\Big(\beta_t^{(i)}-\beta_{t-1}^{(i)}\Big)^2\to\min. \quad (2)$$

Подмножество регрессоров $\hat{I}\subseteq I$ и коэффициент $\rho>0$ являются параметрами критерия. Здесь первое слагаемое отвечает за аппроксимацию наблюдений, а второе слагаемое регулирует изменчивость искомых коэффициентов регрессии во времени. Чем больше ρ, тем более плавной будет последовательность оценок, что уменьшает фактическую «размерность» задачи, делая ее промежуточной между $|\hat{I}|$ и $N|\hat{I}|$. При $\rho\to\infty$ критерий сводится к обычному методу наименьших квадратов $\hat{\beta}_1^{(i)}=\ldots=\hat{\beta}_N^{(i)}$.

Если рассматривать априорную модель последовательности коэффициентов регрессии как совокупность независимых скрытых случайных процессов $\beta_t^{(i)}=\beta_{t-1}^{(i)}+\xi_t^{(i)}$, каждый из которых порождается нормальным белым шумом $\xi_t^{(i)}$, дисперсия которого в ρ раз меньше дисперсии шума e_t в модели наблюдения (1), то критерий FLS (2) максимизирует апостериорную плотность распределения вероятности на множестве реализаций скрытого случайного процесса.

Для выбранного множества регрессоров $\hat{I}$ и фиксированного значения коэффициента сглаживания ρ минимизация квадратичной функции (2) сводится к решению, вообще говоря, очень большой системы линейных уравнений относительно $N|\hat{I}|$ переменных, имеющей однако блочно-трехдиагональную матрицу с блоками $|\hat{I}|\times|\hat{I}|$.

Эта особенность допускает применение метода прогонки, обеспечивающего линейную вычислительную сложность решения системы относительно длины временного ряда N. Методу прогонки эквиваленты квадратичное динамическое программирование [3, 4] и фильтр-интерполятор Калмана-Бьюси [5]. Единственным проблематичным аспектом этой процедуры является выбор параметра ρ, который характеризует величину шума в модели наблюдения. Невозможно угадать значение этого параметра априори. В наших предыдущих работах [3, 4] предлагалось выбирать значение этого параметра с помощью процедуры скользящего контроля. Однако, как правило по физической природе явления нестационарными являются коэффициенты лишь при некоторых регрессорах, что порождает также проблему разделения множества регрессоров на стационарные и нестационарные. Кроме того, степени нестационарности каждого регрессора могут существенно различаться. Наконец, эффективное подмножество регрессоров $\hat{I} \subset I$ выбрать априори невозможно, и уже хотя бы поэтому задачу нестационарного регрессионного анализа следует рассматривать как задачу интеллектуального анализа данных. Эти предположения добавляют в критерий огромное количество дополнительных степеней свободы, которые неизбежно приводят к проблеме переобучения. Применять для подбора большого числа параметров метод скользящего контроля оказывается через утомительно, так как это предполагает необходимость минимизации критерия (2) для каждого набора параметров. Таким образом, возникает насущная необходимость в разработке процедуры, которая оценивала бы не только последовательность коэффициентов регрессии, но и автоматически выбирала подмножество действительно присутствующих в модели регрессоров, определяла среди них нестационарные и подбирала для них степень нестационарности

2. Адаптивный Flexible Least Squares критерий для задачи нестационарной регрессии

В этой работе мы предлагаем алгоритм, который способен решать все задачи, поставленные в предыдущей главе, названный нами адаптивный Flexible Least Squares. Пусть $(\mathbf{x}_t,\ t = 1, ..., N)$, $\mathbf{x}_t = (x_t^{(i)},\ i \in I)$ - последовательность регрессоров, вероятностные свойства которой не изучаются. Рассмотрим анализируемую временную последовательность $(y_t,\ t = 1, ..., N)$ (1) как наблюдаемую часть двухкомпонентного случайного процесса, чьей скрытой частью является неизвестная последовательность коэффициентов регрессии $\left(\beta_t = (\beta_t^{(i)}, i \in I),\ t = 0, 1, ..., N\right)$. Главным аспектом предлагаемой здесь технологии отбора регрессоров является априорная вероятностная модель скрытого процесса коэффициентов регрессии $\beta_t = (\beta_t^{(i)},\ i = 1, ..., n)$. Во-первых, рассмотрим априорную модель последовательности коэффициентов регрессии как совокупность независимых скрытых случайных процессов. Во-вторых, предположим, что каждое последующее значение коэффициентов регрессии формируется как результат авторегрессионного процесса

$$\beta_t^{(i)} = \frac{\delta^{(i)}}{\delta^{(i)} + \lambda^{(i)}} \beta_{t-1}^{(i)} + \xi_t^{(i)}, \tag{3}$$

где $\xi_t^{(i)}$ - нормальный белый шум с нулевым математическим ожиданием $E(\xi_t^{(i)}) = 0$ и дисперсией

$$E\left((\xi_t^{(i)})^2\right) = \frac{\delta^{(i)}\lambda^{(i)}}{\delta^{(i)} + \lambda^{(i)}}.$$

Вспомогательные переменные $\delta^{(i)} \geqslant 0$ и $\lambda^{(i)} \geqslant 0$ выполняют функцию регуляризации. Если $\lambda^{(i)} \to 0$, то $E\left(\xi_t^{(i)}\right)^2 \to 0$, и последовательность коэффициентов при i-м регрессоре всегда будет оставаться постоянной (рисунок 1) $\beta_t^{(i)} = \beta_{t-1}^{(i)}$ с некоторым априори неизвестным значением. Если же $\delta^{(i)} \to 0$, то $E\left(\xi_t^{(i)}\right)^2 \to 0$ вместе с $\frac{\delta^{(i)}}{\delta^{(i)}+\lambda^{(i)}} \to 0$, и i-я последовательность коэффициентов превращается в нулевую константу. Совокупность ненулевых переменных $\delta^{(i)}$ образует множество активных регрессоров $\hat{I} = \{i : \delta^{(i)} > 0\} \subseteq I$, а ненулевые переменные $\lambda^{(i)}$ выделяют среди них подмножество регрессоров с нестационарными коэффициентами $\hat{I}_{\text{var}} = \{i : \lambda^{(i)} > 0\} \subseteq \hat{I}$. Произведение дисперсий возмущающего шума $E\left(\xi_t^{(i)}\right)^2$ по всем регрессорам опреде-

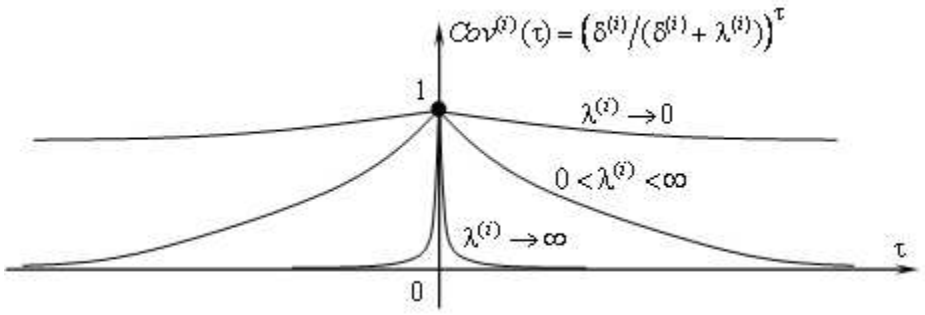


Рис. 1. Изменение характера корреляционной функции $Cov^{(i)}(\tau) = E\left(\beta_t^{(i)}\beta_{t-\tau}^{(i)}\right)/E\left((\beta_t^{(i)})^2\right)$ коэффициентов регрессии при варьировании параметра $\lambda^{(i)}$

ляет объем эллипсоида рассеяния случайного вектора $(\beta_t^{(i)}, i \in I)$ вокруг его условного математического ожидания (3) — чем меньше этот объем, тем интенсивнее подавляется общее отклонение всех коэффициентов регрессии как друг от друга во времени, так и от нуля. Роль переменных $\delta^{(i)}$ и $\lambda^{(i)}$ заключается в управлении соотношением между уровнями вариабельности коэффициентов при разных регрессорах, а не их общей вариабельностью в нестационарной модели. Этот факт приводит к необходимости фиксирования общего объема эллипсоида рассеяния равенством $\prod_{i\in I} \frac{\delta^{(i)}\lambda^{(i)}}{\delta^{(i)}+\lambda^{(i)}} = 1$. Степень общей вариабельности модели определяется дисперсией шума наблюдения в модели нестационарной регрессии (1) $E(e_t) = 0$, $E\left((e_t)^2\right) = \rho$.

Таким образом, мы определили, во-первых, условную априорную плотность распределения скрытой последовательности коэффициентов регрессии $\Psi(\beta_0, \boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_N \mid \delta^{(i)}, \lambda^{(i)}, i \in I)$, и, во-вторых, условную плотность распределения

наблюдаемой переменной $\Phi(y_1,...,y_N\,|\boldsymbol{\beta}_1,...,\boldsymbol{\beta}_N,\rho)$. Очевидно, что совместная плотность распределения скрытой последовательности коэффициентов регрессии и наблюдаемой переменной пропорциональна произведению

$$P(\boldsymbol{\beta}_1,...,\boldsymbol{\beta}_N,\delta^{(i)},\lambda^{(i)},i\in I|y_1,...,y_N,\rho)\propto$$
$$\Phi(y_1,...,y_N|\boldsymbol{\beta}_1,...,\boldsymbol{\beta}_N,\rho)\Psi(\boldsymbol{\beta}_1,...,\boldsymbol{\beta}_N|\delta^{(i)},\lambda^{(i)},i\in I)$$

Кажется совершенно естественным выбрать в качестве оценки последовательности коэффициентов регрессии максимальную точку этой апостериорной плотности

$$(\hat{\boldsymbol{\beta}}_1,...,\hat{\boldsymbol{\beta}}_N,\hat{\delta}^{(i)},\hat{\lambda}^{(i)},i\in I|y_1,...,y_N,\rho)=$$
$$=argmaxP(\boldsymbol{\beta}_1,...,\boldsymbol{\beta}_N,\delta^{(i)},\lambda^{(i)},i\in I|y_1,...,y_N,\rho) \quad (4)$$

Можно легко показать, что максимальная точка апостериорной плотности (4) есть минимальная точка критерия

$$J(\beta_t^{(i)},t=1,...,N,\delta_i,\lambda_i,i\in I\,|\,\rho)=\sum_{i=1}^{N}\left(y_t-\sum_{i\in I}\beta_t^{(i)}x_t^{(i)}\right)^2+$$
$$+\rho\sum_{t=2}^{N}\sum_{i\in I}\frac{\delta^{(i)}+\lambda^{(i)}}{\delta^{(i)}\lambda^{(i)}}\left(\beta_t^{(i)}-\frac{\delta^{(i)}}{\delta^{(i)}+\lambda^{(i)}}\beta_{t-1}^{(i)}\right)^2\to\min,$$
$$\prod_{i\in I}\frac{\delta^{(i)}\lambda^{(i)}}{\delta^{(i)}+\lambda^{(i)}}=1,\ \text{т. е. }\sum_{i\in I}\log\frac{\delta^{(i)}\lambda^{(i)}}{\delta^{(i)}+\lambda^{(i)}}=0. \quad (5)$$

В отличие от (2) адаптивный критерий применяется ко всему множеству регрессоров I.

Легко видеть, что если параметры $(\delta^{(i)},\lambda^{(i)},i\in I)$ фиксированы, результирующий критерий практически совпадает с FLS критерием (2). Однако, присутствие дополнительных переменных $(\delta^{(i)},\lambda^{(i)},i\in I)$ в модели чрезвычайно важно. Если некоторый коэффициент $\delta^{(i)}\to 0$, критерий жестко штрафует отличие соответствующей последовательности коэффициентов регрессии $(\beta_0^{(i)},\beta_1^{(i)},...,\beta_N^{(i)})$ от нуля и, таким образом, практически исключает соответствующий регрессор из модели. Если какой-то коэффициент $\lambda^{(i)}\to 0$, соседние значения скрытого процесса практически совпадают и i-й коэффициент регрессии будет практически постоянным во времени.

3. Итерационная процедура оптимизации адаптивного критерия

Для последовательности наблюдений критерий (1) и фиксированного параметра ρ адаптивный критерий (5) может быть легко минимизирован с помощью применения итерационного метода Гаусса-Зайделя к двум группам переменных $(\delta^{(i)},\lambda^{(i)},\ i\in I)$ и $(\beta_t^{(i)},\ i\in I,\ t=1,\ldots,N)$, начиная с некоторых значений $\delta^{(i),0}$ и $\lambda^{(i),0}$, удовлетворяющих ограничению в (2).

Заметим, что при фиксированных значениях $\delta^{(i)}$ и $\lambda^{(i)}$, в частности, $\delta^{(i),k}$ и $\lambda^{(i),k}$ на k-й итерации, критерий является квадратичной функцией блочно-трехдиагональной структуры относительно последовательности векторных переменных $(\beta_t^{(i)},\ i \in I),\ t = 1, \ldots, N$, и его минимизация осуществляется эквивалентными методами прогонки, квадратичного динамического программирования, либо фильтрации-интерполяции Калмана-Бьюси [2, 3, 4, 5] за время, пропорциональное длине временного ряда N. Нетрудно доказать, что после того, как последовательность $(\beta_t^{(i),k},\ i \in I,\ t = 1, \ldots, N)$ найдена, очередные значения $\delta^{(i),k+1}$ и $\lambda^{(i),k+1}$, минимизирующие (5), вычисляются по формулам, в которых $a^{(i)} = \frac{\delta^{(i)}}{\delta^{(i)}+\lambda^{(i)}}$ и $0 \lessapprox a_0 < a_1 \lessapprox 1$:

$$0 < a^{(i),k+1} = \left(a_0, \frac{\sum_{t=2}^{N} \beta_{t-1}^{(i),k} \beta_t^{(i),k}}{\sum_{t=2}^{N} \left(\beta_{t-1}^{(i),k}\right)^2}, a_1 \right) < 1;$$

$$\delta^{(i),k+1} = \frac{a^{(i),k+1}}{1 - a^{(i),k+1}} \lambda^{(i),k+1};$$

$$\lambda^{(i),k+1} = \frac{1}{a^{(i),k+1}} \frac{\sum_{t=1}^{N} \left(\beta_t^{(i),k} - a^{(i),k+1} \beta_{t-1}^{(i),k}\right)^2}{\left[\prod_{j \in I} \sum_{t=1}^{N} \left(\beta_t^{(j),k} - a^{(j),k+1} \beta_{t-1}^{(j),k}\right)^2\right]^{1/|I|}}, \quad i \in I.$$

Итерационный процесс обычно сходится за 10–15 итераций, проявляя явную тенденцию к практическому обнулению вспомогательных переменных, отвечающих за подавление части регрессоров $\delta^{(i),k} \to \hat{\delta}^{(i)} \gtrapprox 0$, при этом переменные, ответственные за нестационарность коэффициентов регрессии, также почти обнуляются $\lambda^{(i),k} \to \hat{\lambda}^{(i)} \gtrapprox 0$. Предельные значения, оставшиеся существенно ненулевыми $\delta^{(i),k} \to \hat{\delta}^{(i)} > 0$, выделяют подмножество эффективных регрессоров $\hat{I} \subset I$. Однако переменные $\lambda^{(i),k}$ стремятся к нулю, как правило, для большего числа регрессоров, подавляя нестационарность коэффициентов регрессии для части регрессоров, выделенных как эффективные $i \in \hat{I}$. Остальные предельные значения $\hat{\lambda}^{(i)} > 0$ указывают подмножество эффективных регрессоров с нестационарными коэффициентами $\hat{I}_{\text{var}} \subseteq \hat{I}$.

Значение параметра ρ не может быть определено путем дополнительной оптимизации критерия (2) и подбирается с помощью процедуры скользящего контроля, особенности использования которой в задаче оценивания нестационарной регрессии рассмотрены в работе [4].

4. Экспериментальное исследование Adaptive Flexible Least Squares: модельный пример

Эффект выделения регрессоров, адекватных анализируемому временному ряду, и среди них регрессоров, входящих в модель с нестационарными коэффициентами, иллюстрирует следующий модельный эксперимент.

Модельный временной ряд $(y_t, x_t^{(i)}, i \in I)$ вида (1) построен как последовательность ста зашумленных линейных комбинаций, $t = 1, \ldots, 100$, ста регрессоров

$I = \{1, \ldots, 100\}$, в качестве которых использовались независимые реализации нормального белого шума. Среди ста последовательностей коэффициентов регрессии две приняты отличными от нуля, одна из которых образована одним периодом синусоиды $x_t^{(1)} = \sin\left(\frac{2\pi t}{100}\right)$, а вторая является единичной константой $x_t^{(2)} \equiv 1$, остальные же коэффициенты регрессии тождественно равны нулю $x_t^{(3)} = \ldots = x_t^{(100)} \equiv 0$.

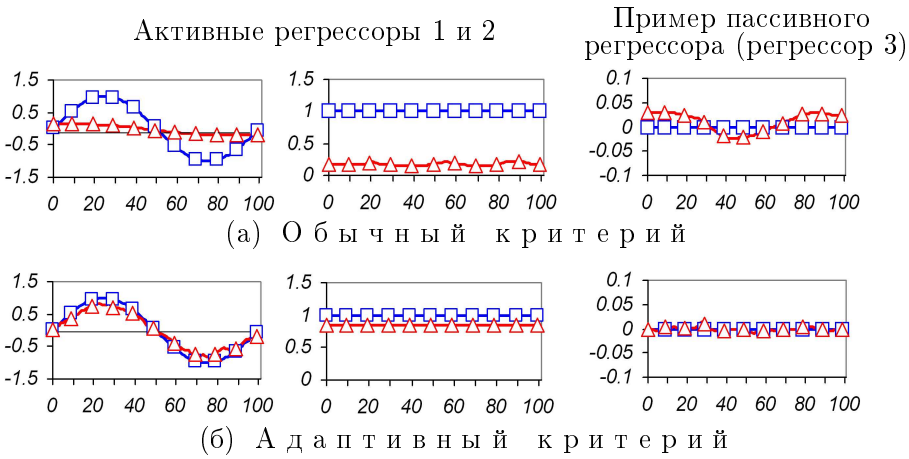


Рис. 2. Результаты экспериментов по оцениванию нестационарной регрессии: □ — модельные последовательности коэффициентов, △ — оцененные последовательности.

Таким образом, если неизвестно, какие именно регрессоры являются активными, то оцениванию подлежат десять тысяч коэффициентов регрессии при наличии всего лишь ста наблюдений. Естественно, что обычный критерий FLS (2), хотя в нем параметр сглаживания ρ подбирался процедурой скользящего контроля, оказался не способен в этом примере даже приближенно восстановить модель нестационарной регрессии, «размывая» вклад двух активных регрессоров на все сто регрессоров, что хорошо видно на Рис. 2(а). В то же время адаптивный критерий (**??**), как показывает Рис. 2(б), практически полностью подавляет пассивные регрессоры. Вся нестационарность модели оказывается сконцентрированной в изменении коэффициента только при первом регрессоре, а коэффициент при втором активном регрессоре идентифицирован как стационарный.

5. Экспериментальное исследование Adaptive Flexible Least Squares: восстановление структуры инвестиционного портфеля

В этом разделе мы рассмотрим задачу слежения за составом инвестиционного портфеля и анализа стратегии инвестиционной компании. Инвестиционная компания - это тип финансового посредника. Они привлекают средства инвесторов и приобретают на них финансовые активы, такие, как например акции, облигации и другие ценные бумаги. Совокупность финансовых активов, которыми владеет инвестиционная компания называется ее портфелем или портфолио.

Целью деятельности любой инвестиционной компании является увеличение стоимости ее портфеля. Вообще говоря, финансовая инвестиционная компания не обязана информировать общественность, и даже своих акционеров, о составе своего портфеля, считая его (состав) своей профессиональной тайной. Естественно, что и собственные акционеры, и инвестиционные компании-конкуренты, отдали бы много за то, чтобы обладать информацией о составе портфеля. Единственной информацией о деятельности инвестиционной компании, к которой открыт свободный доступ, является индекс ее доходности в каждый момент времени. Кроме того, известны котировки всех акций на фондовом рынке, где ведет свою деятельность данная инвестиционная компания, независимо от того. Задачей нашего исследования будет восстановление процентного состава портфеля инвестиционной компании в каждый момент времени по известным значениям уровня доходности инвестиционной компании и котировкам акций на фондовом рынке. Можно показать [6], что если за рассматриваемый период никакие средства не поступали в портфель извне и не изымались из него, то доходность портфеля определяется как линейная комбинация доходностей составляющих его ценных бумаг. Таким образом, мы приходим к задаче нестационарной регрессии.

Как правило, множество активов, в которые действительно вложен капитал инвестиционной компании много меньше, чем множество всех возможных активов. Таким образом, очень важно определить подмножество ненулевых регрессоров. Кроме того управление портфелем зачастую осуществляется посредством торговли только несколькими активами, в то время как остальные активы остаются незадействованными в управлении портфелем. Таким образом, остается актуальной задача поиска среди всех регрессоров действительно меняющихся.

В данном разделе мы рассмотрим пример использования предложенной методологии для анализа инвестиционной политики одного из самых известных хедж фондов Long Term Capital Management (LTCM), кризис которого в 1998 году остается уже на протяжении 10 лет одной из самых драматичных страниц истории мировых финансовых рынков. Проблемы фонда начались в мае-июне 1998 года [7], а к концу сентября, через месяц после кризиса рынка российских государственных облигаций (ГКО), фонд потерял более 92 процентов своего капитала. В данном разделе мы попытаемся, используя адаптивный нестационарный регрессионный анализ, определить факторы, объясняющие падение LTCM. Для нашего анализа мы используем доходности классов ценных бумаг, в которые мог быть инвестирован капитал фонда. Данные доходностей были предоставлены Lehman Brothers и Merrill Lynch.

Мы варьировали параметр нестационарности ρ в интервале от минимального значения $\rho = 10^{-8} \cong 0$, которое соответствует практической независимости мгновенных моделей, до максимального значения, обеспечивающего их полную стационарность. Для каждого значения ρ мы применяли к данным временным рядам итерационную процедуру, предложенную в разделе 3. Используя процедуру скользящего контроля для нахождения оптимального значения параметра нестационарности, мы получили $\rho = 10$.

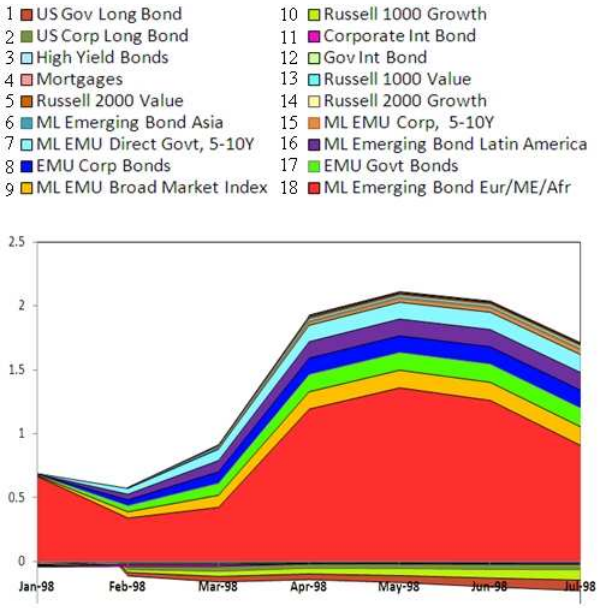


Рис. 3. Адаптивный нестационарный регрессионный анализ инвестиционной стратегии фонда Long Term Capital Management

Соответствующие результат представлен на рисунке 3 в виде стековой диаграммы, на которой долевые коэффициенты активов выстроены вдоль вертикальной оси с учетом знака. Отрицательные позиции соответствуют заемным или хеджевым позициям. Примечательным оказывается тот факт, что предложенный адаптивный алгоритм позволяет подавить несущественные активы и, как можно видеть из рисунка 3, только 8 из 18 классов активов присутствуют в окончательной модели. Другим важным аспектом является то, что 7 из 8 регрессионных коэффициентов, в качестве которых в данной задаче выступают доли классов активов, были оценены как практически постоянные. Актив, доля которого была оценена как меняющаяся, представляет собой вложения фонда на развивающихся финансовых рынках, включая российский рынок краткосрочных государственных облигаций. Таким образом, наш анализ подтверждает гипотезу о том, что причиной краха LTCM стал дефолт по российским государственным краткосрочным облигациям.

Заключение

В данной работе были предложены способы регуляризации задачи оценивания нестационарной регрессии, позволяющие наряду с оцениванием регрессионных коэффициентов во-первых, автоматически выбирать подмножество эффективных регрессоров, во-вторых, автоматически отбирать еще меньшее подмножество нестационарных регрессоров, и, в-третьих, определять индивидуальные коэффициенты сглаживания каждого коэффициента регрессии в отдельности. Эффективность предложенного подхода подтверждается результатами на модельных и прикладных примерах.

Работа выполнена при поддержки РФФИ, проект № 06-07-89249, 06-01-00412, 08-01-00695, 08-01-99003.

Список литературы

1. *Jain A., Zongker D* Feature selection: Evaluation, application, and small sample performance. // IEEE Trans. on Pattern Analysis and Machine Intelligence, February 1997, Vol. 19, no. 2, pp. 153-158.,
2. *R. Kalaba, L. Tesfatsion* Time-varying linear regression via flexible least squares. // International Journal on Computers and Mathematics with Applications, 1989, Vol. 17, pp. 1215-1245.
3. *Костин А. А., Красоткина О. В., Марков М. Р., Моттль В. В., Мучник И. Б.* Алгоритмы динамического программирования для анализа нестационарных сигналов. — ЖВМиМФ, 2004, — Т. 44, № 1. — С. 70–86.
4. *Markov M., Krasotkina O., Mottl V., Muchnik I.* Time-varying regression model with unknown time-volatility for nonstationary signal analysis // 8th IASTED Int. Conf. on Signal and Image Processing, 2006, Honolulu, USA. — Pp. 14–16.
5. *Wells C.* The Kalman Filter in Finance. — Kluwer Academic Publishers, 1996.
6. *Sharpe W.F.* Asset allocation: Management style and performance measurement. The Journal of Portfolio Management, Winter 1992, pp. 7-19.
7. *Ph. Jorion* Risk management lessons from long-term capital management. European Financial Management, September, 2000.

Статья поступила в редакцию 04.05.2008

Borisov A.E., Tuv E.V. ***Zero-Inflated boosted ensemble for small count problem*** // Таврический вестник информатики и математики. – 2008. – №1. – С. 5-15.

УДК 519.23

Стаття описує новий підхід до навчання моделі даних, що описують рідкісні події, у випадку, коли цільова змінна (кількість подій) може бути описана розподілом з доданими нулями (ZIP-розподілом). Запропонована ZIP-модель, що базується на ансамблі дерев, які побудовані за допомогою бустингу. Ця модель основана на комбінації ідей ZIP-дерева та GBT-ансамблю. Наш алгоритм, який названо ZIP-GBT, спочатку виводиться теоретично в межах підходу градієнтного бустингу, запропонованого Фрідманом. Далі наш метод порівнюється емпірично на двох реальних наборах даних і на двох синтезованих. Показано, що ZIP-GBT в більшості випадків переважає ZIP-дерево в термінах обрахованого за допомогою кросівалідації ZIP-правдоподібності та помилки передбачення параметрів ZIP-розподілу.

В статье описан новый подход к обучению модели данных, описывающих редкие события, когда целевая переменная (число событий) может быть описана распределением с добавленными нулями (ZIP-распределением). Предложена ZIP-модель, основанная на ансамбле деревьев, построенном с помощью бустинга. Эта модель основана на комбинации идей ZIP-дерева и GBT-ансамбля. Наш алгоритм, названный ZIP-GBT, сначала выводится теоретически в рамках подхода градиентного бустинга, предложенного Дж.Фридманом. Затем наш метод сравнивается эмпирически на двух реальных наборах данных и на двух синтезированных. Показано, что ZIP-GBT в большинстве случаев превосходит ZIP-дерево в терминах подсчитанного с помощью кросс-валидации ZIP-правдоподобия и ошибки предсказания параметров ZIP-распределения.

Martyanov V.Ju., Eruhimov V.L. ***Time series classification through heterogeneous feature selection*** // Таврический вестник информатики и математики. – 2008. – №1. – С. 16-24.

УДК 681.3: 519.68

У роботі пропонується та досліджується загальний метод розв'язання задачі класифікації часових рядів. По часових рядах обчислюються кілька наборів ознак, таких як статистичні моменти, коефіцієнти розкладання по принципових компонентах, коефіцієнти розкладання по вейвлетам, коефіцієнти розкладання по поліномах Чєбишева та інші, далі із цієї множини алгоритмом Particle Filtering вибираються важливі з погляду задачі ознаки та на них будується класифікаційна модель на

основі ансамблю дерев рішень GBT. Результати, отримані на привселюдно доступних даних з бази UCR, показують, що запропонований підхід у багатьох випадках дозволяє досягти більш точної класифікації, ніж відомі сучасні методи.

В работе предлагается и исследуется общий метод решения задачи классификации временных рядов. По временным рядам вычисляются несколько наборов признаков, таких как статистические моменты, коэффициенты разложения по принципиальным компонентам, коэффициенты разложения по вейвлетам, коэффициенты разложения по полиномам Чебышева и другие, далее из этого множества алгоритмом Particle Filtering выбираются важные с точки зрения задачи признаки и на них строится классификационная модель на основе ансамбля деревьев решений GBT. Результаты, полученные на публично доступных данных из базы UCR, показывают, что предложенный подход во многих случаях позволяет достичь более точной классификации, чем известные современные методы.

Treebushny D., Kotkov V., Chikalov I. ***'Split and peel' rule induction method*** // Таврический вестник информатики и математики. – 2008. – №1. – С. 25-31.

УДК 519.23

Алогиртм [2] був розроблений для побудови асоціативних правил, що описують області у просторі вхідних даних, в яких середнє значення цільової змінної значно вище за середнє значення цієї ж змінної на всьому вхідному просторі. Незважаючи на успішне застосування PRIM у різноманітних задачах, якість розв'язку може погіршуватись при роботі з виборками мультимодального розподілу ймовірностей. У поданій роботі пропонується модифікація вихідного алгоритму: процедура розділення мод, що замінює процедуру уточнення для обробки мультимодальної підвибірки. Наведено порівняння результатів роботи вихідного та модифікованого алгоритмів при аналізі штучного набору даних, що імітують задачу пошуку несправностей.

Алгоритм PRIM [2] предназначен для построения ассоциативных решающих правил, описывающих области в пространстве входных данных, в которых среднее значение целевой переменной значительно выше, чем среднее значение целевой переменной на всем входном пространстве. Несмотря на успешное применение PRIM в различных задачах, качество решения может ухудшаться при работе с выборками из мультимодальных распределений вероятностей. В данной работе предложена модификация исходного алгоритма: процедура разделения мод, которая заменяет процедуру уточнения для обработки мультимодальной подвыборки. Приведено сравнение результатов работы исходного и модифицированного алгоритма при анализе искуственного набора данных, имитирующего задачу поиска неисправностей.

Абламейко С.В., Крючков А.Н., Соболь Л.Н., Апарин Г.П. ***Технология выявления изменений и обновления цифровых карт городского кадастра на основе космических снимков высокого разрешения*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 32-37.

УДК 681.3:51

Розглядається технологія обробки даних дистанційного зондування Землі (ДЗЗ) для вирішення завдань автоматизованого виявлення змін в міській забудові з метою оновлення цифрових карт місцевості міського кадастру на основі космічних знімків високого дозволу. Приводяться приклади використання розробленої технології.

Рассматривается технология обработки данных дистанционного зондирования Земли (ДЗЗ) для решения задач автоматизированного обнаружения изменений в городской застройке с целью обновления цифровых карт местности городского кадастра на основе космических снимков высокого разрешения. Приводятся примеры использования разработанной технологии.

Акимов О.М., Шапцев В.А. ***Интеллектуализация пользовательского интерфейса базы данных*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 38-43.

УДК 004.5+004.657

Розглядається задача перекладу запита, сформульованому природною мовою, у стандартний запит мовою СУБД. У статті описується, як це можна зробити на основі концептуальних графів. Інтерфейс аналізує й інтерпретує ПМ-запрос і представляє його у вигляді концептуального графа. Потім цей граф модифікується інформацією з бази знань про предметну галузь і структуру БД. Результуючий граф стає основою формування SQL-запиту.

Рассматривается задача перевода запроса, сформулированного на естественном языке, в стандартный запрос на языке СУБД. В статье описывается, как это можно выполнить на основе концептуальных графов. Интерфейс анализирует и интерпретирует ЕЯ-запрос и представляет его в виде концептуального графа. Затем этот граф модифицируется информацией из базы знаний о предметной области и структуре БД. Результирующий граф становится основой формирования SQL-запроса.

Амиргалиев Е.Н., Амиргалиева С.Н. ***Методы анализа и синтеза информационной системы распознавания образов*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 44-50.

УДК 5019.7

У даній роботі розглядається розробка концептуальної моделі інформаційної системи розпізнавання і класифікації на основі математичних методів і моделей розпізнавання образів і класифікації, призначена для аналізу і обробки багатовимірних даних. Методологія розробки інформаційної системи реалізована з використанням об'єктно-орієнтованого підходу і здійснена на мові моделювання складних програмних систем – уніфікованої мови моделювання UML. Розглянута концепція побудови інформаційної системи може бути використана в разних наочних областях, як сучасна концепція розробки подібних інформаційних систем. Для реалізації проектних рішень використано CASE-средство Rational Rose.

В данной работе рассматривается разработка концептуальной модели информационной системы распознавания и классификации на основе математических методов и моделей распознавания образов и классификации, предназначенная для анализа и обработки многомерных данных. Методология разработки информационной системы реализована с использованием объектно-ориентированного подхода и осуществлена на языке моделирования сложных программных систем – унифицированного языка моделирования UML. Рассмотренная концепция построения информационной системы может быть использована в разных предметных областях, как современная концепция разработки подобных информационных систем. Для реализации проектных решений использовано CASE-средство Rational Rose.

Амиргалиев Е.Н., Мусабаев Р.Р. ***Методы анализа и проектирования системы синтеза искусственной речи*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 51-58.

УДК 5019.7

У роботі розглядається практичний досвід в галузі реалізації й застосування алгоритмів синтезу мови. Розкрите питання ефективності застосування мовних людино- машинних інтерфейсів при побудові лінгвістичних навчальних систем.

В работе рассматривается практический опыт в области реализации и применения алгоритмов синтеза речи. Раскрыт вопрос эффективности применения речевых человеко-машинных интерфейсов при построении лингвистических обучающих систем.

Бакина И.Г., Местецкий Л.М. ***Многомодальная идентификация личности по форме ладони и голосу*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 59-65.

УДК 004.93

У роботі показаний спосіб поліпшення якості розпізнавання одномодального методу ідентифікації людини за формою долоні на основі комбінування його з менш надійним методом розпізнавання (по вимовленому слову). Приводяться теоретичні і експериментальні оцінки, підтверджуючі ефективність запропонованого підходу.

В работе показан способ улучшения качества распознавания одномодального метода идентификации человека по форме ладони на основе комбинирования его с менее надежным методом распознавания (по произнесенному слову). Приводятся теоретические и экспериментальные оценки, подтверждающие эффективность предложенного подхода.

Бауман Е.В., Дорофеюк А.А., Дорофеюк Ю.А., Киселёва Н.Е. ***Программно-алгоритмический комплекс структурно-классификационного анализа сложноорганизованных данных*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 66-72.

УДК 62-50

У роботе розглянута концепція застосування методів структурно-класифікаційного аналізу,розглянута методів**:** структуризації параметров; класифікації об'єктів; динамічного класификаційного аналізу; шматковій аппроксимації складних залежностей. Концепція реалізована в людино-машинній системі з інтелектуальним інтерфейсом для користувача. Під час розробки комплекса особлива увага приділялась задачам, в яких исследуемые объекты об'єкти мають явно виражену територіальну структуру.

В работе рассмотрена концепция применения методов структурно-классификационного анализа, в том числе методов**:** структуризации параметров; классификации объектов; динамического классификационного анализа; кусочной аппроксимации сложных зависимостей. Концепция реализована в человеко-машинной системе с интеллектуальным интерфейсом для пользователя. При разработке комплекса особое внимание уделялось задачам, в которых исследуемые объекты имеют явно выраженную территориальную структуру.

Бауман Е.В., Гольдовська Р.М., Дорофеюк Ю.А. ***Методы кусочно-лінійній апроксимації і їх використання в завданнях управління*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 73-79.

УДК 62-50

Робота присвячена методам рішення задачі кусочно-лінійної апроксимації розроблених на базі загального підходу до завдань класифікаційного аналізу даних. Основна ідея шматкової апроксимації складної залежності полягає в розбитті простору вхідних параметрів на такі області, в межах кожній з яких складну у всьому

просторі залежність можна апроксимувати лінійною функцією.

Работа посвящена методам решения задачи кусочно-линейной аппроксимации, разработанных на базе общего подхода к задачам классификационного анализа данных. Основная идея кусочной аппроксимации сложной зависимости состоит в разбиении пространства входных параметров на такие области, в пределах каждой из которых сложную во всем пространстве зависимость можно аппроксимировать линейной функцией.

Бериков В.Б. ***Оценки риска в байесовской модели распознавания порядковой переменной по конечному множеству событий*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 80-88.

УДК 519.6

У роботі розглядається завдання порядкової регресії, при рішенні якої використовується клас логічних вирішальних функцій. Пропонується байесовская модель розпізнавання по кінцевій безлічі подій, яка застосовується для знаходження оптимальної складності класу. Приводяться оцінки ризику, отримані відповідно до байесовской моделлю.

В работе рассматривается задача порядковой регрессии, при решении которой используется класс логических решающих функций. Предлагается байесовская модель распознавания порядковой переменной по конечному множеству событий, которая применяется для нахождения оптимальной сложности класса. Приводятся оценки риска, полученные в соответствии с байесовской моделью.

Богуш А.Л., Ковалев В.А. ***Текстурный анализ ультразвуковых изображений щитовидной железы*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 89-96.

УДК 004.9

У даній роботі вивчається ефективність методів аналізу текстур ультразвукових зображень. У результаті проведених досліджень було встановлено, що характеристики, яки встановлюють властивості анізотропії зображень щитовидної залози, містять інформацію, найбільш корисну для цілей класифікації. Найкращі результати попарної класифікації пухлин трьох типів, а також класифікація доброякісних (зоб й аденома) і злоякісних пухлин склали близько 80%.

В данной работе изучается эффективность методов анализа текстур ультразвуковых изображений. В результате проведенных исследований было установлено, что характеристики, представляющие свойства анизотропии изображений щитовидной

железы, содержат информацию, наиболее полезную для целей классификации. Наилучшие результаты попарной классификации опухолей трех типов, а также классификация доброкачественных (зоб и аденома) и злокачественных опухолей составили около 80%.

Брусенцов Н.П. ***Адекватность интеллекта и гераклитово сосуществование противоположностей*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 97-100.

УДК 510.6

Першооснова адекватного інтелекту, як природного, так штучного, складається в гераклітовом принципові співіснування протилежностей. Недотримання цього фундаментального принципу руйнує розсудливу логіку.

Первооснова адекватного интеллекта, как естественного, так искусственного, состоит в гераклитовом принципе сосуществования противоположностей. Несоблюдение этого фундаментального принципа разрушает благоразумную логику.

Викентьев А.А., Викентьев Р.А. ***Меры опровержимости и расстояния на многозначных экспертных высказываниях в адаптивных методах построения логических решающих функций*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 101-106.

УДК 510.67-519.24

У даній роботі запропоновано записувати висловлення експертів у вигляді формул багатозначної логіки Лукасевича та запропоновано спосіб введення відстані та міри спростовності; доведені їхні властивості. Запропонований підхід розширює та узагальнює вивчені раніше випадки $n = 2$ та $n = 3$ і також потрібний для розвитку адаптивних методів побудови логічних вирішуючих функцій по імовірнісних висловленнях.

В данной работе предложено записывать высказывания экспертов в виде формул многозначной логики Лукасевича и дан способ введения расстояния и меры опровержимости; доказаны их свойства. Предлагаемый подход расширяет и обобщает изученные ранее случаи n=2 и n=3 и нужен для развития адаптивных методов построения логических решающих функций по вероятностным высказываниям.

Вятченин Д.А. ***Метод мягкой интерпретации результатов нечеткой кластеризации*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 107-114.

УДК 519.237.8+510.22

У статті запропоновано новий метод інтерпретації результатів нечіткої кластеризації, який дозволяє здійснити автоматичний вибір порогу, по якому виробляється виділення множин найбільш інформативних елементів нечітких класів. Ефективність нового методу ілюструється на прикладі обробки результатів нечіткої кластеризації нечітких даних.

В статье предложен новый метод интерпретации результатов нечеткой кластеризации, который позволяет осуществить автоматический выбор порога, по которому производится выделение множеств наиболее информативных элементов нечетких классов. Эффективность нового метода иллюстрируется на примере обработки результатов нечеткой кластеризации нечетких данных.

Гончаренко В.Г., Архипов В.И., Тузиков А.В. ***Реализация иерархической модели данных в системе компьютерного планирования хирургических операций в ортопедии*** // Таврический вестник информатики и математики. – 2008. – №1. – С. 115-124.

УДК 004.932

У роботі розглядається використання ієрархічної моделі даних, призначеної для опису результатів перетворень різних частин зображення великого розміру. У якості однієї з операцій перетворення бінарного зображення розглядається процедура розрізування об'єкта зображення багатокутником ("різцем"), заданим користувачем. Всі зв'язні компоненти, яки повністю пересічні «різцем», розріжуться на кілька частин.

В работе рассматривается использование иерархической модели данных, предназначенной для описания результатов преобразований различных частей изображения большого размера. В качестве одной из операций преобразования бинарного изображения рассматривается процедура разрезания объекта изображения многоугольником («резцом»), заданным пользователем. Все связные компоненты, полностью пересекаемые «резцом», разрезаются на несколько частей.

Гула О.Ю. ***Питання очистки даних при створенні автоматизованих систем нормативно-довідкової інформації*** // Таврический вестник информатики и математики. – 2008. – №1. – С. 125-134.

УДК 004.622

Робота присвячена методам ідентифікації даних систем нормативно-довідкової інформації. Запропоновано алгоритм об'єднання даних декількох ієрархічних класифікаторів. Запропоновано схему програмної реалізації процесу обробки даних при побудові об'єднаного класифікатора.

Работа посвящена методам идентификации данных систем нормативно-справочной информации. Предложен алгоритм объединения данных нескольких иерархических классификаторов. Предложена схема программной реализации процесса обработки данных при построении объединённого классификатора.

Домахина Л.Г. ***Устойчивость скелетной сегментации*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 135-144.

УДК 004.622

Є достатня кількість робіт, у яких описані методи сегментації фігур, однак найчастіше вони не містять серйозного обґрунтування на користь вибору того або іншого методу сегментації. У даній роботі наводиться формалізоване визначення якості сегментації через поняття стійкісті та обґрунтування стійкості методу кістякової сегментації.

Имеется достаточное количество работ, в которых описаны методы сегментации фигур, однако зачастую они не содержат серьезного обоснования в пользу выбора того или иного метода сегментации. В настоящей работе приводится формализованное определение качества сегментации через понятие устойчивость и обоснование устойчивости метода скелетной сегментации.

Дорофеюк А.А., Гольдовская М.Д., Чернявский А.Л. ***Учёт человеческого фактора в задачах принятия решений для организационных систем управления*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 145-151.

УДК 5:519.876

Розглядаються методи ухвалення рішень за участю експертів, що дозволяють зменшити негативний вплив людського чинника, а іногда – використовувати його для отримання додаткової інформації. Для вирішення подібних завдань пропонується використовувати процедури колективної багатоваріантної експертизи.

Рассматриваются методы принятия решений с участием экспертов, позволяющие уменьшить отрицательное влияние человеческого фактора, а иногда – использовать его для получения дополнительной информации. Для решения подобных задач предлагается использовать процедуры коллективной многовариантной экспертизы.

Дорофеюк А.А., Гучук В.В., Десова А.А., Дорофеюк Ю.А., Покровская И.В. ***Классификационный анализ характеристик пульсового сигнала в задачах диагностики сердечно-сосудистых заболеваний*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 152-158.

УДК 612.821:615.47:616.12-008

Описані результати використання спеціально розробленого комплексу алгоритмів класифікаційного аналізу і розпізнавання образів для діагности серцево-судинних захворювань на прикладі завдання диференціальної діагностики ранньої гіпертензії дітей і підлітків.

Описаны результаты использования специально разработанного комплекса алгоритмов классификационного анализа и распознавания образов для диагностики сердечно-сосудистых заболеваний на примере задачи дифференциальной диагностики ранней гипертензии детей и подростков.

Дорофеюк А.А., Дорофеюк Ю.А., Покровская И.В. ***Методология экспертно-классификационного анализа данных в задачах анализа развития региональных систем*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 159-165.

УДК 62-50

Розглянуті експертно-класифікаційні методи аналізу функціонування регіональних соціально-економічних систем. Як приклад описано використання розробленої методики для порівняльної оцінки суб'єктів Російській Федерації.

Рассмотрены экспертно-классификационные методы анализа функционирования региональных социально-экономических систем. В качестве примера описано использование разработанной методики для сравнительной оценки субъектов Российской Федерации.

Дорофеюк Ю.А. ***Структурно-классификационные методы анализа и прогнозирования в системах управления*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 166-170.

УДК 62-50

Запропонований метод рішення задачі аналізу і прогнозування в слабоформалізованной багатопараметричній системі управління, що складається з достатньо великого числа апріорі не структурованих об'єктів. У якості прогнозній моделі для кожного об'єкту використовується марківський ланцюг з r станами, де r – –число структурних одиниць (класів). Реалізація запропонованого методу базується на алгоритмах класифікаційного аналізу даних.

Предложен метод решения задачи анализа и прогнозирования в слабоформализованной многопараметрической системе управления, состоящей из достаточно большого числа априори не структурированных объектов. В качестве прогнозной модели для

каждого объекта используется марковская цепь с r состояниями, где r — —число структурных единиц (классов). Реализация предложенного метода базируется на алгоритмах классификационного анализа данных.

Дорофеюк Ю.А. ***Комплексный алгоритм автоматической классификации и его использование в задачах анализа и принятия решений*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 171-177.

УДК 62-50

Описаний комплексний алгоритм автоматичної класифікації (кластер-аналіза) який був спеціально розроблений для завдань інтелектуальної обробки складноорганізованих даних і підтримки ухвалення рішень. Він включає алгоритми: m-локальной оптимізації заданого критерію якості класифікації вибору інформативних параметрів, вибору початкового розбиття, вибору числа класів, заповнення пропущених спостережень.

Описан комплексный алгоритм автоматической классификации (кластер-анализа), который был специально разработан для задач интеллектуальной обработки сложноорганизованных данных и поддержки принятия решений. Он включает алгоритмы: m-локальной оптимизации заданного критерия качества классификации, выбора информативных параметров, выбора начального разбиения, выбора числа классов, заполнения пропущенных наблюдений.

Дорофеюк Ю.А. ***Структурно-классификационные методы анализа и прогнозирования в системах управления*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. ??-??.

УДК 519.68

Є достатня кількість робіт, в яких описані методи сегментації фігур, проте часто вони не містять серйозного обгрунтування на користь вибору того або іншого методу сегментації. У справжній роботі приводиться формалізоване визначення якості сегментації через поняття стійкість і обгрунтування стійкості методу скелетної сегментації.

Имеется достаточное количество работ, в которых описаны методы сегментации фигур, однако зачастую они не содержат серьезного обоснования в пользу выбора того или иного метода сегментации. В настоящей работе приводится формализованное определение качества сегментации через понятие устойчивость и обоснование устойчивости метода скелетной сегментации.

Дулькейт В.И., Файзуллин Р.Т., Хныкин И.Г. ***Минимизация функционалов, ассоциированных с задачами криптографического анализа асимметричных шифров*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 178-188.

УДК 519.7

У роботі розглядається модифікація методу логічного кріптоаналізу, заснована на процедурі мінімізації функціоналів, асоційованих із задачами криптографічного аналізу асиметричних шифрів. Запропоновано алгоритми зведення криптографічних алгоритмів RSA, дискретного логарифмування та дискретного логарифмування на еліптичних кривих до задачі ВИКОНУВАНІСТЬ.

В работе рассматривается модификация метода логического криптоанализа [1], основанная на процедуре минимизации функционалов, ассоциированных с задачами криптографического анализа асимметричных шифров. Предложены алгоритмы сведения криптографических алгоритмов RSA, дискретного логарифмирования и дискретного логарифмирования на эллиптических кривых к задаче ВЫПОЛНИМОСТЬ.

Дьяконов А.Г. ***Реализация иерархической модели данных в системе компьютерного планирования хирургических операций в ортопедии*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 199-203.

УДК 519.714

У роботі описуються результати, які отримані в останні роки в рамках алгебраїчного підходу щодо розв'язання задач розпізнавання. Досліджуються алгебраїчні замикання алгоритмів класичної моделі обчислення оцінок.

В работе описываются результаты, полученные в последние годы в рамках алгебраического подхода к решению задач распознавания. Исследуются алгебраические замыкания алгоритмов классической модели вычисления оценок.

Жук Д.В., Тузиков А.В. ***Сопоставление стереоизображений как задача о назначении*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 204-210.

УДК 004.932

У роботі наводиться алгоритм знаходження диспаратности для пари знімків. У цьому алгоритмі задача знаходження сполучених точок формулюється в термінах задачі про призначення. Запропонований алгоритм обробляє кожний рядок зображення

індивідуально, подібно методові динамічного програмування для знаходження відповідностей. Однак на відміну від методу динамічного програмування, розглянутий метод не використовує обмеження на порядок проходження відповідних пікселей. Дане обмеження не виконується, наприклад, коли сцена містить вузькі об'єкти на передньому плані. Запропонований алгоритм дозволяє визначати області, видимі тільки на одному із зображень, а також гарантує однозначність відповідностей.

В работе приводится алгоритм нахождения диспаратности для пары снимков. В этом алгоритме задача нахождения сопряженных точек формулируется в терминах задачи о назначении. Предлагаемый алгоритм обрабатывает каждую строку изображения индивидуально, подобно методу динамического программирования для нахождения соответствий. Однако в отличие от метода динамического программирования, рассматриваемый метод не использует ограничение на порядок следования соответствующих пикселей. Данное ограничение не выполняется, например, когда сцена содержит узкие объекты на переднем плане. Предлагаемый алгоритм позволяет определять области, видимые только на одном из изображений, а также гарантирует однозначность соответствий.

Ильченко А.В. ***Ключевые антицепи решетки описаний интервалов признакового пространства***. – 2008. – № 1. – С. 211-222.

УДК 517.9

У статті розглянуті поняття ключового антиланцюга решіткі описань інтервалів ознакового простору, властивості сімейства ключових антиланцюгів, алгоритм побудування ключових антиланцюгів.

В статье рассматриваются понятие ключевой антицепи решетки описаний интервалов признакового пространства, свойства семейства ключевых антицепей, алгоритм построения ключевых антицепей.

Иофина Г.В. ***Многомерное шкалирование в случае матриц попарных расстояний с элементами из конечного множества*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 223-229.

УДК 519.7

Розглядається задача багатовимірного шкалювання, в якій значення недіагональних елементів матриці попарних відстаней приймають одне з двох різних значень. Вважається, що елементи матриці, що знаходяться вище головної діагоналі, не спадають по рядках і не зростають по стовпцях. В роботі описаний загальний вигляд розглянутих матриць і в просторах розмінності $t = 1, 2, 3$ знайдені всі можливі матриці, що задовольняють умовам. Крім того, знайдені точні межі для

розмірностей матриць, які можуть бути матрицями попарних відстаней для об'єктів з евклідового простору розмірності t.

Рассматривается задача многомерного шкалирования, в которой значения недиагональных элементов матрицы попарных расстояний принимают одно из двух различных значений. Считается, что элементы матрицы, находящиеся выше главной диагонали, не убывают по строкам и не возрастают по столбцам. В работе описан общий вид рассматриваемых матриц, и в пространствах размерности $t = 1, 2, 3$ найдены все возможные матрицы, удовлетворяющие условиям. Кроме того, найдены точные границы для размерностей матриц, которые могут являться матрицами попарных расстояний для объектов из евклидова пространства размерности t.

Ковалев В.А. ***Методы поиска биомедицинских изображений в базе данных по их содержанию*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 230-244.

УДК 004.9

Розглядається проблема пошуку біомедичних зображень у великих базах даних по їхньому змісту. Характеризується сучасний стан проблеми та можливі шляхи її розв'язання на основі використання узагальнених матриць спільної зустрічальності різних типів. Наводиться загальна схема організації систем пошуку. Виклад ілюструється на широкому спектрі прикладів з використанням баз даних реальних біомедичних зображень.

Рассматривается проблема поиска биомедицинских изображений в больших базах данных по их содержанию. Характеризуется современное состояние проблемы и возможные пути ее решения на основе использования обобщенных матриц совместной встречаемости различных типов. Приводится общая схема организации систем поиска. Изложение иллюстрируется на широком спектре примеров с использованием баз данных реальных биомедицинских изображений.

Копылов А.В., Мельников П.А. ***Итерационные процедуры обработки изображений на основе ациклического динамического программирования*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 245-253.

УДК 004.932

Існує достатньо широкий клас завдань обробки зображень, які можуть бути сформульовані математично як оптимізаційні задачі, зокрема як (min, +) задачі розмітки, визначені на гратчастих графах суміжності цільових змінних. Відомо, що в загальному випадку дані задачі є NP-повними, але якщо граф суміжності не має циклів, то вони легко вирішуються на основі процедури динамічного програмування.

У даній роботі розглядається метод розв'язання подібних задач на основі принципу Гаусса-Зайделя, з ітераційним перерахунком значень груп змінних, зв'язаних деревовидними стосунками сусідства і зміною способу такого угрупування від ітерації до ітерації як засобом підвищення стійкості алгоритму до локальних екстремумів.

Существует достаточно широкий класс задач обработки изображений, которые могут быть сформулированы математически как оптимизационные задачи, в частности как (min, +) задачи разметки, определенные на решетчатых графах смежности целевых переменных. Известно, что в общем случае данные задачи являются NP-полными, но если граф смежности не имеет циклов, то они легко разрешимы на основе процедуры динамического программирования. В данной работе рассматривается метод решения подобных задач на основе принципа Гаусса-Зайделя, с итерационным пересчетом значений групп переменных, связанных древовидными отношениями соседства и изменением способа такой группировки от итерации к итерации как средством повышения устойчивости алгоритма к локальным экстремумам.

Котик С.В. ***Скелетизация полутонового изображения на примере изображений отпечатков пальцев*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 254-259.

УДК 004.932

У статті розглядається проблема побудови скелета сірого зображення на прикладі зображень відбитків пальців. Основна ідея побудови такого скелета полягає в тому, що він збирається з частин скелетів бінарних зображень, які отримані при бінаризації вихідного зображення з різними рівнями яскравості. Фрагменти скелетів таких бінарних зображень дозволяють у сукупності побудувати чітке бінарне зображення вихідного сірого зображення, по якому можлива кінцева побудова скелета, який дозволяє проводити порівняння особливих точок папілярного візерунку. Скелет, що отримано у такій спосіб, розглядають як скелет сірого зображення.

В статье рассматривается проблема построения скелета серого изображения на примере изображений отпечатков пальцев. Основная идея построения такого скелета состоит в том, что он собирается из частей скелетов бинарных изображений, полученных при бинаризации исходной картинки с различными уровнями яркости. Фрагменты скелетов таких бинарных изображений в совокупности позволяют построить четкое бинарное изображение исходной серой картинки, по которому можно строить окончательный скелет, позволяющий проводить сравнение особых точек папиллярного узора. Полученный таким образом скелет рассматривается как скелет серого изображения.

Кравцов А.А., Липницкий С.Ф., Степура Л.В. ***Система автоматического индексирования и реферирования текстовых документов*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 260-266.

УДК 004.912

Розглянуті теоретичні передумови і програмна реалізація системи індексування і реферування текстових документів на основі використання бази знань про наочну область у вигляді ситуатівно-синтагматіческой мережі. У системі використовується запропонований авторамипідхід до обчислення інформативності лексем і реченнь, заснований на зіставленністатистичних характеристик словоформ в реферованому тексті і повному корпусі текстів. Система може бути використана в науково-технічних бібліотеках для автоматизованого індексування і анотування науково-технічних статей і книг, а також користувачами Інтернет для попереднього ознайомлення із змістом текстових документів.

Рассмотрены теоретические предпосылки и программная реализация системы индексирования и реферирования текстовых документов на основе использования базы знаний о предметной области в виде ситуативно-синтагматической сети. В системе используется предложенный авторами подход к вычислению информативности лексем и предложений, основанный на сопоставлении статистических характеристик словоформ в реферируемом тексте и полном корпусе текстов. Система может быть использована в научно-технических библиотеках для автоматизированного индексирования и аннотирования научно-технических статей и книг, а также пользователями Интернет для предварительного ознакомления с содержанием текстовых документов.

Краснопрошин В.В., Виссия Х., Вальвачев А.Н. ***Принятие решений в оперативных задачах регионального управления*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 267-273.

УДК 004.9

У роботі розглянуті проблеми, що виникають при рішенні завдань оперативного регіонального й відомчого керування. Запропоновано методи керування для тисяч вилучених організацій на основі різнорідних даних.

В работе рассмотрены проблемы, возникающие при решении задач оперативного регионального и ведомственного управления. Предложены методы управления для тысяч удаленных организаций на основе разнородных данных.

Красоткина О.В., Моттль В.В. ***Методы регуляризации в задаче восстановления нестационарной регрессионной зависимости*** // Таврический вестник информатики и математики. – 2008. – № 1. – С. 274-283.

УДК 004.9311

Задача оцінювання нестаціонарної регресії практично завжди зв'язне з необхідністю вибору підмножини релевантних регресорів і визначення відповідного рівня нестаціонарної регресійної моделі який може змінюватися від повної стаціонарності миттєвих моделей до їх повної незалежності один від одного. У даній роботі завдання нестаціонарної регресії аналізується з погляду байесовського підходу, відповідно до якого оцінювана послідовність коефіцієнтів регресії розглядається як прихований вектор стану лінійної динамічної системи чия апріорна модель містить параметри, що відповідають і за розмір підмножини активних регресорів, і за ступінь тимчасової мінливості нестаціонарних коефіцієнтів регресії. Запропонований в роботі підхід дозволяє оцінити одночасно і вектор стану і параметри прихованої моделі стану.

Задача оценивания нестационарной регрессии практически всегда связна с необходимостью выбора подмножества релевантных регрессоров и определения подходящего уровня нестационарности регрессионной модели, который может изменяться от полной стационарности мгновенных моделей до их полной независимости друг от друга. В данной работе задача нестационарной регрессии анализируется с точки зрения байесовского подхода, в соответствии с которым оцениваемая последовательность коэффициентов регрессии рассматривается как скрытый вектор состояния линейной динамической системы, чья априорная модель содержит параметры, отвечающие и за размер подмножества активных регрессоров, и за степень временной изменчивости нестационарных коэффициентов регрессии. Предложенный в работе подход позволяет оценить одновременно и вектор состояния и параметры скрытой модели состояния.

СПИСОК АВТОРОВ НОМЕРА

Абламейко Сергей Владимирович — д. т. н., профессор, член-корреспондент НАН Беларуси, генеральный директор Объединенного института проблем информатики НАН Беларуси
e-mail: abl@newman.bas-net.by

Акимов Олег Михайлович — аспирант, Тюменский государственный университет, инженер-программист НИИ ИИС
e-mail: akimov-oleg@ya.ru

Амиргалиев Едилхан Несипханович — д. т. н., доцент, Казахский национальный технический университет имени К.И. Сатпаева
e-mail: amir_ed@mail.ru

Апарин Геннадий Петрович — к. т. н., доцент, ведущий научный сотрудник Объединенного института проблем информатики НАН Беларуси
e-mail: aparin@newman.bas-net.by

Архипов Вячеслав Игоревич — аспирант Объединенного института проблем информатики НАН Беларуси
e-mail: arkhipau@gmail.com

Бакина Ирина Геннадьевна — студентка МГУ им. М. В. Ломоносова
e-mail: irina_msu@mail.ru

Бауман Евгений Викторович — д. т. н., профессор, ведущий научный сотрудник Институт проблем управления РАН (ИПУ РАН)
e-mail: bau@ipu.ru

Белоцерковский Алексей Маратович — к. т. н., Объединенный институт проблем информатики НАН Беларуси
e-mail: abelotser@newman.bas-net.by

Бериков Владимир Борисович — д. т. н., доцент, Институт математики СО РАН, с.н.с.
e-mail: berikov@math.nsc.ru

Богуш Армен Лазаревич — к. т. н., зав. лаб. «Анализа биомедицинских изображений» Объединенного института проблем информатики НАН Беларуси
e-mail: v.kovalev@tut.by

Борисов Александр Евгеньевич — к. ф.-м. н., инженер-исследователь, ЗАО «Интел А/О» филиал в Н.Новгороде
e-mail: alexander.borisov@intel.com

Брусенцов Николай Петрович к. т. н., ф-т ВМиК, МГУ им. М.В. Ломоносова, и.о. зав. лабораторией
e-mail: ramil@cs.msu.su

Вальвачев Александр Николаевич с. н. с., НИЛ «Информационных технологий и компьютерной графики», кафедра математического обеспечения АСУ

Викентьев Александр Александрович к. ф.-м. н., доцент,с. н. с. Института Математики СО РАН
e-mail: vikent@math.nsc.ru

Викентьев Руслан Александрович аспирант Института Математики СО РАН
e-mail: vikent@math.nsc.ru

Вятченин Дмитрий Аркадьевич к. филос. н., с. н. с. лаборатории распознавания и обработки изображений Объединенного института проблем информатики НАН Беларуси
e-mail: viattchenin@mail.ru

Гончеренко Василий Георгиевич м. н. с. Объединенного института проблем информатики НАН Беларуси
e-mail: vasily@mpen.bas-net.by

Домахина Людмила Григорьевна аспирантка МГУ ВМиК
e-mail: Ludmila.domakhina@gmail.com

Дорофеюк Александр Александрович д. т. н., профессор, зав. лаб. Института проблем управления РАН (ИПУ РАН)
e-mail: adorof@ipu.ru, daa2@ mail.ru

Дорофеюк Юлия Александровна аспирантка, м. н. с. Института проблем управления РАН (ИПУ РАН)
e-mail: tigress86@bk.ru

Дышкант Наталья Федоровна студентка 5 курса МГУ им. М.В. Ломоносова, кафедра мат. методов прогнозирования
e-mail: nfd3001@gmail.com

Дьяконов Александр Геннадьевич к. ф.-м. н., доцент факультета ВМиК, МГУ им. М.В. Ломоносова
e-mail: djakonov@mail.ru

Ерухимов Виктор Львович инженер-исследователь, ЗАО «Интел А/О» филиал в Н.Новгороде
e-mail: victor.eruhimov@intel.com

Жук Дмитрий Викторович сотрудник Объединенного института проблем информатики НАН Беларуси
e-mail: dzhuk@tut.by

Ильченко Анатолий Васильевич старший преподаватель кафедры информатики Таврического национального университета им. В.И. Вернадского

Иофина Галина Владимировна аспирантка Московского физико-технического института
e-mail: giofina@gmail.com

Ковалев Василий Алексеевич м. н. с. Объединенного института проблем информатики НАН Беларуси
e-mail: bogush@newman.bas-net.by

Копылов Андрей Валериевич к. т. н., доцент каф. «Автоматика и телемеханика» Тульский государственный университет
e-mail: kopylov@uic.tula.ru

Котик Сергей Витальевич аспирант, ВЦ РАН
e-mail: kotik@micex.com

Кравцов Аркадий Анатольевич к. т. н. заведующий отделом Объединенного института проблем информатики НАН Беларуси
e-mail: kravtsov@newman.bas-net.by

Краснопрошин Виктор Владимирович д. т. н., заведующий кафедрой математического обеспечения АСУ, факультет прикладной математики и информатики, Белорусский государственный университет
e-mail: krasnoproshin@bsu.by

Красоткина Ольга Вячеславовна к. ф.-м. н., доцент, Тульский государственный университет
e-mail: krasotkina@uic.tula.ru

Липницкий Станислав Феликсович д. т. н., с. н. с. Объединенного института проблем информатики НАН Беларуси
e-mail: lipn@newman.bas-net.by

Мартьянов Владимир Юрьевич инженер-исследователь, ЗАО «Интел А/О» филиал в Н.Новгороде
e-mail: vladimir.martyanov@intel.com

Моттль Вадим Вячеславович д. т. н., ВЦ РАН, г. Москва
e-mail: vmottl@yandex.ru

Мусабаев Рустам Рафикович аспирант, Казахский национальный технический университет имени К.И. Сатпаева
e-mail: rmusabayev@kib.kz

Степура Людмила Васильевна — м. н. с. Объединенного института проблем информатики НАН Беларуси
e-mail: lstepura@mail.ru

Тузиков Александр Васильевич — д. ф.-м. н., профессор, зам. генерального директора Объединенного института проблем информатики НАН Беларуси
e-mail: bogush@newman.bas-net.by

Файзуллин Рашит Тагирович — д. т. н., профессор, ГОУ ВПО Омский гос. университет, и.о. зав. каф. комплексных систем защиты информации
e-mail: r.t.faizullin@mail.ru, rtf@univer.omsk.su

Херман Виссиа (Нидерланды) — аспирант кафедры математического обеспечения АСУ

Шапцев Валерий Алексеевич — д. т. н., профессор, Тюменский государственный университет, директор НИИ ИИС
e-mail: vashaptsev@ya.ru

ДО ВІДОМА АВТОРІВ

Загальні положення

Для опублікування в журналі "Таврійський вісник інформатики і математики" приймаються раніше не опубліковані наукові праці в галузі математики та теоретичної інформатики, згідно зі списком провідних тематичних розділів.
Автору(-ам) потрібно надавати такі документи:

1. Відомості про автора(-ів) (прізвище, ім'я, по батькові, учені ступені та звання, місце роботи та посада, адреси проживання та організації, телефон, факс, адреса електронної пошти тощо).
2. Рецензію сторонньої організації (бажано).
3. Статтю, надруковану на принтері.
4. Файл статті на дискеті 3,5" або надісланий електронною поштою за адресою редакції.

Вимоги до рукописів

1. Основні елементи статті розміщуються у такій послідовності: індекс УДК, ініціали та прізвище автора, назва статті, анотація (до 10 рядків) українською, російською та англійською мовами (анотація повинна містити конкретну інформацію про отримані результати), текст, список літератури.
2. Стаття може бути написана українською, російською або англійською мовою. Обсяг статті повинен не перевищувати 10 сторінок разом з малюнками, таблицями, графіками (не більше трьох) та бібліографією. Стаття повинна бути структурована (поділена на розділи із заголовками).
3. ***Відповідно до постанови Президії ВАК України від 15 січня 2003 року №7-05/1*** текст статті повинен бути викладений лаконічно, зрозуміло і відповідати такій структурній схемі.
   У вступі необхідно чітко виділити (курсивом) такі пункти:

   Постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями
   Аналіз останніх досліджень і публікацій, в яких започатковано розв'язання даної проблеми і на які спирається автор
   Невирішені раніше частини загальної проблеми, котрим присвячується зазначена стаття
   Формулювання цілей статті (постановка задачі)

   У висновку з данного дослідження необхідно чітко виділити (курсивом)

   результати дослідження *та перспективи подальших розвідок у цьому напрямку*

4. У статті необхідно дотримуватись термінології, прийнятої державним стандартом; використовуючи новий термін або абревіатуру, автор повинен розшифрувати та пояснити їх.
5. Використана література наводиться загальним списком наприкінці статті за порядком посилання на неї в тексті (в квадратних дужках) мовою оригіналу, відповідно до форми Ф23 бюлетеню ВАК України, 2000, №2.

6. Стаття має бути підготовлена за допомогою видавничої системи LATEX з використанням стильового пакету twim.sty, який можно отримати за адресою **www.twim.crimea.edu**.

Робота редакції з авторами

1. Матеріали необхідно надіслати електронною поштою, а також у вигляді "твердої" копії за адресою редакції: **Таврійський національний університет ім. В.І. Вернадського, пр-т Вернадського, 4, м.Симферопіль, Крим, Україна, 95007**, e-mail: **twim_taurida@mail.ru**
2. Редакція залишає за собою право внесення змін редакційного характеру без згоди з автором (-ами).
3. За необхідності автору (-ам) надсилається коректура статті.
4. Остаточне рішення про публікацію приймає редакційна колегія.
5. Рукопис, який надійшов до редакції з порушенням зазначених правил оформлення, не реєструється і не розглядається, а повертається автору (-ам) для доопрацювання.

ДО УВАГИ АВТОРІВ!

Про підвищення вимог до фахових видань, внесених до переліків ВАК України

ПОСТАНОВА ПРЕЗИДІЇ ВИЩОЇ АТЕСТАЦІЙНОЇ КОМІСІЇ УКРАЇНИ від 15.01.2003 р. №7-05/1

Необхідною передумовою для внесення видань до переліку наукових фахових видань України є їх відповідність вимогам пункту 7 постанови Президії ВАК України від 10.02.1999 р. №1-02/3 "Про публікації результатів дисертацій на здобуття наукових ступенів доктора і кандидата наук та їх апробацію".

... Редакційним колегіям організувати належне рецензування та ретельний відбір статей до друку. Зобов'язати їх приймати до друку у видання 2003 року та й у подальші роки лише наукові статті, які мають такі необхідні елементи: постановку проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями; аналіз останніх досліджень і публікацій, в яких започатковано розв'язання даної проблеми і на які спирається автор; виділення невирішених раніше частин загальної проблеми, котрим присвячується зазначена стаття; формулювання цілей статті (постановка задачі); виклад основного матеріалу дослідження с повним обгрунтуванням наукових результатів; висновки з даного дослідження і перспективи подальших розвідок у цьому напрямку.

Голова ВАК України В.В.Скопенко

Вчений секретар Л.М.Артюшин

Подписано к печати 16.05.2008. Формат 38x30/2. Бумага тип ОП. Объем 9.5 п.л. Тираж 500 экз. Заказ 335.
Издано в редакционном отделе КНЦ НАНУ
просп. Вернадского, 2, г. Симферополь, АРК, Украина, 95007