ON A LOGICAL REGULARITIES BASED METHOD OF DEFINITE QUALITY OBJECT SYNTHESIS

Alexander Dokukin

Abstract: *In this paper a method of definite quality object synthesis is proposed. The quality is described only by precedent information, i. e. examples of its presence or absence in similar objects. The method is based on using of special logical regularities which come out of solving a standart recognition task. It is applicable in a certain case of data and cannot be considered a general one.*

Keywords: *object synthesis, algebraic approach, pattern recognition.*

ACM Classification Keywords: *I.5 Pattern Recognition — I.5.0 General.*

Introduction

The task which is considered in the present article arises in connection to a general theme of definite structure protein synthesis. Yet we will try to abstract away from the application domain and to refrain from stating any biophysical conclusions and even affixing of the proposed methods to a specific data. The only way we intend to use the data is illustrating recognition quality at some point. Nevertheless, the end goal of this and following articles, that is developing of a method for construction of polipeptide chain with the definite profile of structural information and some restrictions to the amino-acid composition, demands such an interpretation. That is why the proposed methods are designed with additional requirement of interpretability.

Problem stating

In general the synthesis task can be described as following. Let we have a sample of objects which do or don't have some quality. Let we also have some set of restrictions. The problem is finding an object that satisfies the restrictions and has the quality. It is assumed that the initial sample can be strongly incomplete and inconsistent that eliminates simple enumeration. In addition it is considered desirable to find an object that has the target quality "in uppermost degree" for reliability of the solution.

Let's write down the above-said more formally.

Definition 1. *A set $P = \{P_1, \ldots, P_k\}$ will be called the alphabet and its elements will be refered to as letters.*

Definition 2. *By a word of length t a vector $\beta = (\beta_1, \ldots, \beta_t)$ will be considered, such that $\beta_i \in P$.*

Definition 3. *A set $B_i \times \ldots \times B_t$, where $B_i \subset P$ and either $|B_i| = 1$ or $B_i = P$ will be called a template of length t.*

The meaning of a template is quite simple. It is a set of words restricted to a single specific letter in some positions and unrestricted in others.

Let us have a training sample $\beta_1, \ldots, \beta_m$ of words of length n. Let this sample be split into two classes K_1 and K_0, that is words having and not having some μ-quality correspondingly. Finally, let us have a template T of length n.

Definition 4. *By the elementary synthesis task we will call a search for the word $\beta' \in T$, such that*

$$\beta' = \arg\max_{\beta \in T} d(\beta)\,,$$

where $d(\beta)$ is an estimate of presence of μ-quality in a word β.

Remark 1. *The task is called elementary since the length of the template is equal to the length of sample words. The synthesis of a word by a longer template some parts of which are marked as having or not having the quality is of interest also. Such a general task is not considered in the present paper.*

The key quality of the task is that the spicific of the application domain doesn't allow defining the functional d. That is why it is proposed using the training sample for its calculation. Hereat it seems natural using recognition algorithms for the purpose.

The standard recognition task [Zhuravlev, 1977] is stated as follows. Let us have a training sample $S_1, \ldots, S_m$, described by vectors of some nature $S_i = (a_{i1}, \ldots, a_{in})$. The sample is split into l classes $K_1, \ldots, K_l$ that can overlap in general case. The training sample classification that is vectors $\boldsymbol{\alpha_i} = (\alpha_{i1}, \ldots, \alpha_{il})$ is known. Here α_{ij} is the value of "$S_i \in K_j$" predicate. It is required to construct an algorithm A that can calculate classification of a new object S.

There is a theorem [Zhuravlev, 1977] among the basic propositions of the algebraic recognition theory stating that any recognition algorithm A can be represented as a composition CB. Here B is a recognition operator calculating some realvalue estimates of class membership

$$\{S_1, \ldots, S_m, \boldsymbol{\alpha_1}, \ldots, \boldsymbol{\alpha_m}, S\} \overset{B}{\to} (\Gamma_1, \ldots, \Gamma_l) \in \mathbb{R}^l ,$$

and C is a decision rule

$$\mathbb{R}^l \overset{C}{\to} (0,\ 1,\ \Delta)^l ,$$

where Δ means denial.

Let's construct a recognition operator B using synthesis training sample.

Definition 5. *The functional d is defined as $d(\beta) = \Gamma_1(\beta) - \Gamma_0(\beta)$, where $\Gamma_1(\beta)$ and $\Gamma_0(\beta)$, are estimates of the word β for classes K_1 and K_0 calculated by the operator B correspondingly.*

Recognition algorithm

Now it is important to mention that all the experimental research of ideas and algorithms described here is made with quite a specific data. The alphabet consists of 20 letters corresponding to amino-acid residues. The words of length 5 describe information units [Nekrasov, 2004] that are basic segments of polieptide chain that can have structure forming quality. The sample is divided into two classes by presence of the quality.

During initial experiments a recognition algorithm has been proposed that provides $88,8\%$ recognition quality [Senko et al., 2011] for the task. There is a couple of drawbacks preventing the solution from being used in synthesis task.

First of all, each letter was encoded by ten contiuous features. That alone complicates interpretation of any results in terms of application domain. Ideally it is required to exercise synthesis based on a small number of simple and interpretable rules. Consiquently a set of simple rules must underlie the recognition algorithm.

Secondly, initial test were carried out with a one thousand objects sample that has been split into two parts for the case. Now we deal with a sample containing at least five hundreed thousend information units equally divided into two classes. This quantities themself prevent use of conventional recognition methods. Besides, there is an five percent overlap of classes that was not the case in initial study.

Thus, we require the recognition algorithm to have the following features:

1. recognition results must be interpretable in therms of application domain,

2. recognition quality must not be much worse than the previously achieved one,

3. the algorithm has to deal with enormous data sets.

We propose using special kind of logical regularities [Ryazanov, 2007] for the task.

The regularities will be sought in form of templates (see Definition 3). Its significance will be determined by Fisher's test [Fisher, 1922] using the approximate formula [Vorontsov, 2007].

Definition 6. *Let's consider a template T. We will call it a logical regularity of class K_1 with significance $P(T)$ if*

$$P(T) = -ln\left(\frac{C_{X+Y}^{x+y}}{C_X^x C_Y^y}\right) > 0 ,$$

where $X = |\{\beta_1, \ldots, \beta_m\} \cap K_1|$, $Y = |\{\beta_1, \ldots, \beta_m\} \cap K_0|$, $x = |T \cap K_1|$, $y = |T \cap K_0|$.
Class K_0 regularities are defined the same way.

Let R_1 and R_0 be sets of logical regularities of classes K_1 and K_0 correspondingly.

Definition 7. *We will call an estimate $\Gamma_s(\beta)$ of the word β for K_s class the value*

$$\Gamma_s(\beta) = \sum_{U \in \{T_k \in R_s \,|\, \beta \in T;\ T_i \not\subset T_j,\ i \neq j\}} P(U) .$$

Remark 2. *As an estimate for the class a word gets sum of significances of all satisfied regularities of that class minus sum of significances of satisfied regularities of the other one. Hereat if the word satisfies two regularities one of which contains the other the latter's significance is not taken into account.*

The described method satisfies all the requirements we demand from it.

First, the templates of the proposed structure allow determining dependencies between letters in different positions in a most explicit way. Nested templates then can be used to refine those dependecies if needed.

Second, the structure of logical regularities allows performing complete enumeration of templates for the required ammounts of data.

Statement 1. *Let n, m and p be the sizes of template, training sample and alphabet correspondingly. There is an algorithm that calculates values x and y of Definition 7 for all possible regularities in C operations, where*

$$C = n2^{n-1}m .$$

Hereat $2\left((p+1)^n - 1\right)$ integer values is enough to store all the required counters.

Finally, we have achieved $78,4\%$ recognition quality after testing the method on an independent subset that was equal to training sample in size (the whole set of data had been divided into two equal subsets for training and testing). The quality is achieved with significance threshold 20. On the one hand, the quality is ten percent worse than the initial one. On the other, we already mentioned that the new training sample has classes overlaped by $5,2\%$ objects. That is why we consider the result acceptable.

Synthesis algorithm

Template sizes in elemetary synthesis task also allow searching the solution by complete enumeration. But having in mind possible generalizations we must take into account its extreme growth. That is why we consider important achieving an algorithm less dependent of the template size even for elementary synthesis.

Let's consider a template $T = B_1 \times \ldots \times B_n$. Let $B_i = P$, i. e. all letters are allowed in i-th position.

Definition 8. *Let's denote by $T[i]\gamma$ the refining of template T in the position i,*

$$T[i]\gamma = \{\beta \in T \,|\, \beta_i = \gamma\}\ ,$$

where γ is some letter from the alphabet, $\gamma \in P$.

We propose the following synthesis algorithm. For each position i of the template in which the letter is not constricted we determine it by the formula

$$\gamma'_i = \arg\max_{\gamma \in P}\Big(\sum_{U \in \{B_k \in R_1 \,|\, T[i]\gamma \subset B_k\}} P(U) - \\ - \sum_{V \in \{B_k \in R_0 \,|\, T[i]\gamma \subset B_k\}} P(V)\Big)\ .$$

It means that each letter will be independantly refined by the value that achieves the highest estimate as refinement of the initial template. The estimate of the refined template is calculated as difference of significances of templates containing it.

Theorem 1. *Let the functional d be determined by Definitions 5 and 7. If the proposed algorithm finds a unique solution for each unrestricted letter of template T then it calculates the optimal solution of the elementary synthesis task (see Definition 4).*

Proof. Let's prove the theorem ad absurdum. The following is the definition of the optimal solution of the synthesis task β':

$$\beta' = \arg\max_{\beta \in T} \sum_{U \in \{B_k \in R_1 \,|\, \beta \in B_k;\ B_i \not\subset B_j,\ i \neq j\}} P(U) - \\ - \sum_{V \in \{B_k \in R_0 \,|\, \beta \in B_k;\ B_i \not\subset B_j,\ i \neq j\}} P(V)\ .$$

Let there be a position h in which $\gamma'_h \neq \beta'_h$, where γ'_h is achieved by the synthesis algorithm.
Let's consider a set of regularities generating estimates of the optimal solution for the class K_s.

$$R'_s = \{B_k \in R_1 \,|\, \beta' \in B_k;\ B_i \not\subset B_j,\ i \neq j\}\ .$$

This set is divided into two parts

$$R'_s = \{U \,|\, U_h = \beta'_h\} \cup \{U \,|\, U_h = P\}\ .$$

Hereat the second item takes part in estimation regardless the letter in position h. The first one consists of the unique element U such that $T[h]\beta'_h \subset U$. Indead one such regularity must exist by the theorem condition because the algorithm comes to a solution. Refinements of the regularity in turn do not take part in estimating.

Since the regularity U must be used in search of γ'_h by the synthesis algorithm definition, we come to a contradiction that proves the theorem. ■

Since we don't have exact criteria for estimating synthesis algorithm quality, that for example can be used in comparison of two methods, we will show some indirect indicators achieved experimentally. For the experiment the same testing sample was used that consisted of the half of initial data. For each word a template was generated by removing two letters in random positions. Theese templates were used for synthesis without dublicate filtering. Synthesis algorithm was made searching structure forming units if the initial word corresponded to a structure forming one and vice versa.

It is important that in all cases the proposed algorithm has come to a solutions, i. e. the theorem condition was satisfied. Hereat in most cases both letters were changed comparing to the initial ones that seems natural.

Only in $39,7\%$ of cases the synthesized word was found in the initial sample (among all the five hundred thousand objects) that justifies its name in some kind. Among them in $91,1\%$ cases the target class matched the class of the found precedent.

Conclusion

In the present paper a synthesis task has been described, i. e. the task of searching for an object of definite quality that is described by a set of precedents. An approach for solving the task has been proposed. An algorithm has been constructed for an important task family.

At the same time the question of formalizing the method quality estimate remains. Its solution as well as different generalization of the task, such as synthesis by the longer template or number of classes growth, are the most obvious ways for developement of theese results.

Acknowledgements

The paper is supported by RFBR, grant No. 12-01-31166.

The author also thanks A. N. Nekrasov, senior researcher in IBC RAS for the granted data.

Bibliography

Fisher R. A. On the Interpretation of X2 from Contingency Tables, and the Calculation of P // Journal of the Royal Statistical Society. — 1922. — Vol. 85, No. 1. — Pp. 87–94.

Zhuravlev Yu. I. Correct algebras over sets of incorrect (heuristic) algorithms I (in Russian) // Cybernetics. — 1977. — No. 4. — Pp. 14–21.

Nekrasov A. N. Analysis of the Information Structure of Protein Sequences: A New Method for Analyzing the Domain Organization of Proteins // Journal of Biomolecular Structure and Dynamics. — 2004. — Vol. 21, Iss. 5. — Pp. 615–623.

Ryazanov V. V. Logical regularities in pattern recognition (parametric approach) // Computational Mathematics and Mathematical Physics. — 2007. — Vol. 47, No. 10. — Pp. 1720–1735.

Vorontsov K. V. Lections on classification algorithms (in Russian). — 2007. — URL:
http://www.ccas.ru/voron/download/LogicAlgs.pdf.

Senko O. V., Nekrasov A. N., Ryazanov V. V., Dokukin A. A. Prediction of Structure forming Properties of Protein with Help of Pattern Recognition Methods // Proceedings of 8-th Open German"=Russian Workshop “Pattern Recognition and Image Understanding” OGRW-8-2011. — Pp. 38–40.

Authors’ Information

Alexander Dokukin *— Computing Centre of Russian Academy of Sciences, researcher, 40 Vavilova St., Moscow, Russia, 119333; e-mail: dalex@ccas.ru.*
Major Fields of Scientific Research: Algebraic Approach to Pattern Recognition.