FEATURE REPRESENTATION OF DNA SEQUENCES FOR MACHINE LEARNING TASKS

Robertas Damasevicius
Kaunas University of Technology

FEATURE REPRESENTATION OF DNA SEQUENCES FOR MACHINE LEARNING TASKS

Robertas Damaševičius

Software Engineering Department, Kaunas University of Technology,
Studentų 50-415, LT-51368, Kaunas, Lithuania
robertas.damasevicius@ktu.lt

ABSTRACT

Recognition of specific functionally-important DNA sequence fragments is one of the most important problems in bioinformatics. Common sequence analysis methods such as pattern search can not solve this problem because of noisy data and variability of consensus sequences across different species. Machine learning methods such as Support Vector Machine (SVM) can be used for sequence classification, because they can learn useful descriptions of genetic concepts from data instances only rather than explicit definitions. Before applying SVM classification one must define a mapping of classified sequences into a feature space. We analyze binary and k-mer frequency-based feature mapping rules. The efficiency of such rules is demonstrated for the recognition of promoters and splice-junction sites.

1. INTRODUCTION

Biomolecular data mining is the activity of finding significant information in DNA, RNA and protein molecules. The significant information may refer to periodicities, motifs, clusters, genes, protein signatures, grammar rules and classification rules. Common methods are sequence alignment using dynamic programming (Needleman-Wunsch, Smith-Waterman) and substitution-matrix based methods (PAM, BLOSUM). However, such methods are not very effective for recognition of some types of DNA sequence such as *promoters* (short sequences that precede the beginnings of genes) or *splice-junction sites* (boundary points between exons and introns where splicing occurs), because of noisy data and large variability of consensus sequences across species. For such difficult cases machine learning techniques such as Support Vector Machine (SVM) are applied [1-6].

SVM [7] is a supervised learning method for creating binary classification functions from a set of labeled training data. The accuracy of a particular parametric classifier on a given dataset will depend on the relationship between the classifier and the available dataset [8].

There are two groups of approaches for improving the classification results. The *algorithmic approach* focuses on improving or proposing new classification algorithms or kernels, whereas the *data processing approach* focuses on modifying the dataset or extracting relevant features to improve the accuracy of classification. SVM requires that each data instance is represented as a vector of binary/real numbers in a *feature space*. Thus, if there are categorical attributes, we first have to convert them into numeric data. Good mapping of data into feature space can allow achieving better classification accuracy.

Feature mapping rules such as binary mapping [9], frequency-based encoding [10], determinative degree [11], and features constructed based on a combination of various k-mers (k-base long sequences) [12-14] have been an object of extensive research. Here we analyze different position-dependent (binary) and position-independent (frequency-based) data encoding schemes for nucleotides (A, C, G, T) and their groupings (S/W, K/M, R/Y), which can achieve optimal classification results for promoter and splice-site recognition problems.

2. DNA MAPPING RULES

Before applying SVM classification to the available dataset, first, one must define a mapping of classified objects to the feature space. This mapping is called a *feature vector representation* of the subject area. A feature space can be constructed using a) *position-dependent* information, b) *position-independent* information, or c) *both* position-dependent and position-independent information about the presence or absence of specific nucleotides or their k-mers.

A feature mapping rule can be described as a function $M : \hat{S} \to F$, where $\hat{S} = (s_1, s_2, ..., s_N), s_i \in \{A, C, G, T\}^k$ is a DNA sequence, N is the length of a DNA sequence, $k = \|s_i\|$ is the length of the mapped sequence, and $F = (f_1, f_2, ..., f_M)$, where $f_j \in \{0,1\}^l$ in case of a binary feature space, or $f_j \in \Re$ in case of a real number feature space, M is the length of a feature vector, $l = \|f_i\|$ is the length (dimension) of a feature.

Depending upon the values of k and l, we classify binary feature mapping rules as the binary $1 \to 4$, $1 \to 2$, $1 \to 1$ and $2 \to 1$ rules. Feature mapping rules based on k-mer frequency are categorized according to the DNA alphabet they are applied on.

2.1. Binary mapping rules

2.1.1. Binary $1 \to 4$ rule

An example of the $1 \to 4$ rule is *orthogonal encoding*, where the nucleotides in a DNA sequence are represented by 4-dimensional orthogonal binary vectors:

$$\left\langle A \to (0001), C \to (0010), G \to (0100), T \to (1000) \right\rangle \quad (1)$$

For this rule, feature vector size is $4N$.

This rule allows achieving better classification results than the mapping of the nucleotides into 2-dimensional binary vectors, due to the identical Hamming distances between the nucleotide encodings [15].

2.1.2. Binary $1 \to 2$ rule

There are several methods to represent DNA nucleotides using a binary 2-bit code. Jiménez-Montaño *et al.* [16] suggested the rule $\mathrm{A}=00, \mathrm{G}=01, \mathrm{T}=10, \mathrm{C}=11$. Stambuk [17] defined the rule $\mathrm{T}=00, \mathrm{C}=01, \mathrm{G}=10, \mathrm{A}=11$. Karasev and Stefanov [18] suggested the rule $\mathrm{C}=00, \mathrm{T}=01, \mathrm{G}=10, \mathrm{A}=11$. He *et al.* [19] used the rule $\mathrm{C}=00, \mathrm{T}=10, \mathrm{G}=11, \mathrm{A}=01$.

Actually, there are $4!=24$ such rules; however, only 3 rules are essentially different, while the remaining rules can be obtained from these by inversion:

$$\text{Binary 1: } \left\langle A \to (0,0), C \to (0,1), G \to (1,0), T \to (1,1) \right\rangle \quad (2)$$

$$\text{Binary 2: } \left\langle A \to (0,0), C \to (0,1), G \to (1,1), T \to (1,0) \right\rangle \quad (3)$$

$$\text{Binary 3: } \left\langle A \to (0,0), C \to (1,1), G \to (0,1), T \to (1,0) \right\rangle \quad (4)$$

For these rules, feature vector size is $2N$.

2.1.3. Binary $1 \to 1$ rule

The $1 \to 1$ rules are unequal representation rules that map one nucleotide into 1 and the remaining nucleotides into 0. There are four such rules: A-rule, C-rule, G-rule and T-rule [20]. These rules reflect the distribution of a particular type of nucleotides along the DNA sequence:

$$\text{A-rule: } \left\langle A \to 1, B \to 0 \right\rangle, B = \{C,G,T\} \quad (5)$$

$$\text{C-rule: } \left\langle C \to 1, D \to 0 \right\rangle, D = \{A,G,T\} \quad (6)$$

$$\text{G-rule: } \left\langle G \to 1, H \to 0 \right\rangle, H = \{A,C,T\} \quad (7)$$

$$\text{T-rule: } \left\langle T \to 1, V \to 0 \right\rangle, V = \{A,C,G\} \quad (8)$$

For these rules, feature vector size is N.

2.1.4. Binary $2 \to 1$ rules

The $2 \to 1$ rules are based on the grouping of the 4-letter DNA alphabet into two subsets of two nucleotides each. There are 3 different such partitions, therefore there are 3 different binary mapping rules that map a nucleotide onto a binary number. Each of these rules represents a different aspect of the DNA molecule structure.

The SW mapping rule ($\{A,T\}$ vs. $\{G,C\}$) reflects the difference in the number of hydrogen bonds in the DNA molecule. Each strong (S) nucleotide (C or G) has 3 hydrogen bonds, and each weak (W) nucleotide (A or T) has only 2 hydrogen bonds. This rule is particularly appropriate to analyze genome-wide correlations [21].

$$\text{SW rule: } \left\langle S \to 1, W \to 0 \right\rangle, S = \{A,T\}, W = \{C,G\} \quad (9)$$

The RY rule ($\{A,G\}$ vs. $\{T,C\}$) describes how *purines* (R) and *pyrimidines* (Y) are distributed along the DNA sequence. This rule corresponds to the chemical composition bias in the DNA strand.

$$\text{RY rule: } \left\langle R \to 1, Y \to 0 \right\rangle, R = \{A,G\}, Y = \{C,T\} \quad (10)$$

The KM rule ($\{A,C\}$ vs. $\{T,G\}$) describes how *amines* (M) and *ketones* (K) are distributed along the DNA sequence.

$$\text{KM rule: } \left\langle K \to 1, M \to 0 \right\rangle, K = \{A,C\}, M = \{G,T\} \quad (11)$$

For these rules, feature vector size is N.

2.2. K-mer frequency rules

K-mers are lists or ordered sets of nucleotide sequence elements, which can be described as a k-tuple $(a_1, a_2, \ldots, a_k)$, where $a_i \in \hat{S}$ for all $\mathrm{i}=1,2,\ldots,\mathrm{k}$. Feature vector is constructed using a frequency (or probability) $p_j = \dfrac{n_j}{N-k+1}$ of each k-mer in a N-length sequence $\hat{S}$, where n_j is the number of j-th k-mer in $\hat{S}$. Traditionally, k-mers have been used with 4-letter DNA alphabet $\{A,C,G,T\}$. The disadvantage of such mapping rule is its explosive feature space growth: there may be 4^k distinct k-mers in a nucleotide sequence (actually, there are $N-k+1$ such k-mers in N-length sequence), and a feature vector is composed of 4^k elements:

$$\text{ACGT: } \left\langle \hat{S} \to \left(p_j\right) \right\rangle, \hat{S} \in \{A,C,G,T\}^N, j = 1,\ldots,4^k \quad (12)$$

We can construct smaller feature vectors based on the grouping of the 4-letter DNA alphabet into two subsets of two nucleotides each. There are three different such partitions, therefore there are three different grouping-based k-mer frequency mapping rules [22]:

SW k-mer rule: $\left\langle \hat{S} \to \left(p_j\right)\right\rangle, \hat{S} \in \{S,W\}^N, j = 1,...,2^k$ (13)

RY k-mer rule: $\left\langle \hat{S} \to \left(p_j\right)\right\rangle, \hat{S} \in \{R,Y\}^N, j = 1,...,2^k$ (14)

KM k-mer rule: $\left\langle \hat{S} \to \left(p_j\right)\right\rangle, \hat{S} \in \{K,M\}^N, j = 1,...,2^k$ (15)

Note that using SW, RY or KM groupings, a feature vector is much smaller than in case of full nucleotide alphabet, and is composed of only 2^k elements.

3. CASE STUDY

3.1. Datasets

For promoter classification, we use the 2002 collection of data of drosophila (*D. melanogaster*) core promoter regions [23]. The training file contains 1260 examples (372 promoters, 361 introns, 527 coding sequences). The test file contains 6500 examples (1842 promoters, 1799 introns, 2859 coding sequences).

For splice site recognition, we use the dataset from the UCI repository [24] obtained from Genbank 64.1 primate data. The dataset contains 3175 sequences, each 60 bp length starting at position -30 bp and ending at position +30 bp with regard to splice site location, of which 767 (25%) sequences contain exon/intron (EI) sites (donors), 768 (25%) sequences contain intron/exon (IE) sites (acceptors), and 1655 (50%) sequences contain neither EI nor IE sites (negative, N).

3.2. Problem definition

Dataset sequences are mapped into a feature space using feature mapping rules described in Eq. (1-15). A training dataset is an ordered set of features $F = \left(f_1, f_2, ..., f_M\right)$ of the sequences and their assigned class:

$$LS_M = \left\{\left(F_i, c_i\right) \mid i = 1,...,M\right\} \qquad (16)$$

The objective of the classification is to derive from LS_M a classifier $\hat{c}\left(F_j\right)$, which predicts the class of unseen sequence s_j as accurately as possible based on some selected classification accuracy metric. As a classifier, we use SVM$^{\text{light}}$ [25] with power series kernel [26].

3.3. Results

To represent the precision of promoter classification for binary mapping rules (Eq. 1-11) graphically, the *Receiver Operating Characteristic* (ROC) is used (see Figure 1). The perfect classification corresponds to the $(0,100)$ point in the ROC plot.

The best classification results are obtained using Binary 1, KM, A and T rules. This can be explained by the fact that drosophila promoter sequences are characterized by the repeating occurrences of the so called TATA box (TATAA or TATAAA) or the Pribnow box (TATAAT), thus the best results can be achieved using the rules, where A and T nucleotides are coded using different binary values (as, e.g., in the KM rule).

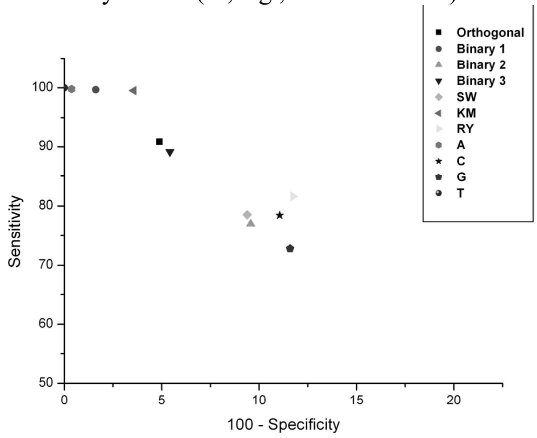

Figure 1. Comparison of promoter classification results for binary feature mapping methods

The splice site classification results for k-mer frequency rules (Eq. 12-15) are summarized in Figure 2.

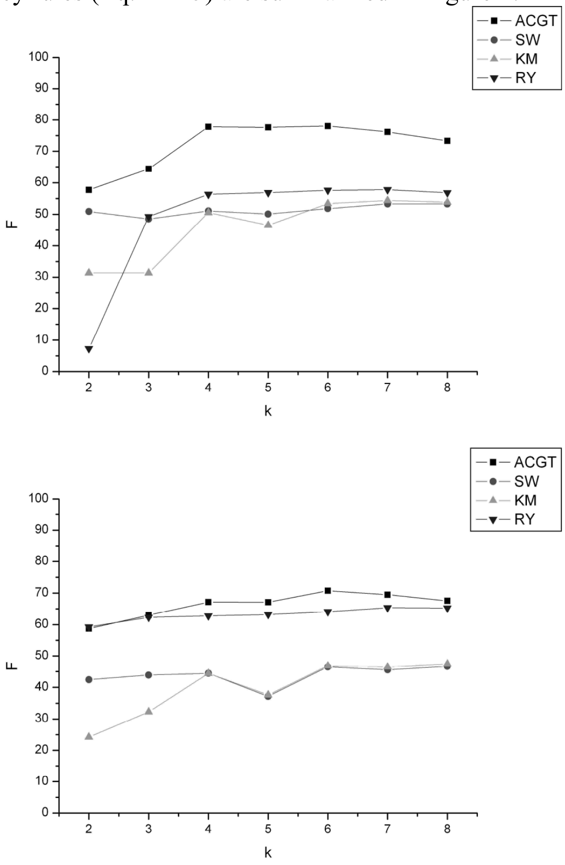

Figure 2. Comparison of splice-site sequence classification results for different k-mers using F-measure: EI vs. N (top), IE vs. N (bottom)

EI splice sites are best recognized with ACGT alphabet 4-mer frequencies (78.05%) and IE splice sites are best recognized using ACGT alphabet 6-mer frequencies (70.75%). Interestingly, 5-mers in both cases have worse recognition results for all types of frequency mapping rules than 4-mers or 6-mers. The classification accuracy for larger k-mers decreases. Out of the 2-nucleotide grouping based feature mapping rules, the RY frequency based feature mapping rule has the best results in both cases, and the results are only slightly worse than the results of the 4-nucleotide frequency mapping for IE splice site recognition. Therefore, if the classification speed is the issue, the RY frequency mapping for IE splice site recognition may yield nearly the same accuracy, though its feature space is significantly smaller (2^k instead of 4^k elements).

The k-mer frequency-based mapping rules perform worse than binary rules (though we solve different problems and use different accuracy metrics). This can be explained by the fact that in frequency based feature representation the nucleotide (k-mer) position information is lost. However, the advantage of frequency rules is smaller feature space, if long sequences are classified.

4. CONCLUSION

The selection of the appropriate feature mapping rule can greatly influence the DNA sequence classification results. The mapping rule should be selected based on the properties of the available data for a specific classification problem. The obtained classification results confirm that the mapping rule(s) with the best classification results correspond to the characteristics of the repeating subsequences ("boxes", consensus sequences) of the analyzed sequences. The selection between binary and frequency mapping rules can provide a trade-off between classification precision and speed.

Future work will focus on the feature space reduction problem to identify features that do not contribute to classification accuracy and can be discarded thus yielding higher recognition speed and better accuracy.

REFERENCES

[1] G. Rätsch, S. Sonnenburg and C. Schäfer, "Learning Interpretable SVMs for Biological Sequence Classification", *BMC Bioinformatics* 2006, 7(Suppl 1):S9.

[2] T. Werner, "The state of the art of mammalian promoter recognition",*Briefings in Bioinformatics* 4(1):22-30, 2003.

[3] A.C. Lorena, and A.C.P.L.F. de Carvalho, "Human Splice Site Identification with Multiclass Support Vector Machines and Bagging", in *Proc. of ICANN 2003*, Istanbul, Turkey, LNCS 2714, Springer, 234-244.

[4] S. Sonnenburg, G. Rätsch, A. Jagota and K. Müller, "New Methods for Splice Site Recognition", in *Proc. of ICANN 2002*, Madrid, Spain, LNCS 2415, Springer, 329-336.

[5] A. Baten, B. Chang, S. Halgamuge and J. Li, "Splice site identification using probabilistic parameters and SVM classification", *BMC Bioinformatics* 2006, 7:S15.

[6] S. Rampone, "Recognition of splice junctions on DNA", *Bioinformatics*, 14(8):676–684, 1998.

[7] V. Vapnik. *Statistical Learning Theory.* Wiley-Interscience, New York, 1998.

[8] C.M. van der Walt and E. Barnard, "Data characteristics that determine classifier performance", in *Proc. of the 16th Annual Symp. of the Pattern Recognition Association of South Africa*, pp. 160-165, 2006.

[9] B. Podobnik, J. Shao, N.V. Dokholyan, V. Zlatic, H.E. Stanley, I. Grosse, "Similarity and dissimilarity in correlations of genomic DNA", *Physica A,* 373:497-502, 2006.

[10] R. Ranawana, and V. Palade, "A neural network based multiclassifier system for gene identification in DNA sequences", *J. Neural Comput. Appl.* 2005, 14, 122–131.

[11] D. Duplij, and S. Duplij, "DNA sequence representation by trianders and determinative degree of nucleotides", *J Zhejiang Univ Sci B.* 2005 August; 6(8): 743–755.

[12] R. Islamaj, L. Getoor, and W.J. Wilbur, "A Feature Generation Algorithm for Sequences with Application to Splice-Site Prediction", in *Proc. of PKDD 2006,* Berlin, Germany, LNCS 4213, Springer, 553-560.

[13] Y. Saeys, S. Degroeve, D. Aeyels, P. Rouzé and Y. Van de Peer, "Feature selection for splice site prediction: A new method using EDA-based feature ranking", *BMC Bioinformatics* 2004, 5:64.

[14] T. Sobha Rani, S. Durga Bhavani and R.S. Bapi, "Analysis of E.coli promoter recognition problem in dinucleotide feature space", *Bioinformatics* 2007 23(5):582-588

[15] B. Demeler, and G.W. Zhou, "Neural network optimization for E. coli promoter prediction", *Nucleic Acids Res* 19:1593–9, 1991.

[16] M. Jimenez-Montano, C. Mora-Basanez, and T. Poschel, "The hypercube structure of the genetic code explains conservative and non-conservative amino acid substitutions in vivo and in vitro", *Biosystems* 1996, 39, 117-125.

[17] N. Stambuk, "Universal Metric Properties of the Genetic Code", *Croatica Chemica Acta* 2000, 73, 1123-1139.

[18] V.A. Karasev and V.E. Stefanov, "Topological Nature of the Genetic Code", *J. Theor. Biol.* 2001, 209, 303-317.

[19] M. He, S. Petoukhov, and P.E. Ricci, "Genetic Code, Hamming Distance and Stochastic Matrices", *Bull. Math. Biol.* 2004, 00, 1–17.

[20] R.F. Voss, "Evolution of long-range fractal correlations and 1/f noise in DNA base sequences", *Phy. Rev. Lett.* 1992, 68(25): 3805–3808.

[21] A. Arneodo, E. Bacry, P. Graves, and J.F. Muzy, "Characterizing long-range correlations in DNA sequences from wavelets analysis", *Phys. Rev. Lett.* 1995, 74, 3293–3296.

[22] R. Damaševičius, "Splice Site Recognition in DNA Sequences Using K-mer Frequency Based Mapping for Support Vector Machine with Power Series Kernel", *Proc. of Int. Conf. on Complex Software Intensive Systems (CISIS 2008)*, March 4–7, 2008, Barcelona, Spain, 687-692.

[23] Drosophila promoter dataset. http://www.fruitfly.org/seq_tools/datasets/Drosophila/

[24] The Machine Learning Database Repository. http://mlearn.ics.uci.edu/databases/molecular-biology/

[25] SVMlight. http://svmlight.joachims.org/

[26] R. Damaševičius, "Optimization of SVM Parameters for Promoter Recognition in DNA Sequences", *Int. Conf on Continuous Optimization and Knowledge-Based Technologies EurOPT-2008*, May 20-23, Neringa, Lithuania.