On Utilizing Optimal and Information Theoretic Syntactic Modeling for Peptide Classification

Eser Aygün[1], B. John Oommen[2], and Zehra Cataltepe[3]

[1] Department of Computer Eng., Istanbul Technical University, Istanbul, Turkey
[2] School of Computer Science, Carleton University, Ottawa, Canada : K1S 5B6, and *Adjunct Professor* at the University of Agder in Grimstad, Norway
[3] Department of Computer Eng., Istanbul Technical University, Istanbul, Turkey
eser.aygun@itu.edu.tr

Abstract. Syntactic methods in pattern recognition have been used extensively in bioinformatics, and in particular, in the analysis of gene and protein expressions, and in the recognition and classification of bio-sequences. These methods are almost universally distance-based. This paper concerns the use of an Optimal and Information Theoretic (OIT) probabilistic model [11] to achieve peptide classification using the information residing in their syntactic representations. The latter has traditionally been achieved using the edit distances required in the respective peptide comparisons. We advocate that one can model the differences between compared strings as a mutation model consisting of random Substitutions, Insertions and Deletions (SID) obeying the OIT model. Thus, in this paper, we show that the probability measure obtained from the OIT model can be perceived as a sequence similarity metric, using which a Support Vector Machine (SVM)-based peptide classifier, referred to as OIT_SVM, can be devised.

The classifier, which we have built has been tested for eight different "substitution" matrices and for two different data sets, namely, the *HIV-1 Protease Cleavage* sites and the *T-cell Epitopes*. The results show that the OIT model performs significantly better than the one which uses a Needleman-Wunsch sequence alignment score, and the peptide classification methods that previously experimented with the same two datasets.

Keywords: Biological Sequence Analysis, Optimal and Information Theoretic Syntactic Classifcation, Peptide Classification, Sequence Processing, Syntactic Pattern Recognition.

1 Introduction

The syntactic methods that have been traditionally used in the analysis, recognition and classification of bioinformatic data include distance-based methods, and probabilistic schemes which are, for example, Markovian. A probabilistic model, distinct from these, is the one proposed by Oommen and Kashyap [11]. The model, referred to as the OIT model, attains the optimal and information theoretic bound. This paper reports the first known results in which the OIT model has been applied in any bioinformatic application.

V. Kadirkamanathan et al. (Eds.): PRIB 2009, LNBI 5780, pp. 24–35, 2009.

Peptides are relatively short amino acid chains that occur either as separate molecules or as building blocks for proteins. Apart from their significance in analyzing proteins, peptides themselves may have various distinct chemical structures that are *themselves* related to different molecular functions. These functions, such as cleavage or binding, while being interesting in their own right, have also been shown to be important in areas such as biology, medicine, drug design, disease pathology, and nanotechnology Indeed, for more than a decade, researchers have sought computational techniques to rapidly identify peptides that are known to be, or can be, related to certain molecular functions.

The research in peptide classification is not new –indeed, a host of techniques have been proposed for *in silico* peptide classification[1]. In 1998, Cai and Chou [3] presented one of the pioneering works in this area. They classified 8-residue peptides and used artificial neural networks with 20 input nodes per residue, thus involving a total of 160 input nodes. In their work, each amino acid was encoded using 20 bits so that the 20 amino acids were encoded as $A = 100\ldots00, B = 010\ldots00, \ldots, Y = 000\ldots01$. Similarly, Zhao *et al.* [15] mapped the amino acid sequences of peptides directly into feature vectors and fed them into a Support Vector Machine (SVM). They, however, represented the amino acids by a set (more specifically, ten) of their biophysical properties, such as hydrophobicity or beta-structure preference, instead of an orthonormal representation, as advocated by [3]. By resorting to such a representation, they were eventually able to reduce the dimensionality of the input space by 50%. To further increase the information density of input vectors, Thomson *et al.* [13] used bio-basis artificial neural networks, which are a revision of radial-basis function networks, that use biological similarities rather than spatial distances. This work was subsequently enhanced by Trudgian and Yang [14] by optimizing the substitution matrices that are used to compute the latter biological similarities. Kim *et al.* [8] followed a rule-based approach to achieve results which were interpretable. It should be mentioned that there were also earlier studies based on the properties of quantitative matrices, binding motifs and hidden Markov models, which should really be treated as precursors to the results cited above. The differences between our results and those which use Hidden Markov Models (HMMs) will be clarified presently.

A completely different sequence representation technique was introduced in the area of protein fold recognition by Liao and Noble [9]. Liao and Noble represented protein sequences by their pairwise biological similarities, which were measured by ordinary sequence alignment algorithms. Subsequently, by considering *these* similarities as feature vectors, relatively simple classifiers were trained and successfully utilized for classifying and discriminating between different protein folds.

The primary intention in this study is to use a SVM-based classifier in achieving the classification and discrimination. However, rather than use distances, we shall advocate the use of a rigorous probabilistic model, namely one which has

[1] The review and bibliography presented here is necessarily brief. A more detailed review is found in [1].

been proven to be both optimal and to attain the information theoretic bound. Indeed, in this study, we combine the strategy of Liao and Noble (i.e., to use pairwise SVM classifiers) with a probabilistic similarity metric, and to successfully classify peptides. Observe that, instead of resorting to the alignment scores, we quantify the similarity by means of their Optimal and Information Theoretic (OIT) garbling probabilities as described by Oommen and Kashyap [11]. The latter OIT garbling probability is the probability of obtaining a sequence Y from a sequence U based on the OIT mutation model, whose properties will be clarified later. One clear difference between the alignment scores and OIT garbling probabilities is that whereas an alignment score considers only the *shortest* path between two sequences, the OIT garbling probabilities covers all possible paths. Furthermore, since it assigns a probability mass to every possible path (i.e., possible garbling operations), it contains more information about the similarity between the two sequences.

It is pertinent to mention that a similar transition probability measurement based on HMMs was earlier proposed by Bucher and Hofman [2]. Indeed, since then, HMM-based similarity metrics have been used in many biological applications. The difference between our work and the ones which use HMMs can be, in all brevity stated as follows: Unlike the latter, the OIT model permits non-Geometric-based distributions for the number of insertions occurring in any sequence of mutations [1,11]. Additionally, the superiority of OIT model, say Π^*, to "distance-based" approaches are (a) Π^* is Functionally Complete because it comprehensively considers all the ways by which U can be mutated into Y using the three elementary Substitutions, Insertions and Deletions (SID) operations, (b) The distributions and the parameters involved for the various garbling operations in Π^* can be completely arbitrary, (c) Π^* captures the scenarios in which the probability of a particular string U being transformed into another string Y, is arbitrarily small, (d) For a given U, the length of Y is a random variable whose distribution does not necessarily have to be a mixture of Geometric distributions, and (e) If the input U is itself an element of a dictionary, and the OIT channel is used to model the noisy channel, the technique for computing the probability $\Pr[Y|U]$ can be utilized in a Bayesian way to compute the *a posteriori* probabilities, and thus yield an optimal, minimum probability of error pattern classification rule. Most importantly, however, in both the Bayesian and non-Bayesian approaches, the OIT model actually attains the information theoretic bound for recognition accuracy when compared with all the other models which have the same underlying garbling philosophy These issues are also clarified in greater detail in [1,11].

We have tested our solution, the OIT_SVM, which involves the combination of the SVM-pairwise and the OIT model, on two peptide classification problems, namely the *HIV-1 Protease Cleavage* site and the *T-cell Epitope* prediction problems. Both of these problems are closely related to pharmacological research work that has been the focus of a variety of computational approaches [3,8,13,14,15]. The results, which we present in a subsequent section, indicate that our solution paradigm leads to an extremely good classification performance.

2 Modeling – The String Generation Process

We now describe the model by which a string Y is generated given an input string $U \in A^*$, where A is the alphabet under consideration, and ξ and λ are the input and output null symbols, respectively.

First of all, we assume that the model utilizes a probability distribution G over the set of positive integers. The random variable in this case is referred to as Z, and is the number of insertions that are performed in the mutating process. G is called the *Quantified* Insertion Distribution, and in the most general case, can be conditioned on the input string U. The quantity $G\left(z|U\right)$ is the probability that $Z = z$ given that U is the input word. Thus, G has to satisfy the following constraint:

$$\sum_{z \geq 0} G\left(z|U\right) = 1. \tag{1}$$

The second distribution that the model utilizes is the probability distribution Q over the alphabet under consideration. Q is called the *Qualified* Insertion Distribution. The quantity $Q\left(a\right)$ is the probability that $a \in A$ will be the inserted symbol conditioned on the fact that an insertion operation is to be performed. Note that Q has to satisfy the following constraint:

$$\sum_{a \in A} Q\left(a\right) = 1. \tag{2}$$

Apart from G and Q, another distribution that the model utilizes is a probability distribution S over $A \times (A \cup \{\lambda\})$, where λ is the output null symbol. S is called the Substitution and Deletion Distribution. The quantity $S\left(b|a\right)$ is the conditional probability that the given symbol $a \in A$ in the input string is mutated by a stochastic substitution or deletion –in which case it will be transformed into a symbol $b \in (A \cup \{\lambda\})$. Hence, $S\left(c|a\right)$ is the conditional probability of $a \in A$ being substituted for by $c \in A$, and analogously, $S\left(\lambda|a\right)$ is the conditional probability of $a \in A$ being deleted. Observe that S has to satisfy the following constraint for all $a \in A$:

$$\sum_{b \in (A \cup \{\lambda\})} S\left(b|a\right) = 1. \tag{3}$$

Using the above distributions we now informally describe the OIT model for the garbling mechanism (or equivalently, the noisy string generation process). Let $|U| = N$. Using the distribution G, the generatorfirst randomly determines the number of symbols to be inserted. Let Z be random variable denoting the number of insertions that are to be inserted in the mutation. Based on the output of the random number generator, let us assume that Z takes the value z. The algorithm then determines the position of the insertions among the individual symbols of U. This is done by randomly generating an input edit sequence $U' \in (A \cup \{\xi\})^*$. We assume that all the possible strings are equally likely.

Note that the positions of the symbol ξ in U' represents the positions where symbols will be inserted into U. The non-ξ symbols in U' are now substituted for or deleted using the distribution S. Finally, the occurrences of ξ are transformed independently into the individual symbols of the alphabet using the distribution Q. This defines the model completely. The process followed by the model, and its graphical display, are formally included in the unabridged version of this paper, and omitted here in the interest of brevity [1]. The theoretical properties of the OIT model can be found in [11].

3 Proposed Methodology

In this section, we provide the explicit details of the syntactic probabilities of the OIT model, and also explain the way by which we utilize it together with the SVM-pairwise scheme for peptide classification.

For a mutation consisting of random SID operations as per the OIT model, Oommen and Kashyap [11] have derived the syntactic probability of obtaining the sequence $Y = y_1 y_2 \dots y_M$, from the sequence $U = u_1 u_2 \dots u_N$ as:

$$P(Y \mid U) = \sum_{z=\max\{0,M-N\}}^{M} \frac{G(z)\ N!\, z!}{(N+z)!} \sum_{U'} \sum_{Y'} \prod_{i=1}^{N+z} p\left(y'_i \mid u'_i\right),$$

where $G(z)$ is the probability of inserting z elements into U, and $p\left(y'_i \mid u'_i\right)$ is the probability of substituting the symbol element u'_i with the symbol element y'_i. Observe that in the above,

$$u'_i = \xi \Rightarrow y'_i \neq \lambda, \text{ and } y'_i = \lambda \Rightarrow u'_i \neq \xi.$$

The sum over the strings $U' = u'_1 u'_2 \dots u'_{N+z}$ and $Y' = y'_1 y'_2 \dots y'_{N+z}$ (of the same length), represent the sum over all possible pairs of strings U' and Y' of equal length $N + z$, generated by inserting ξ's into random positions in string U, and λ's into random positions in strings Y respectively, and which are to represent the insertion and the deletion operations respectively. Although this requires a summation over a combinatorially large number of elements (represented by U' and Y'), Oommen and Kashyap [11] have shown that this can be computed in an extremely efficient manner in cubic time, i.e., with complexity $O\left(M \cdot N \cdot \min\{M, N\}\right)$. Based on the work of Oommen and Kashyap [11], we have programmed our own toolkit to efficiently compute the syntactic probabilities between two arbitrary sequences, and adapted it to this particular domain.

Since the OIT model essentially requires three "parameters" namely, S for the Substitution/Deletion probabilities, Q, for the insertion distribution, and G, we list the issues crucial to our solution:

1. The input and output alphabets in our application domain consist of *twenty* amino acids and one gap element, which for the input strings is the null symbol, ξ, representing an inserted element, and for output strings is the null symbol, λ, representing a deleted element.
2. The substitution of an amino acid with another corresponds to a series of mutations in the biological context. Based on this premise, we have computed

our substitution probabilities on the mutation probability matrix referred to as PAM1 derived by Dayhoff *et al.* [5]. PAM1 is a 20 × 20 matrix, M, where each cell $\mathtt{m}_{ij}$ corresponds to the probability of replacing amino acid i with amino acid j after 1% of the amino acids are replaced. It is possible to generate matrices for a series of longer mutations using successive multiplications of PAM1, and thus, for example, PAM250 is equal to PAM249×PAM1 [5].

3. The first major deviation from the traditional PAM matrices involves the operation of deletion. Observe that PAM matrices generally do not specify deletion probabilities for amino acids. As opposed to this, the OIT model of Oommen and Kashyap [11] suggests that an element can be deleted (substituted by λ) as well as substituted by another element. In this vein, we advocate that the matrix PAM1 be extended by appending another column for λ, where the value Δ is assigned to the deletion probabilities of amino acids, and where each row is normalized to satisfy the probability constraint:

$$\sum_{y \in A \cup \{\lambda\}} p\left(y \mid u\right) = 1, \tag{4}$$

   where A is the set of all amino acids, and u is the amino acid corresponding to the row.
4. There is no standard method of determining the deletion probabilities of amino acids. Comparing the widely-used gap penalties as per [12] to the $log - odd$ PAM matrices, we opted to use $\Delta = 0.0001$. The question of how to optimally determine Δ is open, and we are currently considering how it can be obtained from a training phase using known Input/Output patterns.
5. The second major deviation from utilizing the traditional PAM matrices involves the operation of insertion. As in the case of deletion, we propose to extend the new PAM matrix by appending a row for ξ and assigned to $p\left(y \mid \xi\right)$ (i.e. the probability that a newly inserted amino acid is y) the relative frequency of observing y, $f\left(y\right)$. In our experiments, the relative frequencies were computed in a maximum likelihood manner by evaluating the limit of the PAMn matrix as n goes to infinity, i.e., as each row of the limiting matrix converges to $f\left(y\right)$. Finally, the remaining cell of our extended PAM matrix, $p\left(\lambda \mid \xi\right)$, is, by definition, equal to zero. The resulting matrix has been referred to as the OIT_PAM matrix, and is a 21 × 21 matrix. Table 1 gives a typical OIT_PAM matrix for the amino acid application domain. Observe that as in the case of the traditional PAM matrices, it is possible to derive higher order OIT_PAM matrices for longer mutation sequences by multiplying OIT_PAM1 by itself. In our work, we have experimented with OIT_PAM matrices of different orders to observe the effect of different assumptions that concern evolutionary distances.
6. The final parameter of the OIT model involves the Quantified Insertion distribution, $G\left(z\right)$, which specifies the probability that the number of insertions during the mutation is z. In our experiments, we have assumed that the probability of inserting an amino acid during a single PAM mutation is equal to the deletion probability of an amino acid, Δ. This assumption leads

to the conclusion that for longer mutation series, the insertion distribution converges to a Poisson distribution such that

$$G(z) = \texttt{Poisson}(z; n\Delta) = \frac{(n\Delta)^z e^{-\Delta n}}{z!}, \tag{5}$$

where n is the number of PAMs (i.e. the length of the mutation series). In other words, we have currently used $\texttt{Poisson}(z; n\Delta)$ as the insertion distribution whenever we use $\texttt{OIT\_PAM}n$ as the substitution probability matrix.

7. Using the OIT model and the parameters assigned as described above, a classification methodology based on the SVM-pairwise scheme proposed by Liao and Noble [9] was devised. This will be explained in the next subsection.

Having explained how the OIT-based scheme works, we shall now also present the results obtained from our experiments.

4 Experimental Results and Discussions

4.1 Experimental Setup

In our experiments, we used two peptide classification data sets, which are accepted as benchmark sets. The first one, referred to as *HIV*, was produced for the *HIV-1 Protease Cleavage* sites prediction problem by Kim *et al.* [8]. This set contains 754 8-residue peptides with 396 positives and 358 negatives. The second data set, referred to as *TCL*, was produced for the *T-cell Epitope* prediction problem by Zhao *et al.* [15], and it contains 203 10-residue peptides of which 36 were positives and 167 were negatives.

As mentioned earlier, our classification scheme was based on the SVM-pairwise scheme proposed by Liao and Noble [9] to detect remote evolutionary relationships between proteins. According to our scheme, m representative peptides were chosen *a priori* from the training set. Subsequently, for each instance, an m-dimensional vector of scores was computed by comparing the instance to the representatives. The classifiers were trained and tested with these feature vectors.

As a computational convenience, we used the logarithm of the OIT probability as the measure of similarity because the logarithm is a monotonic function, and it turns out that this can be computed more efficiently than the original OIT probabilities. To compare the performance of the OIT_SVM to the standard measures, we have also used the Needleman-Wunsch (NW) alignment score [10], which is a commonly used sequence comparison method in bioinformatics, to achieve an analogous classification. Our representative peptides were chosen to be the positive training instances, and in each case, we used eight different substitution matrices with mutation lengths 10, 50, 100, 200, 250, 300, 400 and 500.

Each feature set was tested on a SVM classifier with a linear kernel. A preliminary evaluation showed that the SVM with a linear kernel performs slightly better than the SVM with a *radial*-basis kernel on all the feature sets. Based on this observation, we fixed the classifier prior to the experiments, and merely focused on the comparison of feature sets themselves. In the testing phase, we estimated the performance of different methods by means of a cross-validation process. To do this, we divided the *HIV* data set into ten partitions and the *TCL* data set, which is rather small, into five partitions as was done in [8] and [15] respectively. In our case, we chose to not divide the *TCL* data set into more than five partitions because the number of positive examples was too low, and it consequently prevented us from providing the necessary variation across the partitions. This choice also rendered our results to be compatible with the results of [15]. Finally, we also ensured the preservation of the ratio of positive and negative instances across the partitions. All the classification and performance estimations were performed on the Mathworks MATLAB [7] system with the help of the PRTools 4.1 pattern recognition toolbox [6] and the LIBSVM 2.88 support vector machine library [4].

4.2 Experimental Results and Discussions

The performance of the OIT-based features were compared to the scores obtained by a Needleman-Wunsch (NW) alignment strategy. In each case, and for each of the experiments, we recorded the area under the ROC (AUC), the Accuracy (Acc), the Sensitivity (Sens) and the Positive Predictive Value (PPV). Tables 2 and 3 show the averaged values and average widths of the 95% confidence intervals for the *HIV* and *TCL* data sets, respectively. It is worth mentioning that the OIT-based scheme is uniformly superior to the NW-based scheme, and in some cases the superiority is categorically marked –for example, whereas the best accuracy for the NW-based method is 85.7%, the corresponding best accuracy for the OIT-based scheme is 91.7%.

Also note that the 95% confidence intervals are generally wider for the *TCL* dataset than they are for the *HIV* dataset. This is because the cross validation was performed through a five-fold strategy on the former, and through a ten-fold strategy on the latter.

For the *HIV* data set, [8] report accuracies for ten different methods, and our OIT-based method outperforms nine of them, while the accuracy of the tenth is marginally better. With regard to the *TCL* data set, it should be mentioned that the OIT_SVM leads to better results than those reported by [15] –when it concerns *any* performance criterion.

The behaviors of the two methods for different score matrices can be seen in Figures 1.These two figures display how the AUCs vary as the assumption of the mutation lengths increases from 10 PAMs to 500 PAMs. The reader will observe that for the *HIV* data set, both the OIT and the NW reach their highest performances between 100 and 300 PAMs. For the *TCL* data set, however, the NW prefers `PAM400`. When it concerns the means of the average AUCs, it should be mentioned that the OIT outperforms the NW even in its worst cases. Table 4 records the t-test results that validate this observation. Also, the average widths of the confidence intervals point to the conclusion that the OIT leads to more robust classifications than the NW.

Table 1. The `Log-OIT_PAM1` matrix used for the OIT model. Each element $\mathtt{M}_{i,j}$ is equal to the logarithm of the probability associated with the event of replacing the i^{th} element with j^{th} element. The symbols ξ and λ represent the insertion and deletion of an element of the alphabet, respectively. Please see Section 3 for more details on the `Log-OIT_PAM1` matrix.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	λ
A	-0.01	-9.21	-7.82	-7.42	-9.21	-8.11	-6.91	-6.17	-9.21	-8.52	-8.11	-8.52	-9.21	-9.21	-6.65	-5.88	-6.12	-36.04	-9.21	-6.65	-9.21
R	-8.52	-0.01	-9.21	-36.04	-9.21	-7.01	-36.04	-9.21	-7.13	-8.52	-9.21	-5.60	-9.21	-9.21	-7.60	-6.81	-8.52	-8.52	-36.04	-8.52	-9.21
N	-7.01	-9.21	-0.02	-5.47	-36.04	-7.82	-7.26	-6.73	-6.32	-8.11	-8.11	-5.99	-36.04	-9.21	-8.52	-5.68	-6.65	-36.04	-8.11	-9.21	-9.21
D	-6.91	-36.04	-5.63	-0.01	-36.04	-7.60	-5.19	-6.81	-8.11	-9.21	-36.04	-7.42	-36.04	-36.04	-9.21	-7.26	-7.82	-36.04	-36.04	-9.21	-9.21
C	-8.11	-9.21	-36.04	-36.04	0.00	-36.04	-36.04	-9.21	-9.21	-8.52	-36.04	-36.04	-36.04	-36.04	-9.21	-6.81	-9.21	-36.04	-8.11	-8.11	-9.21
Q	-7.13	-6.91	-7.82	-7.42	-36.04	-0.01	-5.66	-8.11	-6.21	-9.21	-7.42	-6.73	-8.52	-36.04	-7.13	-7.82	-8.11	-36.04	-36.04	-8.52	-9.21
E	-6.38	-36.04	-7.42	-5.24	-36.04	-5.91	-0.01	-7.26	-9.21	-8.52	-9.21	-7.26	-36.04	-36.04	-8.11	-7.42	-8.52	-36.04	-9.21	-8.52	-9.21
G	-6.17	-36.04	-7.42	-7.42	-36.04	-9.21	-7.82	-0.01	-36.04	-36.04	-9.21	-8.52	-36.04	-9.21	-8.52	-6.44	-8.52	-36.04	-36.04	-8.11	-9.21
H	-8.52	-6.91	-6.17	-7.82	-9.21	-6.07	-8.52	-9.21	-0.01	-36.04	-7.82	-8.52	-36.04	-8.52	-7.60	-8.52	-9.21	-36.04	-7.82	-8.11	-9.21
I	-7.42	-8.11	-8.11	-9.21	-9.21	-9.21	-8.11	-36.04	-36.04	-0.01	-6.12	-7.82	-7.60	-7.13	-9.21	-8.52	-6.81	-36.04	-9.21	-5.17	-9.21
L	-7.82	-9.21	-9.21	-36.04	-36.04	-8.11	-9.21	-9.21	-9.21	-7.01	-0.01	-9.21	-7.13	-7.42	-8.52	-9.21	-8.52	-36.04	-9.21	-6.81	-9.21
K	-8.52	-6.27	-6.65	-8.11	-36.04	-7.42	-7.82	-8.52	-9.21	-8.52	-8.52	-0.01	-7.82	-36.04	-8.52	-7.26	-7.13	-36.04	-36.04	-9.21	-9.21
M	-7.42	-7.82	-36.04	-36.04	-36.04	-7.82	-9.21	-9.21	-36.04	-6.73	-5.40	-6.21	-0.01	-7.82	-9.21	-7.82	-7.42	-36.04	-36.04	-6.38	-9.21
F	-8.52	-9.21	-9.21	-36.04	-36.04	-36.04	-36.04	-9.21	-8.52	-7.26	-6.65	-36.04	-9.21	-0.01	-9.21	-8.11	-9.21	-9.21	-6.17	-9.21	-9.21
P	-6.12	-7.82	-8.52	-9.21	-9.21	-7.42	-8.11	-8.11	-8.11	-36.04	-8.11	-8.11	-36.04	-36.04	-0.01	-6.38	-7.60	-36.04	-36.04	-8.11	-9.21
S	-5.66	-7.42	-6.21	-7.60	-7.60	-8.52	-7.82	-6.17	-9.21	-9.21	-9.21	-7.13	-9.21	-8.52	-6.73	-0.02	-5.74	-9.21	-9.21	-8.52	-9.21
T	-5.74	-9.21	-7.01	-8.11	-9.21	-8.52	-8.52	-8.11	-9.21	-7.26	-8.11	-6.81	-8.52	-9.21	-7.82	-5.57	-0.01	-36.04	-9.21	-6.91	-9.21
W	-36.04	-7.13	-9.21	-36.04	-36.04	-36.04	-36.04	-36.04	-9.21	-36.04	-7.82	-36.04	-36.04	-8.11	-36.04	-7.60	-36.04	0.00	-8.52	-36.04	-9.21
Y	-8.52	-36.04	-7.82	-36.04	-8.11	-36.04	-9.21	-36.04	-7.82	-9.21	-8.52	-9.21	-36.04	-5.88	-36.04	-8.52	-8.52	-9.21	-0.01	-8.52	-9.21
V	-6.32	-9.21	-9.21	-9.21	-8.52	-9.21	-8.52	-7.60	-9.21	-5.71	-6.50	-9.21	-7.82	-36.04	-8.52	-8.52	-7.01	-36.04	-9.21	-0.01	-9.21
ξ	-2.43	-3.21	-3.20	-3.04	-3.43	-3.27	-2.99	-2.41	-3.42	-3.33	-2.46	-2.54	-4.21	-3.19	-2.96	-2.66	-2.84	-4.68	-3.45	-2.76	$-\infty$

Table 2. The performance measurements for the *HIV* data set using the OIT and NW metrics. The highest value over each column is shown in bold. The last row displays the average widths of the 95% confidence intervals (Avg. w) for each measurement.

(O)PAM	OIT				NW			
	AUC	Acc	Sens	PPV	AUC	Acc	Sens	PPV
10	0.948	0.887	0.863	0.884	0.906	0.839	0.821	0.837
50	0.962	0.902	0.891	0.904	0.909	0.849	0.841	0.843
100	0.968	**0.917**	**0.897**	0.927	0.917	0.846	**0.846**	0.833
200	**0.969**	0.911	0.877	0.932	**0.927**	**0.857**	0.833	**0.862**
250	0.965	0.913	0.874	0.938	0.925	0.853	0.830	0.857
300	0.965	0.911	0.863	**0.948**	0.921	0.849	0.829	0.852
400	0.958	0.901	0.849	0.937	0.912	0.849	0.838	0.848
500	0.949	0.893	0.830	0.938	0.924	0.846	0.813	0.859
Avg. w	0.011	0.018	0.037	0.021	0.019	0.025	0.040	0.029

Table 3. The performance measurements for the *TCL* data set using the OIT and NW metrics. The highest value over each column is shown in bold. The last row displays the average widths of the 95% confidence intervals (Avg. w) for each measurement.

(O)PAM	OIT				NW			
	AUC	Acc	Sens	PPV	AUC	Acc	Sens	PPV
10	0.918	0.852	0.922	0.901	0.883	0.837	**0.928**	0.882
50	0.937	0.872	0.934	0.912	0.892	0.842	0.922	0.891
100	0.943	0.882	0.929	0.928	0.889	0.847	0.922	0.895
200	**0.947**	0.897	0.940	0.935	0.889	0.853	0.905	0.917
250	0.944	**0.902**	**0.946**	**0.936**	0.885	0.853	0.893	0.927
300	0.945	0.887	0.940	0.924	0.895	0.852	0.916	0.905
400	0.939	0.887	**0.946**	0.919	**0.904**	**0.867**	0.911	**0.928**
500	0.936	0.882	0.929	0.928	0.819	0.793	0.881	0.871
Avg. w	0.016	0.023	0.022	0.020	0.028	0.030	0.041	0.021

Table 4. The t-test results for the 1% significance level comparing the AUC values of the OIT and NW based schemes

(O)PAM	HIV		TCL	
	OIT > NW	p-value	OIT > NW	p-value
10	no	0.013	no	0.018
50	**yes**	0.001	no	0.025
100	**yes**	<0.001	no	0.047
200	**yes**	<0.001	no	0.014
250	**yes**	<0.001	**yes**	<0.001
300	**yes**	<0.001	no	0.015
400	no	0.012	**yes**	0.001
500	no	0.014	**yes**	0.001

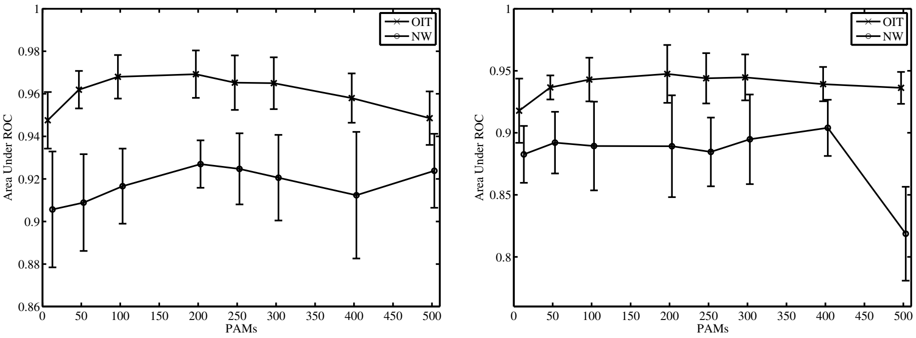

Fig. 1. The figure on the left displays the behavior of the OIT and NW similarity metrics on the *HIV* data set when the mutation length assumption changes between 10 PAMs and 500 PAMs. The figure on the right displays the corresponding behavior of the OIT and NW similarity metrics for the *TCL* data set. In each case, the error bars display the respective 95% confidence intervals.

5 Conclusions

In this paper, we have considered the problem of classifying peptides using syntactic pattern recognition methodologies. Unlike the traditional distance-based or Markovian methods, we have considered how the pattern recognition can be achieved by using the Optimal and Information Theoretic (OIT) model of Oommen and Kashyap [11]. We have shown that one can model the differences between the compared strings as a mutation model consisting of random SID operations which obeys a OIT model. Consequently, by using the probability measure obtained from the OIT model as a pairwise *similarity* metric, we have devised a Support Vector Machine (SVM)-based peptide classifier, referred to as OIT_SVM. The classifier has been tested for eight different "substitution" matrices and for two different data sets, namely, the *HIV-1 Protease Cleavage* sites and the *T-cell Epitopes*, and the results obtained categorically demonstrate that the OIT model performs significantly better than the one which uses a Needleman-Wunsch sequence alignment score. Further, when combined with a SVM, it leads to, probably, the best peptide classification method available.

The main drawback of of the OIT method is its higher time complexity. Otherwise, the avenues for future work include the learning of the PAM matrices using maximum likelihood or Bayesian methods. The use of an OIT model for other bioinformatic pattern recognition problems remains open.

Acknowledgments

The first and third authors, Aygün and Cataltepe, were partially supported by TÜBİTAK, The Scientific and Technological Research Council of Turkey. The second author, Oommen, was partially supported by NSERC, the Natural Sciences and Engineering Research Council of Canada.

References

1. Aygün, E., Oommen, B.J., Cataltepe, Z.: Peptide Classification Using Optimal and Information Theoretic Syntactic Modeling (submitted for publication)
2. Bucher, P., Hofmann, K.: A sequence similarity search algorithm based on a probabilistic interpretation of an alignment scoring system. In: Proceedings of the Conference on Intelligent Systems for Molecular Biology, pp. 44–51 (1996)
3. Cai, Y.D., Chou, K.C.: Artificial neural network model for predicting HIV protease cleavage sites in protein. Advances in Engineering Software 29(2), 119–128 (1998)
4. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), `http://www.csie.ntu.edu.tw/~cjlin/libsvm`
5. Dayhoff, M., Schwartz, R., Orcutt, B.: A model of evolutionary change in proteins. Atlas of Protein Sequence and Structure 5(suppl. 3), 345–352 (1978)
6. Duin, R.P.W., Juszczak, P., Paclik, P., Pekalska, E., de Ridder, D., Tax, D.M.J.: PRTools, a Matlab Toolbox for Pattern Recognition. Delft University of Technology (2004)
7. Guide, M.R.: The MathWorks. Inc., Natick, MA (1998)
8. Kim, H., Zhang, Y., Heo, Y.S., Oh, H.B., Chen, S.S.: Specificity rule discovery in HIV-1 protease cleavage site analysis. Computational Biology and Chemistry 32(1), 71–78 (2008)
9. Liao, L., Noble, W.S.: Combining Pairwise Sequence Similarity and Support Vector Machines for Detecting Remote Protein Evolutionary and Structural Relationships. Journal of Computational Biology 10(6), 857–868 (2003)
10. Needleman, S.B., Wunsch, C.D.: A general method applicable to the search for similarities in the ammo acid sequence of two proteins. J. Mol. Biol. 48(3), 443–453 (1970)
11. Oommen, B.J., Kashyap, R.L.: A formal theory for optimal and information theoretic syntactic pattern recognition. Pattern Recognition 31(8), 1159–1177 (1998)
12. Tatusova, T.A., Madden, T.L.: BLAST 2 Sequences, a new tool for comparing protein and nucleotide sequences. FEMS Microbiology Letters 174(2), 247–250 (1999)
13. Thomson, R., Hodgman, T.C., Yang, Z.R., Doyle, A.K.: Characterizing proteolytic cleavage site activity using bio-basis function neural networks. Bioinformatics 19(14), 1741–1747 (2003)
14. Trudgian, D.C., Yang, Z.R.: Substitution Matrix Optimisation for Peptide Classification. In: Marchiori, E., Moore, J.H., Rajapakse, J.C. (eds.) EvoBIO 2007. LNCS, vol. 4447, pp. 291–300. Springer, Heidelberg (2007)
15. Zhao, Y., Pinilla, C., Valmori, D., Martin, R., Simon, R.: Application of support vector machines for T-cell epitopes prediction. Bioinformatics 19(15), 1978–1984 (2003)