Pairwise Protein Substring Alignment With Latent Semantic Analysis and Support Vector Machines To Detect Remote Protein Homology

Surayati Ismail[1] , Razib M. Othman[1,*], Shahreen Kasim[2], Rohayanti Hassan[1], Hishammuddin Asmuni[1], Jumail Taliba[1]

[1]*Laboratory of Computational Intelligence and Biotechnology, Universiti Teknologi Malaysia, 81310 UTM Skudai, MALAYSIA.*
surayatiismail@gmail.com, razib@utm.my, rohayanti@utm.my, hishamudin@utm.my, jumail@utm.my

[2]*Department of Web Technology, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, MALAYSIA.*
shahreen@uthm.edu.my
**Corresponding author*

Abstract

Remote protein homology detection has been widely used as a part of the analysis of protein structure and function. In this study, the good quality of protein feature vectors is the main aspect to detect remote protein homology; as it will assist discriminative classifier model to discriminate all the proteins into homologue or non-homologue members precisely. In order for the protein feature vectors to be characterized as having good quality, the feature vectors must contain high protein structural similarity information and are represented in low dimension which is free from any contaminated data. In this study, the contaminated data which originates from protein dataset was investigated. This contaminated data may prevent remote protein homology detection framework to produce the best representation of high protein structural similarity information in order to detect the homology of proteins. To reduce the contaminated data and extract high protein structural similarity information, some research has been done on the extraction of protein feature vectors and protein similarity. The extraction of protein feature vectors of good quality is believed could assist in getting better result for remote protein homology detection. Where, the good quality of protein feature vectors containing the useful protein similarity information and represent in low dimension will be used to identify protein family precisely by discriminative classifier model. Referring to this factor, a method which combines Protein Substring Scoring (PSS) and Pairwise Protein Substring Alignment (PPSA) from sequence comparison model, chi-square and Singular Value Decomposition (SVD) from generative model, and Support Vector Machine (SVM) as discriminative classifier model is introduced.

Keywords: *Remote Protein Homology Detection, Protein Substring Scoring, Pairwise Protein Substring Alignment, Latent Semantic Analysis, Support Vector Machines.*

1. Introduction

We would like to draw your attention to the fact that it is not possible to modify a paper in any way, once it has been published. This applies to both the printed book and the online version of the publication. Every detail, including the order of the names of the authors, should be checked before the paper is sent to the Volume Editors. Remote protein homology detection is one of the methods that have been used widely by researchers to manage protein sequences by classifying proteins into their respectively family [14]. Protein family classification is a technique which able to assist in giving clues to help cure any genetic diseases and to perform drug design [4, 5, 6]. Nowadays, many methods in remote protein homology detection have been developed. Every method has their own ways to produce the best result. Most methods focus on handling certain problem in order to achieve better result, such as handling hard-to-align proteins [13], improving the sensitivity in sequence comparison model to detect protein similarity [4], and handling complex classification [20].

Basically, remote protein homology detection can be divided into three models [16]: sequence comparison model, generative model and discriminative classifier model. In sequence comparison model, the example method such as Smith-Waterman algorithm is a well-known algorithm for pairwise sequence comparison because of its ability to produce more accurate results which prioritize efficiency [8]. Nowadays many methods apply Smith-Waterman algorithm such as Zaki and Deris [22], Mohseni-Zadeh et al. [17], and Liao and Noble [16]. Other sequence comparison models are BLAST [1] and FASTA [18], which prioritize accuracy in their methods. A generative model such as Hidden Markov Model (HMM) and Latent Semantic Analysis (LSA) have demonstrated remote protein homology detection with great success. HMM is capable to handle big protein dataset, whereas LSA is capable to handle high dimensional protein feature vectors. The discriminative classifiers algorithm such as Support Vector Machine (SVM) has been used to separate each given structural protein class from the 'rest of the world' [2]. SVM will discriminate positive and negative members with the appropriate kernel. Similar with other models, SVM also able to handle certain problems such as SVM-Fisher [12] which is developed to handle multi-domain, SVM-String-Scoring (SVM-SS, Zaki and Deris [22]) is developed to handle big dataset, and SVM-Pattern-LSA [7] is developed to handle noisy data. All of these methods have been successful in producing improved results as compared to the other methods.

The success of SVM in classification actually depends on the choice of the quality protein feature vectors to describe each similarity of protein [8]. In order to identify protein feature vectors of good quality, further effort is needed, which focuses on finding good representation of similarity between protein sequences. The major problems associated with the effort to find good representation of similarity between sequences are the lack of protein similarity information and high dimensional protein feature vectors caused by redundant and noisy data These are the problems which normally prevent SVM to produce precise result. In order to get a maximum margin that will lead to produce more homologue protein members, string kernel is applied in SVM; whereas to extract high protein similarity information, Protein Substring Scoring (PSS) and Pairwise Protein Substring Alignment (PPSA) are applied. These methods extract high protein similarity information by checking region by region of protein sequence. On the other hand, redundant and noisy data which caused high dimensional protein feature vectors is reduced by using chi-square and Singular Value Decomposition (SVD) respectively. Chi-square is the most effective feature selection method in document classification [21] whereas SVD [15] is an efficient feature extraction method. Example of methods that use SVD model are SVM-Ngram, SVM-Motif [8], and SVM-

Pattern-LSA [7]. These methods significantly improve the performance of remote protein homology detection.

In this paper, three models are combined to detect remote protein homology. The three models comprise of sequence comparison model, generative model, and discriminative classifier model. Sequence comparison model uses PSS to split the protein sequence into protein substrings in order to assist PPSA in extracting protein similarity based on sensitive and non-sensitive regions [22]; generative model presents chi-square to reduce redundant data and SVD to remove noisy data which at the same time extracts and represents protein similarity information in protein feature vectors; discriminative classifier model presents the SVM method to discriminate homologous and non-homologous proteins [3]. To measure the quality of the results, Receiver Operating Characteristics (ROC), Median Rate of False Positives (MRFP), and family by family comparison of ROC scores are used. The ROC score is used to normalize area under a curve that plots true positives against false positives of different possible thresholds for classification. MRFP calculates the number of false positives scoring as high as or better than the median scoring true positives. Family by family comparison of ROC score is a comparison of ROC score of every family. The comparison is between the numbers of true positives against false positives of the 54 families with different possible thresholds for classification. Experimental results have shown that the use of LSA method has successfully produced better results compared to the other methods such as SVM-Ngram, SVM-Pattern, SVM-Motif, SVM-Ngram-LSA, SVM-Motif-LSA, SVM-Pattern-LSA, SVM-Fisher, and SVM-String-Scoring.

2. Methods

SVM-SS-LSA as shown in Figure 1 is an enhancement method to detect remote protein homology. SVM-SS-LSA uses protein substring to check region by region of protein sequence in order to get the highest similarity information which is measured using PPSA method. The high similarity information will then be used to represent the good quality protein feature vectors in low dimension with less redundant and noisy data. In order to make protein feature vectors are represented in low dimension with less redundant and noisy data, chi-square and SVD from LSA is applied respectively. Chi-square algorithm [21] is used to reduce redundant data by removing any data based on chi-square independent value using distribution with one degree of freedom to judge extremeness whereas, SVD [12] is performed on the similarity information to remove noisy data and produce protein feature vectors. Then, the representation of good quality protein feature vectors in low dimension will be classified by SVM.

3. Dataset

In order to prove the performance of SVM-SS-LSA, standard evaluation datasets [16] are used. The datasets are taken from the SCOP (http://scop.mrc-lmb.cam.ac.uk/scop/) database version 1.53. These datasets are measured using E-value threshold of 10^{-25} to choose protein sequences. The algorithm yields 4352 distinct protein sequences grouped by families and superfamilies. The protein sequences within family and in the same superfamily are taken as positive training examples. Whereas, negative training examples are taken from outside the family's fold. Details about the complete datasets and the various families can be found at http://www.cs.columbia.edu/compbio/ svm-pairwise.

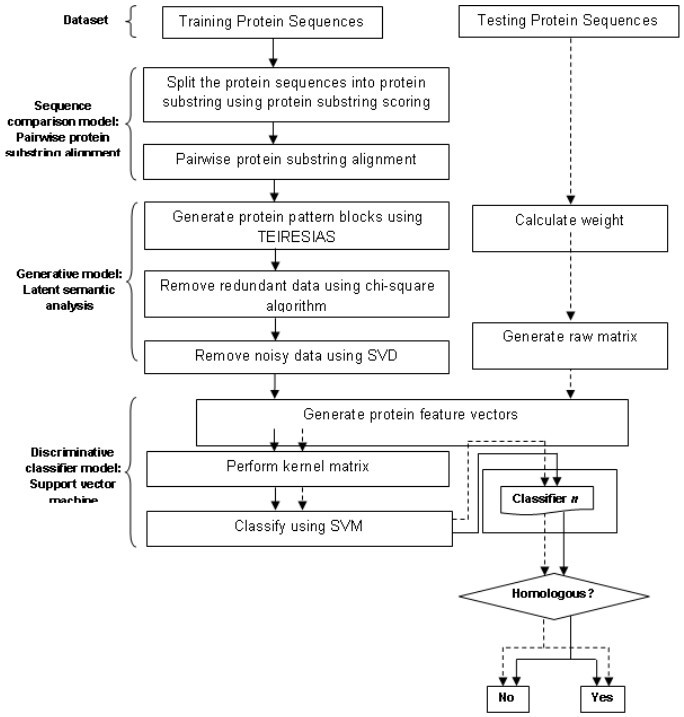


Figure 1. Overview of SVM-SS-LSA method

4. Sequence Comparison Model

4.1. Protein Substrings

The purpose of producing protein substrings is to get all the possible substrings of amino acid based on sensitive and non-sensitive regions. To get protein substrings, Protein Substring Scoring (PSS) method [22] is used. The method is implemented by simply sliding a window of a length $k > 1$ over the protein sequences. The method is illustrated as follows:

Example of protein sequence:
>d3sdha_ 1.1.1.1.1 Hemoglobin I {Ark clam (Scapharca inaequivalvis)}
SVYDAAAQLTADVKKDLRDSWKVIGSDKKGNGVALMTTLFADNQETI

Assume k =15,
SVYDAAAQLTADVKKDLRDSWKVIGSDKKGNGVALMTTLFADNQETI
SVYDAAAQLTADVKK**DLRDSWKVIGSDKKG**NGVALMTTLFADNQETI
SVYDAAAQLTADVKKDLRDSWKVIGSDKKG**NGVALMTTLFADNQETI**
The yields of 4 protein substrings are as follows:

1. **SVYDAAAQLTADVKK**
2. **DLRDSWKVIGSDKKG**
3. **NGVALMTTLFADNQETI**

For the last sliding of amino acid in the protein sequence, if the balance of amino acid $k<15$ it will be included in the previous protein substring.

4.2. Pairwise Protein Substring Alignment

Pairwise protein substring alignment (PPSA) is used to compare a protein substring with the aims of inferring structural, functional, and evolutionary relationships. In this study, the pairwise protein substring alignment based on Smith-Waterman algorithm is used. The Smith-Waterman is a one of the algorithms from sequence comparison model. The main purpose of PPSA is to determine an optimal alignment region by region between two protein substrings. The Smith-Waterman algorithm also known as dynamic programming used to determine distance or similarity between protein sequences. The distance is defined as the number of insertions, deletions, and replacements of characters. The Smith-Waterman algorithm is chosen because it produces accurate results based on sensitivity and selectivity aspects [1].

In this study, protein substrings produced by the PSS method is used as input. The PPSA process will begin by calculating the structural similarity score between protein substrings and searching for the optimal alignment by tracing back the similarity matrix. To make it more clearly, the process of the PPSA is shown as below:

For two protein substrings A and B, the length of A is g, $|A|= g$; the length of B is m, $|B| = m$; $V(d,e)$ is the optimal alignment score of two protein substrings $A[1]\ldots A[d]$ and $B[1]\ldots B[e]$, the calculation of $V(d,e)$ is defined as Equation 1 and Equation 2:

Initialization:

$$\begin{cases} V(d,0)=0, 0\le d\le g \\ V(0,e)=0, 0\le e\le m \end{cases}. \tag{1}$$

Recursion relation:

$$V(d,e)=\max\begin{cases} 0 \\ V(d-1,e-1)+(A[d],B[e]) \\ V(d-1,e)+(A[d],-) \\ V(d,e-1)+(-,B[e]) \end{cases}, 1\le d\le g, 1\le e\le g \ . \tag{2}$$

In these formulas, a "-" stands for a null character or gap; $V(d,0)$ stands for the result of comparing each character in A with a gap in B; the definition of $V(0,e)$ is the counterpart of the comparison of each character in B with a gap in A; and $(A[d],B[e])$ is the value of substitution matrix. While calculating the similarity matrix, the score of any matrix element $V(d,e)$ always depends on the score of three other elements: the up-left neighbor element $V(d-1,e-1)$, the left neighbor $V(d,e-1)$, and the up neighbor $V(d-1,e)$. Therefore, the calculation protein substring begins from the top-left element to the bottom-right element according to the direction as shown by the arrow. Through the observation of the similarity matrix calculation process, we found that for each clock cycle, every element on an anti-diagonal line marked with the same number could be calculated simultaneously, with the standing for the elements that could be calculated at the same time.

To further describe the level of similarity between two protein substrings, an affine gap model was introduced to the Smith-Waterman algorithm by Gotoh [11]. In the affine gap model, the gap is used to compensate for the insertion or deletion, to make the alignment more condensed in satisfying an expecting model. The gap is usually a consecutive null

character string in a sequence and should be as long as possible. In the affine gap model, the penalty score for the first gap is called gap open, and the penalty score for the following gaps are called the gap extension. According to the affine gap model, the formulas to calculate the similarity matrix are described below:
Initialization:

$$\cdot \begin{cases} V(d,0) = E(d,0) = 0, 0 \le d \le g \\ V(0,e) = F(0,e) = 0, 0 \le e \le m \end{cases} \tag{3}$$

Recursion relation:

$$V(d,e) = \max \begin{cases} 0 \\ E(d,e) \\ F(d,e) \\ V(d-1,e-1) + (A[d], B[e]) \end{cases} ,1 \le d \le g, 1 \le e \le m \ , \tag{4}$$

$$E(d,e) = \max \begin{cases} V(d,e-1) - \alpha \\ E(d,e-1) - \beta \end{cases} ,1 \le d \le g, 1 \le e \le m \ , \tag{5}$$

$$F(d,e) = \max \begin{cases} V(d-1,e) - \alpha \\ F(d-1,e) - \beta \end{cases} ,1 \le d \le g, 1 \le e \le m \ . \tag{6}$$

In these formulas, α stands for the gap open, and β stands for the gap extension. $E(d,e)$ and $F(d,e)$ are the maxima of the following two items: open a new gap or keep extending an existing gap.

The protein substrings produced from a split method are presented in the j-dimensional protein feature vectors where j is the total number of protein substrings. All the protein substrings are scored against the protein substrings of interest. The alignment scores are based on the notion of distance, counting the number of transformation from one protein substring is required to obtain the second protein substring. Transformations include substituting one character for another, inserting a string of characters, or deleting a string of characters.

After the alignment of all protein substrings are implemented, the protein structural similarity score for every alignment is need to be found. The calculation of the score is referred to the substitution matrix defined by the BLOSUM50 matrix. After the protein structural similarity scores for every alignment are calculated and the optimal alignment has been gained. The optimal alignment is gained from the highest protein structural similarity score which is chosen from several possibility alignments for two protein substrings and it will be used for the next process.

5. Generative Model

5.1. Protein Words

Like strings of letters and words in a text, protein sequences are linear chains of amino acids. The linear chains can be one of 20 residue standards of amino acids, labeled as (A C D E F G H I K L M N P Q R S T V W Y). The string of amino acids is called words of protein.

In LSA, each protein substring that belongs to a particular class is treated as a document which contains words of protein. This is composed of bags-of-X, where X can be any basic building blocks of protein substring [7]. Words of protein are not as clear as words in a natural language due to the absence of any language information. In this paper, a protein pattern is selected to produce the basic building blocks for the document. The protein pattern uses a 'don't care' symbol to represent any 20 residues of amino acids, example: protein pattern C. . . L H is present in both CVWLH and CKELH, where C, L and H are solid characters. The '.' symbol is denoted as 'don't care'. This delegation can be shown as $\sum \cup \{.\}$ where $\sum$ is the set of 20 residues of amino acids and {.} is any residue of amino acids and gaps. Every pattern has their score. The score is calculated using score matrices which associate weight with each pair of characters protein pattern in a string is as follows:

$$\sum \cup \sum (\sum \cup \{.\})^* \sum , \quad (7)$$

where a string on the alphabet $\sum \cup \{.\}$ starts and ends with solid character. The length of a string l (in $\sum^*$ or $(\sum \cup \{.\})^*$) is denoted by |l|, the letter at position *i* in *l* is denoted by *l*[*i*] and is shown as *l*= *l*[0] *l*[1]… *l*[|*l* |-1].

5.2. Protein Pattern Blocks

To extract protein patterns, the TEIRESIAS algorithm [19] is used. TEIRESIAS is implemented in two phases: scanning and convolution. In the scanning phase, elementary protein patterns with sufficient support are identified. The scanning process builds and located all <*L*,*G*> elementary protein patterns with support at least *K* distinct sequence. In this case, *L* is the number of acid amino residue in protein sub-pattern and *G* is length of protein sub-pattern. Protein sub-pattern is protein substring of protein pattern F, that it itself is the protein pattern. Then, all elementary protein patterns constitute the building blocks for the convolution phase. The convolution phase makes it easy to identify and discard non-maximal protein patterns. To generate maximal protein patterns, all-against-all approach is used. Where, all the elementary protein patterns are combined into progressively larger and larger protein patterns until maximal protein patterns are generated. In this study, TEIRESIAS algorithm is applied by using testing and training protein sequences. The parameter settings used were *L*=3, *G*=35, and *K*=7. Parameter *L* was set to 3 because this is the smallest value for which the benefits of convolution become apparent as the prefixes and suffixes used are non-trivial. *L*>3 will affect the performance of the algorithm by decreasing the convolution time while increasing the scanning time. For the parameter *G*, larger value has been tried but no more substantial protein patterns were discovered. Whereas, *K*=7 is suitable with the value of *L*=3 because there is subset of at least K input sequences exhibiting extensive degree of similarity. By following these parameters, the results show that a total of 75811 protein patterns were extracted. The extracted protein patterns contain redundant information and this raises a problem for machine learning algorithms to perform well in a high-dimensional feature space using high dimensional protein feature vectors. The following sections will provide a detailed explanation of the procedure.

5.3. Chi-square

With references to the previous problem, that redundant data exists in original dataset; it is desirable to reduce the dimension of the protein feature vectors by reducing redundant data.

This problem can be solved by using the feature selection method such as document frequency, information gain, mutual information, chi-square, and term strength [21]. But in this study, the chi-square algorithm is selected because it is one of the most effective feature selection methods in document classification [21].

In the chi-square x^2 measures, the lack of independence between a feature t and a classification category c can be compared to the chi-square distribution with one degree of freedom to judge extremes. The chi-square x^2 value of feature t relative to category c is defined as follows:

$$x^2(t,c) = \frac{N \bullet (A \bullet D - C \bullet B)^2}{(A+C) \bullet (B+D) \bullet (A+B) \bullet (C+D)} \tag{8}$$

where N is the total number of documents, A is the number of times t and co-occur B is the number of times t occurs without c, C is the number of times c occurs without t, and D is the number of times neither c nor t occurs.

The chi-square x^2 statistics has a natural value of zero if t and c are independent. Each category of chi-square x^2 is computed between each unique term in a training corpus within its category. Then, the category specific scores of each feature are combined into two scores as follows:

$$x^2_{avg}(t) = \sum_{i=1}^{m} P_r(c_i)\, x^2(t,c_i) \tag{9}$$

$$x^2_{\max}(t) = \max_m^{i=1} \left\{ x^2(t,c_i) \right\}. \tag{10}$$

In this method, the maximum feature value is used since its performance is better than the average value.

5.4. Singular Values Decomposition

SVD performed on the protein patterns can remove the noisy data and leading to reduce the dimensions of the protein feature vectors in order to get the good quality representation of protein feature vectors [9]. In order to remove noisy data from the protein patterns, all the protein patterns need to be changed to protein feature vectors. In this study, the protein pattern similarity is treated as words and the protein sequences are viewed as the documents. Through collecting the weight of each word in the document, the word-document matrix is constructed and then the LSA is performed on the matrix to produce the protein feature vectors, leading to noise removal. To extract the good quality protein feature vectors with less noisy data, SVD needs to decompose the protein patterns into three components.

$$W = USJ^T, \tag{11}$$

where S is the $R \times R$ diagonal matrix of singular values, U is the $C \times R$ matrix of eigenvectors derived from the protein pattern correlation matrix given by WW^T, and J is the $D \times R$ matrix of eigenvectors derived from the document-document correlation matrix given by W^TW. Matrix W is leading to the dimensionality reduction. This occurs because SVD is able to make the similar data appears more similar. In the reduced versions of U and J, the protein feature vectors are more similar. The protein feature vectors contain components ordered from the most to the least amount of variation accounted for in the original data. By deleting the protein feature vectors representing the dimensions which do not exhibit meaningful variation which is only the top $R\left(R \square \min(C,D)\right)$ dimensions for which the elements in S are

greater than threshold are considered for further processing, and then effectively eliminate the noisy data in the representation of protein feature vectors. Thus, the dimensions of matrices *U*, *S* and *J* are reduced to $C \times R$, $R \times$ R and $D \times R$, leading to data noise removal. By deleting protein feature vectors representing dimensions which do not exhibit meaningful variation, in the same time the noisy data in the representation of protein feature vectors effectively eliminate. Now the protein feature vectors are transformed into leading to the low dimensional protein feature vectors, and contain only the elements that account for the most significant correlations among protein patterns in the original dataset.

6. Discriminative Classifier Model

6.1. Support Vector Machines

Support Vector Machines (SVM) is a class of machine learning based on statistical learning and has shown excellent performance in practice. SVM addresses the general problems of learning by analysing a linear decision boundary to discriminate between positive and negative members of a given class of *n*-dimensional vectors. The basic idea of applying SVM can be stated by mapping the *n*-dimensional vectors into a feature space which is relevant with the selection of the kernel function. Then, SVM constructs a hyper-plane to fit into the linear or non-linear curve which then separates these two classes of vectors with the maximum margin of separation. A maximum margin or an optimal hyper-plane is needed to lead maximal generalization when predicting classification of unlabeled example.

The samples of training set E used in this method consists of input vectors (x_i, y_i) and they are shown as follows:

$$x_i \in R_d \, (i = 1, ..., n) \tag{12}$$

with corresponding labels

$$y_i \in \{+1, -1\} (i = 1, ..., n) \tag{13}$$

where R_d refers to input space, and $+1$ and -1 are used to stand respectively for the two classes.

If the analyses from the input consist of two-category target variables with two predictor variables, a linear classification rule f will be used by the pair (w, b) and is shown as follows:

$$f(x) = (w \bullet x_i) + b \tag{14}$$

where x is classified as positive if $f(x) > 0$ and $f(x) < 0$ for negative data. Geometrically, the decision boundary is a hyper plane

$$\{x \in R_d : w_0 \bullet x + b_0 = 0\}. \tag{15}$$

For analyses with more than two predictor variables, SVM needs to separate the points using non-linear classification. To do so, SVM maps the data by a function Φ into a higher dimensional space (feature space), and defines a separating hyper plane there. The kernels of the SVM are:

$$K(x, z) = (\Phi(x) \bullet \Phi(z)), \text{ for all } x, z \in R_d \tag{16}$$

which will be used to realize a non-linear mapping with a feature space. $K(x, z)$ is a symmetric function. The appropriate hyper-plane can be found by an SVM in the feature space that corresponds to a decision boundary in the input space. The concept of a kernel

mapping function is very powerful. It allows SVM models to perform separations of vectors even with very complex boundaries. All these capabilities make SVMs an attractive classification system.

In this study, the process to implement SVM begins with testing dataset which are vectorized in the same way as the training dataset. It will be fed into the classifier constructed for a given class to make separation between positive and negative members. The SVM assigns each protein in the testing dataset a discriminative score which indicates a predicted positive level of protein. The proteins with discriminative scores higher than the threshold zero are classified as positive members and the others as negative members. This process is iterated until all proteins are tested. In order to implement this process, the Gist.2.3 SVM package implemented by Liao and Noble (2003) is used. It is available at: http://bioinformatics.ubc.ca/gist /download.html.

7. Results and Discussion

The performance of the proposed method is compared with the current successful homology detection methods. This method is compared with other methods that have combined SVM with LSA (SVM-Pattern-LSA, SVM-Ngram-LSA, and SVM-Motif-LSA), SVM without LSA (SVM-Pattern, SVM-Ngram, and SVM-Motif), and SVM with kernel matrix (SVM-String-Scoring and SVM-Fisher). In order to assess the recognition performance of each method, testing is done on its ability to classify protein sequences into homologue or non-homologue members in the SCOP version 1.53. The dataset contains 54 families within at least 10 family members and 5 superfamily members outside of the family.

To measure the performance of the proposed method, an evaluation in terms of the average Receiver Operating Characteristics (ROC) and Median Rate of False Positives (MRFP) values over 54 experiments are summarized and presented in Table 1. The results have shown that the proposed method has performed better than the other methods. Refer to the Figure 2(a) and Figure 2(b) present the ranks of the ROC and MRFP scores. In each graph, a higher curve corresponds to the more accurate homology detection performance. Compared to ROC or MRFP, the SVM-SS-LSA method performs significantly better than the other methods. The results in Table 1, Figure 2(a), and Figure 2(b) have shown that the use of combination PSS, PPSA, LSA, and SVM has been successfully applied in the remote protein homology detection. The implementation of the four methods has helped to extract good quality protein feature vectors. The PSS and PPSA methods have assisted in extracting more structural similarity information between protein substrings where this information will be kept by the protein feature vectors. On the other hand LSA has been utilized were assisting to reduce the dimension of protein feature vectors by reducing the redundant and noisy data in the protein patterns and protein feature vectors. All these methods have contributed to produce good quality protein feature vectors. This makes SVM-SS-LSA a better option compared to other methods.

Another evaluation to show the performance of the proposed method is by using family by family comparison of the ROC. The results are plotted in Figure 3 and Figure 4. Figure 3 shows the performance comparisons between the SVM-SS-LSA and methods without LSA and Figure 4 presents the performance comparisons between the SVM-SS-LSA and methods with LSA. The SVM-SS-LSA performs better compared to the other methods that have applied SVM with LSA and SVM without LSA. In Figure 3 and Figure 4, the families in the right-bottom area mean that the method labeled by *y*-axis outperforms the method labeled by

x-axis on this family. The use of the SVM-SS-LSA method shows a better performance compared to the other methods. This is due to the usage of a chi-square and SVD. Chi-square is used to reduce the number of redundant data. The success of chi-square to reduce redundant data is achieved by the ability of the chi-square to select the most discriminative features of proteins by their average chi-square scores. Whereas, the SVD is used to reduce the high dimension of protein features vector by reducing the noisy data. SVD decomposes the matrix of structural similarity information into three matrices in order to choose the good quality protein feature vectors. In this procedure, the top $H\left(H \square \min\left(C,D\right)\right)$ dimensions for the elements in diagonal matrix *S* that are greater than the threshold are considered for further processing. Both methods are capable to produce best representation of protein feature vectors in low dimension with less redundant and noisy data. This success has contributed to extract the good quality protein feature vectors.

The success of method to classify protein sequences into their homologue or non-homologue members of protein is dependent on the protein similarity score that represents the value of similarity structure of protein sequences. Almost all of the protein substrings present high similarity scores due to the use of Smith-Waterman algorithm that is known to be the most sensitive pairwise comparison method.

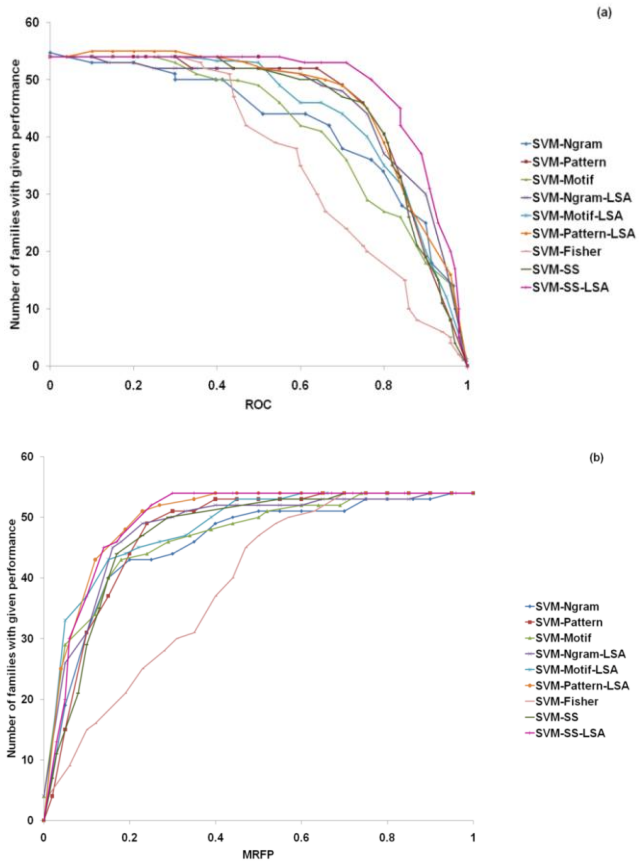
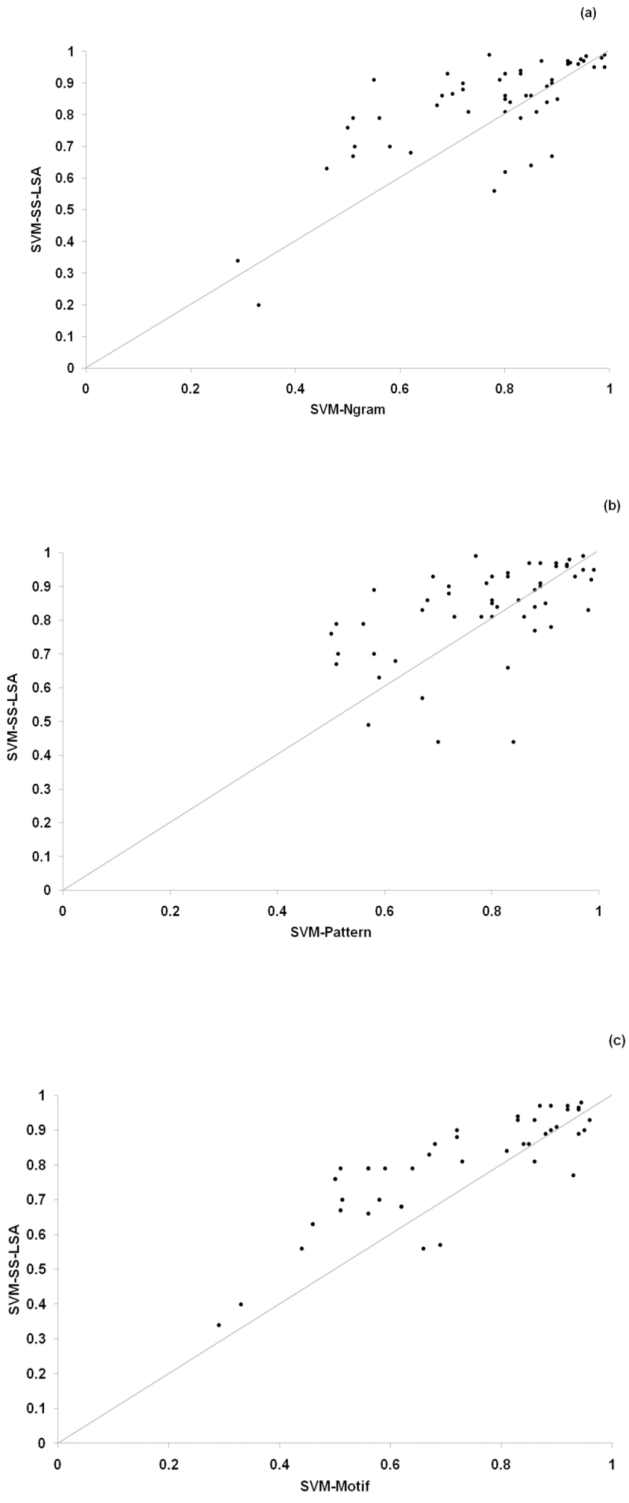


Figure 2. Relative performance of the nine homology detection methods. Each graph plots the total number of families for which a given method exceeds a score threshold. The top graph (a) uses ROC scores and the bottom graph (b) uses MRFP scores


(a)
SVM-SS-LSA
SVM-Ngram
(b)
SVM-SS-LSA
SVM-Pattern
(c)
SVM-SS-LSA
SVM-Motif

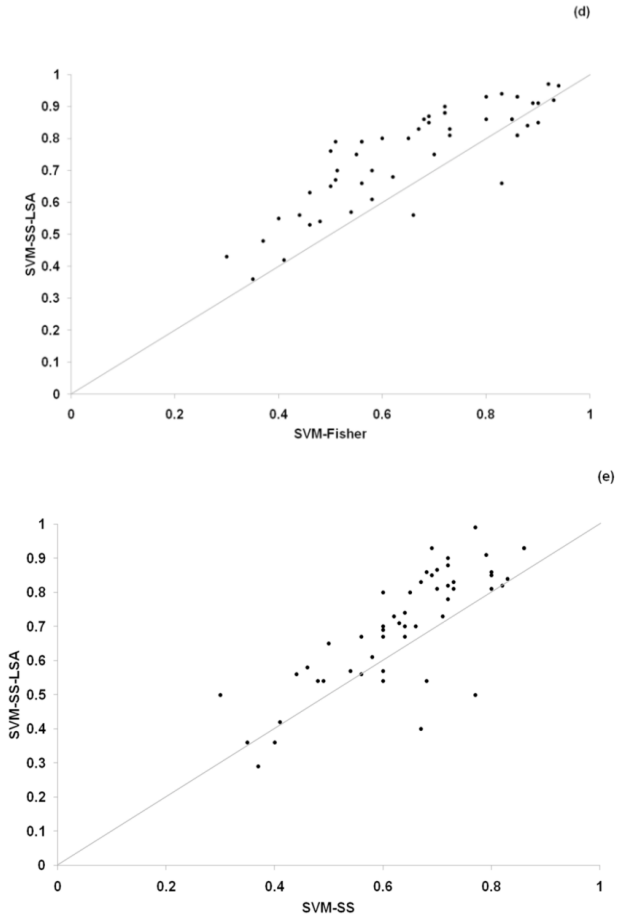
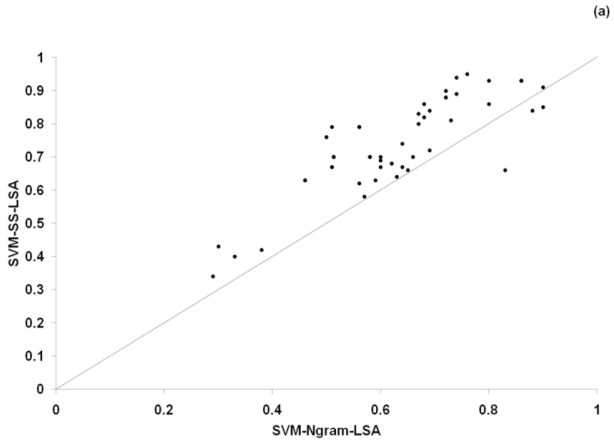


Figure 3. Family by family comparison of ROC based on the SVM-SS-LSA method and those without LSA. Each point on the graph corresponds to one of the 54 SCOP families. In each figure, the axes are ROC scores achieved by the two primary methods compared. Figure (a), (b), (c), (d), and (e) are based on SVM-Ngram, SVM-Pattern, SVM-Motif, SVM-Fisher, and SVM-SS



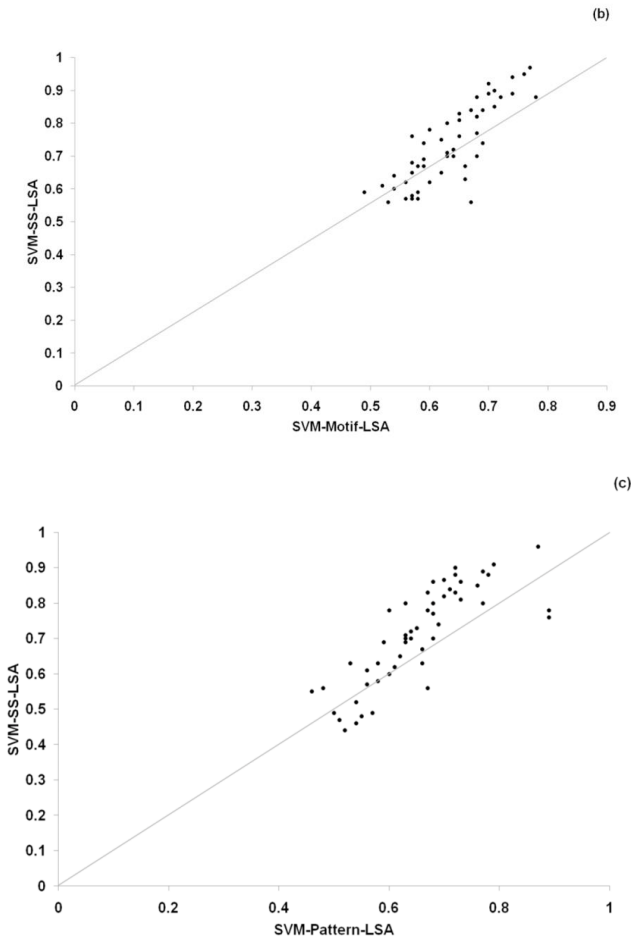


Figure 4. Family by family comparison based on the SVM-SS-LSA method and those with LSA. Each point on the graph corresponds to one of the 54 SCOP families. In each figure, the axes are ROC scores achieved by the two primary methods compared. Figure (a), (b), and (c) are based on SVM-Ngram-LSA, SVM-Motif-LSA, and SVM-Pattern-LSA

Table 1. Average ROC and MRFP scores for the 54 families.

Methods	**Mean ROC**	**Mean MRFP**
SVM-Ngram	0.773000	0.204000
SVM-Pattern	0.791415	0.144053
SVM-Motif	0.813560	0.134893
SVM-Ngram-LSA	0.835387	0.124572
SVM-Motif-LSA	0.859193	0.101688
SVM-Pattern-LSA	0.859484	0.099527
SVM-Fisher	0.878926	0.096300
SVM-SS	0.887630	0.070287
SVM-SS-LSA	0.890626	0.069435

Furthermore, this algorithm also measures the sensitive and non-sensitive regions. Finally, all the high similarity sequences scores are used by LSA to produce the protein feature vectors. In this study, experiment to show the performance of protein substrings against protein sequences was implemented. All the protein sequences were split into different number of *k* values. Different *k* sizes namely 25, 35, 45, 55, 65 have been chosen in order to investigate which value can produce the best results. The result shows that k =45 is better than the other chosen values of k, where it produced the highest value of mean ROC and mean RFP. The results detail can be referred in Table. 2.

Table 2. The number of split protein sequence with ROC and MRFP scores.

	Length of protein sequence				
Measures	***l*=25**	***l*=35**	***l*=45**	***l*=55**	***l*=65**
Mean ROC	0.00000	0.88764	0.89063	0.88961	0.88990
Mean MRFP	0.00000	0.17532	0.06944	0.07042	0.09211

In remote protein homology detection, computational efficiency is the one important aspect. In this regard, SVM-SS-LSA which comprised from the combination of PSS and PPSA, LSA, and SVM is comparable with SVM with LSA and SVM without LSA. The results show that SVM-SS-LSA is more efficient than SVM with LSA and slightly worse than SVM without LSA. Any SVM-based method includes a vectorization and optimization step. The time complexity of the vectorization step SVM-based method is $O(nml)$, where n is the number of training dataset, m is the total number of words, and l is the length of the longest training protein sequence. The protein feature vector extraction of SVM-SS-LSA involves computing n^2 pairwise scores. Using Smith-Waterman, the vectorization step for SVM-SS-LSA is $O(m^2)$ and protein pattern extraction using TEIRESIAS takes $O(nl1lognl)$, where m is the length of the longest protein substring, yielding a total running time of $O(nl1lognl + n^2m^2)$. SVM-SS-LSA has shorter words of protein sequences; thus this will lead to the lowest running time compared to other methods that utilize SVM with LSA. SVM without LSA method uses n-gram, pattern, and motif. By not incorporating LSA, lower running time will be produced as compared to SVM-SS-LSA and SVM with LSA. SVM-Ngram which extracts n-gram using PSI-BLAST takes $O(n^2mt)$, where t is the minimum of n and m, SVM-Pattern which extracts protein pattern using TEIRESIAS algorithm takes $O(nl1lognl + n^2l^2m)$, and SVM-Motif which extracts motif by MEME takes $O(n^2l^2W)$ where W is the width of motif. The SVM with LSA is similar with SVM without LSA and it has an additional SVD process which roughly added t.

8. Conclusion

In this paper, a new method called SVM-SS-LSA is introduced and it has been successfully used in remote protein homology detection. SVM-SS-LSA is presented to manage protein sequences by classifying the protein sequences according to their family. SVM-SS-LSA is based on the detection of high similarity information from protein substring extracted by PSS and PPSA from sequence comparison model, low dimensional representation of protein feature vectors with less redundant and noisy data filtered by chi-square and SVD from generative model, and discrimination of homologue and non-

homologue members by SVM which is a discriminative classifier model. The performance of SVM-SS-LSA is shown by using several measures such as ROC, MRFP, and family by family comparison. The experimental results show that the use of high similarity information of protein substring and low dimension protein feature vectors will yield protein feature vectors of good quality that has been proved to be more successful than the other methods which are based on SVM with LSA and SVM-without LSA. The experiment has been implemented on a benchmark experiment of SCOP superfamily recognition which was designed to simulate the problem of remote protein homology detection.

The remarkable application of SVM-SS-LSA comes from a combination of good methods such as PSS, LSA, and SVM. The PPSA which applies Smith-Waterman algorithm is the best method currently to do protein substring alignment precisely [22]. Whereas, the LSA and SVM are well known methods [10] and have shown a good performance in detecting remote protein homology [8, 22]. In this study, the Smith-Waterman algorithm which has been applied in PPSA has been developed to quantify the similarity information of protein sequences in order to extract high protein similarity information. Their parameters have been optimized over the years to provide relevant measures of similarity for homologous protein substring, and they now represent core tools in computational biology. Besides that, many of the more recent SVM-based methods focus on finding useful representations of protein sequences. Such representations suffer from the peaking phenomenon in many machine-learning methods because the large dimensional protein feature vectors may be introduced. The methods in SVM-SS-LSA are capable to handle large dimensional feature vectors using chi-square and SVD from LSA. Both methods are capable to produce low dimensional protein feature vectors with less redundant and noisy data. Chi-square uses independent technique to reduce redundant data whereas SVD uses thresholding technique to reduce the dimension of matrices from vector space S which the D-dimensional space spanned by the Us, leading to noise removal and good quality representation of protein feature vectors.

The successful application of PSS and PPSA from sequence comparison model, LSA from generative model, and SVM from discriminative classifier model to remote protein homology detection is of great significance. Future improvements are necessary to make SVM-SS-LSA more efficient in detecting remote protein homology. The following directions to be considered in future research are optimizing the protein substring width in order to get all possible substrings of amino acid and having a secondary and tertiary prediction structure with functional properties of proteins that can be investigated to perform as a dataset to detect remote protein homology.

Acknowledgments

This work is supported by the Malaysian Ministry of Science, Technology, and Innovation (MOSTI) under grant no. 01-01-06-SF0228.

References

[1] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman, “Basic local alignment search tool”, Journal of Computational Biology, vol. 215, no. 3, 1990, pp. 403-410.
[2] C.J.C. Burges, “A tutorial on support vector machines for pattern recognition”, Data Mining and Knowledge Discovery, vol. 2, no. 1, 1998, pp. 121-167.
[3] Y.D. Cai, X.J. Liu, X.B. Xu, and G.P. Zhou, “Support vector machines for predicting protein structural class”, BMC Bioinformatics, vol. 2, no. 3, 2001, pp. 1471-2105.

[4] K.C. Chou, "Review: structural bioinformatics and its impact to biomedical science", Current Medicinal Chemistry, vol. 11, no. 16, 2004, pp. 2105-2134.
[5] K.C. Chou, and D.W. Elrod, "Prediction of membrane protein types and subcellular locations", Proteins: Structure Function Genetics, vol. 34, no. 1, 1999, pp. 137-153.
[6] K.C. Chou, and H.B. Shen, "Predicting protein subcellular location by fusing multiple classifiers", Journal of Biochemistry and Cell, vol. 99, no. 2, 2006, pp. 517-527.
[7] Q.W. Dong, L. Lin, X.L. Wang, and M.H. Li, "A pattern-based SVM for protein remote homology detection", in: International Conference on Machine Learning and Cybernetics of the Guangzhou of China, 2005, pp.3363-3368.
[8] Q. Dong, X.L. Wang, and L. Lin, "Application of latent semantic analysis to protein remote homology detection", Bioinformatics, vol. 22, no. 3, 2006, pp. 285-290.
[9] A. Fukushima, M. Wada, S. Kanaya, and M. Arita, "SVD based anatomy of gene expressions for correlation analysis in arabidopsis thaliania", DNA Research, vol. 15, no. 1, 2008, pp. 367-374.
[10] B. Gabrys, R.J. Howlet, and L.C. Jain, "Knowledge-Based intelligent information and engineering systems", in: Proceeding of the Tenth Conference KES of the Bournemouth of United Kingdom, 2006, pp.393-400.
[11] O. Gotoh, "An improved algorithm for matching biological sequences", Molecul Biology, vol. 162, no. 1, 1982, pp. 705-708.
[12] T. Jaakkola, M. Diekhans, and D. Haussler, "A discriminative framework for detecting remote protein homologies", Journal of Bioinformatics and Computational Biology, vol. 7, no. 1-2, 2000, pp. 95-114.
[13] A. Kelil, S. Wang, R. Brzezinski, and A. Fleury, "CLUSS: clustering of protein sequences based on a new similarity measure", BMC Bioinformatics, vol. 8, no. 1, 2007, pp. 1-19.
[14] R. Kuang, E. Ie, K. Wang, K. Wang, M. Siddiqi, Y. Freund, and C. Leslie, "Profile-Based string kernels for remote homology detection and motif extraction", Journal of Bioinformatics and Computational Biology, vol. 3, no. 3, 2004, pp. 152-160.
[15] T.K. Landauer, P.W. Foltz, and D. Laham, "Introduction to latent semantic analysis", Discourse Process, vol. 25, no. 1, 1998, pp. 259-284.
[16] L. Liao, and S.N. Noble, "Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships", Journal of Computational Biology, vol. 10, no. 1, 2003, pp. 857-868.
[17] S. Mohseni-Zadeh, P. Brezellec, and J.L. Risler, "Cluster-C, an algorithm for the large-scale clustering of protein sequences based on the extraction of maximal cliques", Computational Biology and Chemistry, vol. 28, no. 1, 2004, pp. 211-218.
[18] W.R. Pearson, "Rapid and sensitive sequence comparison with FASTP and FASTA", Methods Enzymo, vol. 183, no. 1, 1990, pp. 63-98.
[19] I. Rigoutsos, and A. Floratos, "Combinatorial pattern discovery in biological sequences: the TEIRESIAS algorithm", Bioinformatics, vol. 14, no. 1, 1998, pp. 55-67.
[20] Y. Tang, B. Jing, and Y.Q. Zhang, "Granular support vector machines with association rules mining for protein homology prediction", Artificial Intelligence in Medicine, vol. 25, no. 1, 2005, pp. 121-134.
[21] Y. Yang, and J.O. Pedersen, "A comparative study on feature selection in text categorization", in: Proceedings of the International Conference on Machine Learning of the Salvador of Brazil, 1997, pp.412-420.
[22] M.N. Zaki, and S. Deris, "Detecting remote protein evolutionary relationships via string scoring method", International Journal of Biomedical Sciences, vol. 2, no. 1, 2007, pp. 59-66.

Authors

Surayati Ismail is a Researcher at the Laboratory of Computational Intelligence and Biotechnology at the Universiti Teknologi Malaysia. She received the B.Sc. and M.Sc. degrees in Computer Science both from the Universiti Teknologi Malaysia, in 2006 and 2010, respectively. Her research interests focus on remote protein homology detection, machine learning algorithm, and computational biology.

Razib M. Othman is a Director of Laboratory of Computational Intelligence and Biotechnology at the Universiti Teknologi Malaysia. He received the B.Sc, M.Sc. and Ph.D. degrees in Computer Science from the Universiti Teknologi Malaysia, in 1993, 2003, and 2008, respectively. His research interests are in the areas of computational intelligence, computational biology, and software engineering.

Shahreen Kasim is a Tutor at the Faculty of Computer Science and Information Technology, the Universiti Tun Hussein Onn Malaysia. She received the B.Sc., M.Sc., and Ph.D degrees in Computer Science from the Universiti Teknologi Malaysia, in 2003, 2005, and 2011 respectively. Her research interests focus on gene function prediction, clustering algorithm, and computational biology.

Rohayanti Hassan is a Lecturer at the Faculty of Computer Science and Information Systems, the Universiti Teknologi Malaysia. She received the B.Sc., M.Sc., and Ph.D degrees in Computer Science from the Universiti Teknologi Malaysia, in 2003, 2006, and 2011 respectively. Her research interests focus on protein structure prediction, clustering algorithm, and computational biology.

Hishammudin Asmuni is a Senior Lecturer at the Faculty of Computer Science and Information Systems, the Universiti Teknologi Malaysia. He received the Ph.D. degree in Computer Science from The University of Nottingham, UK in 2008, the M.Sc. degree in Computer Science from the Universiti Teknologi Malaysia in 1999, and the B.Sc. degree in Computer Science from the Universiti Malaya in 1996. His research interests focus on timetabling/scheduling, fuzzy systems, and bioinformatics.

Jumail Taliba is a Lecturer at the Faculty of Computer Science and Information Systems, the Universiti Teknologi Malaysia. He received the B.Sc. and M.Sc. degrees in Computer Science both from the Universiti Teknologi Malaysia, in 1997, and 2001, respectively. His research interests focus on protein-protein interaction prediction, image processing algorithm, and computational biology.