International Journal of Computational Intelligence Systems, Vol.2, No. 4 (December, 2009), 343-352

Granular Decision Tree and Evolutionary Neural SVM for Protein Secondary Structure Prediction

Anjum Reyaz-Ahmed
Department of Computer Science,
Georgia State University, Atlanta, GA 30302-3994, USA
E-mail: anjumrahmed@gmail.com

Yan-Qing Zhang
Department of Computer Science,
Georgia State University, Atlanta, GA 30302-3994, USA
E-mail: yzhang@cs.gsu.edu

Robert W. Harrison
Department of Computer Science,
Georgia State University, Atlanta, GA 30302-3994, USA
E-mail: cscrwh@asterix.cs.gsu.edu

Received: 05/01/09
Accepted: 08/10/09

Abstract

A new sliding window scheme is introduced with multiple windows to form the protein data for SVM. Two new tertiary classifiers are introduced; one of them makes use of support vector machines as neurons in neural network architecture and the other tertiary classifier is a granular decision tree based on granular computing, decision tree and SVM. Binary classifier using multiple windows is compared with single window scheme. The accuracy levels of the new classifiers are better than most available techniques.

Keywords: evolutionary computation; granular decision tree; neural networks; protein structure prediction; support vector machines; tertiary classifier.

1. Introduction

In Bioinformatics, selecting suitable classification algorithms is important in terms of classification accuracy and efficiency. Therefore, it is important to search available methods to get optimum classification models. In this paper, two novel tertiary classifiers are proposed for protein secondary structure prediction. The first tertiary classifier based on supervised learning methods makes use of support vector machines (SVM), neural networks and genetic algorithms, and another is the granular decision tree based on the inclusion method that makes uses of information already in binary classifiers. In general, both supervised learning methods and un-supervised learning methods are used to verify how to select an effective classifier for accurate classification.

Protein tertiary structure determines the functional characteristics of the proteins. The secondary structure is closely related to the tertiary structure. The success of genome sequencing program resulted in massive amounts of protein sequence data (that are produced by DNA sequencing) [HUMAN GENOME PROJECT]. There are many more protein amino acid sequences than there are experimentally determined structures. Therefore, it is becoming increasingly important to predict protein structure from its amino acid sequence, using insights obtained from already known structures.

The main objective of this study is to compare the different classifier techniques. The method adopted here uses SVM, neural network and genetic algorithm for optimization. This classifier is also an adaptive one and can be used for different applications. The other classifier is a granular decision tree called a complete SVM decision tree and it makes use of binary classifier's results.

The SVM method is a comparatively new learning system that is mostly used in pattern recognition problems. This machines uses hypothesis space of linear functions in a high-dimensional feature space, and it is trained with a learning algorithm based on optimization theory. To compare the results of this study with previous results RS126 data set is used [1]. The RS126 set is a relatively small data set (126 proteins, while there are more than 57000 proteins in the pdb), but despite this limitation it is important to use a standard data set to evaluate the differences in the machine learning algorithms. A "production grade" machine learning tool should be trained on a larger and more complete data set to enhance its accuracy. Among neural networks Chandonia and Karplus [3] introduced a novel method for processing and decoding the protein sequence with NNs by using large training data set such as 681 non homologous proteins. And with the use of jury method, this scheme records 74.8% percentage accuracy. Many recent studies adopting the SVM learning machine for secondary structure prediction use frequency profiles with evolutionary information. Examples include: profiles as an encoding scheme for SVM [4] two layers of SVM, with a weighted cost function [5], PSI-BLAST PSSM profiles [6] as an input vector and a sliding window scheme with SVM Representative architecture [7]. This paper is a complete summary of the research done on protein secondary structure prediction. The individual sections containing details about new tertiary architecture which combines both the SVM and the neural networks and uses genetic algorithms for optimization [8] and SVM-based decision fusion method using multiple granular windows [9] are referenced.

This research introduces new encoding scheme of multiple windows instead of the traditional single window scheme used in other researches. In this study, the single window technique is challenged with new multiple windows technique. Here multiple sliding windows are used instead of single window. The center element of the middle window is the target residue. All other residues inside the windows are used as feature values to train and test the SVM. Sliding window technique is used to move to the next residue. The tertiary classifier makes use of both the traditional single window as well as the new multiple windows encoding scheme. The results are compared with other prominent tertiary classifiers. The binary classifier makes use of BLOSUM62 matrix and orthogonal matrix for effective encoding of the protein data.

Granular computing is a study in which the details of data are seen from different scales. In granular computing the data are presented different levels of detail [11] [12]. The difference between multiple windows and single window method can also be considered as different approach for scaling of protein sequence data to predict its structure. The different course employed in considering granules in the encoding scheme reflects on the accuracy of the method. Two novel tertiary architectures are introduced. In both the architectures the results have better accuracy when compared to the method proposed by Hu [7]. In Hu's method the tertiary classifier uses only three of the six binary classifiers. In the proposed methods of this paper all the six binary classifiers are applied to form the tertiary classifier. This is to make use of knowledge from all the binary classifiers. Both single window as well as multiple windows schemes was tested for getting the best results.

The rest of the paper is organized as follows. Section 2 introduces a new granular window encoding scheme that is a general framework for commonly used single window encoding scheme. Section 3 presents a basic knowledge of SVM and introduces neural networks and genetic algorithms. Section 4 proposes a novel Evolutionary Neural Support Vector Machines (ENSVM) that is a new classifier based on SVM, neural networks and genetic algorithms. Section 5 discusses a new granular decision tree called SVM_Complete for effective binary classification. Section 6 shows different simulation results. Finally, Section 7 concludes the paper and proposes future works.

2. Multiple Windows Encoding Scheme

The RS 126 data set is proposed by Rost & Sander [1] and according to their definition, it is non-homologous set. They used percentage identity to measure the homology and defines non-homologous as no two

proteins in the set share more than 25% sequence identity over a length of more than 80 residues.

For each data set, the seven fold cross validation is done [1,4,7]. In the seven-fold cross validation test, one subset is chosen for testing and remaining 6 subsets are used for training and this process is repeated until all the subsets are chosen for the testing.

The secondary structure is converted from the experimentally determined tertiary structure by DSSP [13], STRIDE [14] or DEFINE [15] . In this study, the DSSP scheme is used since it is the most generally used secondary structure prediction method.

Table 1 8-to-3 state reduction method in secondary structure assignment

DSSP Class	8-state symbol	3-state symbol	Class name
3_{10} – helix α-helix π-helix	G H I	H	Helix
β-strand	E	E	Sheet
isolated β-bridge Bend Turn Rest (connection region)	B S T -	C	Loop

The DSSP classifies residues into eight different secondary structure classes: H (α-helix), G (3_{10} – helix), I (π-helix), E (β-strand), B (isolated β-bridge), T (turn), S (bend), and – (rest). In this study, these eight classes are reduced into three regular classes based on the following Table 1. There are other ways of class reduction as well but the one applied in this study is considered to be more effective.

In the case of single window encoding, to train the SVM with protein sequence and structural information, a sliding window scheme is used [7]. In this sliding scheme, a window becomes one training pattern for predicting the structure of the residue at the center of the window. And in this training pattern, the information about the local interactions among neighboring residues is embedded.

In the case of multiple windows scheme, instead of using a single sliding window multiple sliding windows are used. The center element of the middle window becomes the target and all other windows are used as feature values to train and test the SVM. Only the elements/residues/granules inside the window forms the training/testing data, some residues in the middle are skipped. Sliding window technique was applied to move to the next residue. In this study windows of equal sizes are considered. Windows of different sizes will be studied as future technique. In the case of different size windows, the window in the middle will have more residues than windows at each side. In all the multiple windows cases consider have three windows with different lengths.

Initially the BLOSUM 62 matrix [10] coupled with orthogonal encoding scheme was used. The BLOSUM matrices originate from the paper by Henikoff and Henikoff (1992). Their idea was finding a good measure of difference between two proteins specifically for more distantly related proteins. The value in the BLOSUM62 matrix are 'log-odds' scores for the likelihood that a given amino acid pair will interchange. Amino acids with similar physical properties are more likely to replace one another than dissimilar amino acids. To obtain the optimal input profile, which offers the most informative feature to predict the secondary structure with high accuracy, the orthogonal input profile and BLOSUM matrix profile are combined together. When more than one encoding scheme is used the weight is applied based on a position inside a window. In other words, even though each amino acid has 20 different 'log odds' scores, those values are always same regardless of the position inside the sliding window. Therefore by assigning different weights based on their position inside the window, the machine could be trained with more specific information.

To achieve high testing accuracy, a suitable kernel function, its parameter and the regularization parameter C should be properly selected. Hua and Sun [4] has proved that the Gaussian kernel can provide superior performance in the generalization ability and convergence speed. Therefore, in this study, according to the previous result, Gaussian radial basis function kernel was adopted. Once the kernel function is selected, the parameter of the kernel function, γ, and the regularization parameter, C which controls the trade-off between complexity and misclassified training example, should be specified by the user.

Six SVM binary classifier including three one-versus-rest classifier ('one': positive class, 'rest': negative class) names H/~H, E/~E and C/~C and three one-versus-one classifier named H/E, E/C, C/H were constructed. For example, the classifier H/E is constructed on the training samples having helices and sheets and it

classifies the testing sample as helix or sheet. The programs for constructing the SVM binary classifier were written in the C language.

The single window technique was compared with multiple windows encoding scheme. The same parameter values were used in both schemes. The comparison of the two techniques reveals single window scheme not to be good in all cases. For window of size 15 the simulation results show the multiple windows to be better than single window for all the six binary classifiers. The results are shown in the Table 2. In this case the single window is of length 15 and in the multiple windows case, 3 windows each of size 5 with gaps between the windows are used. In both the cases RBF kernel is used with the same parameter values (gamma γ and cost co-efficient C).

Table 2 Comparing Single Window and Multiple Windows

Binary Classifier	Multiple Windows	Single Window
H/H	73.59	73.52%
E/E	78.39%	78.39%
C/C	69.69%	69.62%
H/E	72.94%	72.33%
E/C	75.9%	75.59%
C/H	71.93%	71.74%
Average	73.74%	73.53%

In another simulation single window of size 21 is compared with 3 windows, each of size 5 and a gap of 3 residues (gap means these three residues was not considered to form the data for SVM) between the windows. The results of this simulation are shown in Table 3. This indicates single window not be good in all cases and multiple windows has less information to process (as it has only 15 residues to consider where as single window have 21 residues in each set).

The optimal window length and other optimal values of the parameters are selected to be the same as those used in previous studies. As the previous studies have already run simulations and have obtained the optimal values for all the parameters, further research is avoided.

Considering all the points multiple windows shows scope for performance. This study was conducted to determine if single window scheme is solely the best method to do protein secondary structure prediction, empirically there is scope for other methods too. The optimal window length and other optimal values of the parameters are selected to be the same as those used in previous studies. As the previous studies have already run simulations and have obtained the optimal values for all the parameters, further research is avoided.

Table 3 Simulation II: Single Window vs. Multiple Windows.

Binary Classifier	Accuracy of Multiple Windows	Accuracy of Single Window
H/H	72.37%	74.67%
E/E	78.41%	78.34%
C/C	70.00%	69.63%
H/E	72.24%	73.70%
E/C	75.45%	74.30%
C/H	71.43%	72.90%
Average	73.32%	73.92%

3. Machine Learning Techniques

Main focus of a machine learning algorithm is to make intelligent decisions based on available knowledge from some database. For this research we have considered the following algorithms.

3.1. *Support vector machines*

Support Vector Machines (SVM) are learning systems that use a hypothesis space of linear function in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias derived from statistical learning theory. This learning strategy introduced by Vapnik [16] and co-workers is a principled and very powerful method that in the few years since its introduction has already outperformed most other systems in a wide variety of applications [17].

Since SVM approach has a number of superior such as effective avoidance of over fitting, the ability to handle large feature spaces, information condensing of the given data set, it has been successfully applied to a wide range of pattern recognition problems, including isolated handwritten digit recognition, objective recognition, speaker identification, and text categorization, etc [18].

Binary classifier is frequently implemented by using a real-valued function $f: X \subseteq \Re^n \to \Re$ in the following way: the input $x = (x_1,....,x_n)'$ is assigned to the positive class, if $f(x) \geq 0$, and otherwise to the negative class. If we consider the case where $f(x)$ is a linear function of $x \in X$, so that it can be written as

$$f(x) = w \bullet x + b \quad (1)$$

$$= \sum_{i=1}^{n} w_i x_i + b \qquad (2)$$

Where, $(w, b) \in \Re^n \times \Re$ are the parameters that control the function and the decision rule given by $\text{sgn}(f(x))$. And these parameters must be learned from the data.

If we interpret this hypothesis geometrically, input space X is split into two parts by the hyperplane defined by the equation $w \bullet x + b = 0$. For example, in Figure 3.1, the hyper plane is the dark line, with the positive region above and the negative region below. The vector w defines a direction perpendicular to the hyperplane, while varying the value of b moves the hyperplane parallel to itself. And these quantities are referred as the weight vector and bias which are the terms borrowed from the neural networks literature.

The above algorithm for separable data, when applied to non-separable data, will find no feasible solution: this will be evidenced by the objective function) i.e. the dual Langrangian) growing arbitrarily large. To extend these ideas to handle non-separable data, the constraints (1) and (2) are relaxed, but only when necessary, that is, a further cost (i.e. an increase in the primal objective function) is introduced. This can be done by introducing positive slack variables $\xi_i, i = 1,...1$ in the constraints, which then become:

$$x_i \bullet w + b \geq +1 - \xi_i \quad for\ y_i = +1 \qquad (3)$$

$$x_i \bullet w + b \leq -1 + \xi_i \quad for\ y_i = -1 \qquad (4)$$

Thus, for an error to occur the corresponding ξ_i must exceed unity, so is an upper bound on the number of training errors. Hence a natural way to assign an extra cost for error is to change the objective function to be minimized from $\|w\|^2/2$ to $\|w\|^2/2 + C(\sum_I \xi_i)^k$, where C is a parameter to be chosen by the user, a larger C corresponding to assigning a higher penalty to error [17]. The soft margin classifier is an extension of linear SVM. The kernel method is a scheme to find the nonlinear boundaries. The concept of the kernel method is transformation of the vector space to a higher dimensional space. By transforming the vector space from two-dimensional to three-dimensional space, the non-separable vectors can be separated.

The prediction of protein secondary structure is done using SVM*light* software. SVM*light* software is the implementation of Vapnik's Support Vector Machine (Vapnik 1995) for the problem of pattern recognition, regression and ranking function. SVM*light* software consists of two parts, the first part i.e. is the svm_learn part takes care of the learning module and the second part svm_classify part does the classification of the data after training.

3.2. *Neural networks and genetic algorithms*

A neural network, implemented as a parallel distributed processing network, is a computing paradigm that is loosely modeled after cortical structures of the brain. It consists of interconnected processing elements called nodes or neurons that work together to produce an output function. The output of a neural network relies on the cooperation of the individual neurons within the network to operate [20]. Here neural network is used to form the new tertiary architecture. The neurons in this network are SVM machines that classify the input data into two classes of protein structures. The two classes are the binary classes that the SVM machines are actually trained for. The tertiary classifier's architecture is explained in subsequent sections.

A genetic algorithm (or GA) is a search technique used in computing to find true or approximate solutions to optimization and search problems. Genetic algorithms are categorized as global search heuristics [20]. Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination).

Genetic algorithms are implemented as a computer simulation in which a population of abstract representations (called chromosomes or the genotype or the genome) of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem evolves toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible.

The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly mutated) to form a new population [20]. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. If the

algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.

There are many ways to combine the output from the binary classifier for secondary structure prediction. In this research, several tertiary classifiers proposed by previous studies [4,7] were tested and compared with the new tertiary classifier of this study. Here, the new tertiary classifier is designed based on the results of three one-versus-one binary classifier.In the tertiary architecture introduces in this study constructed using neural network, genetic algorithm is used to train the neural network. This architecture is termed as 'Evolutionary Neural Support Vector Machines' (ENSVM). The weights obtained by training the neural network is then used in testing phase.

4. Evolutionary Neural support vector machines: ENSVM

The new tertiary classifier proposed in this paper, makes use of both one-versus-one as well as one-versus-rest binary classifiers. The novel architecture makes use of all the six binary classifiers in neural net architecture. The architecture is shown in the Fig. 1.

The first phase in the construction of the architecture, is the formation SVM (binary classifiers) which are built to perform to their best (i.e. by using optimal window size in the case of single window or optimal slide size and window sizes in the case of multiple windows scheme; and also considering the optimal parameters for the construction of the RBF kernel, as it has been proved by the previous works that RBF kernel has superior performance in the generalization ability and convergence speed). Based on the former studies the binary classifiers have an average nearing 80%.

The next phase is to make use of the new neural network architecture. In this architecture the SVM are used as neurons as shown in Figure 1. There are two hidden layers; the output of the first one is the same as the output of the individual SVM.

Outputs of first hidden layer are as follows.

O_1 = SVM (H/~H)

O_2 = SVM (E/~E)

O_3 = SVM(C/~C)

The output of the second hidden layer considers the output of the first layer as well as the SVM machine stored in that layer. For example the output of the neuron 4 has an SVM binary classifier that positively classifies H and negatively classifies E, the result of this SVM is combined with that of the first layer outputs. This method uses the outputs of the three one-versus-rest classifier in a single neuron.

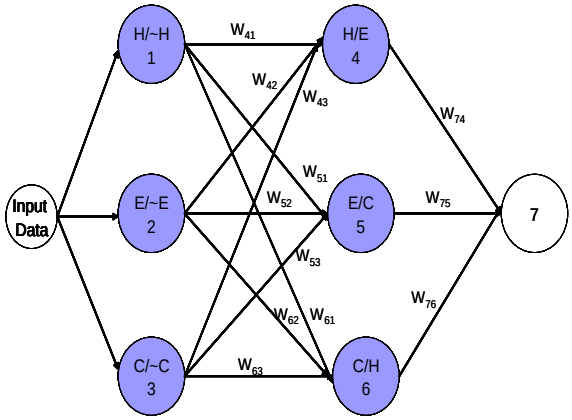

Figure 1 Evolutionary Neural Support Vector Machines

In the formulations the output of the second hidden layer is formed by adding the output of the SVM (sitting inside the neuron) with the product of the weight and output of the corresponding neuron (i.e. the neuron which positively classifies the same class as the current neuron) and by subtracting the products of the other two neurons in the first layer.

The output of the second layer are calculated as

$$O_4 = \text{SVM (H/E)} + W_{41} O_1 - W_{42} O_2 - W_{43} O_3$$

In the above formula we add the values of the SVM that positively classify the same class (H) and subtract those that positively classify other classes. Here W_{41} means weight between neuron 1 and neuron 4. Similarly other weight corresponds to output, input naming pattern. See Fig. 1.

Similarly outputs of other two neurons in the second hidden layer are calculated as

$$O_5 = \text{SVM (E/C)} + W_{52} O_2 - W_{51} O_1 - W_{53} O_3$$

$$O_6 = \text{SVM (H/E)} + W_{63} O_3 - W_{62} O_2 - W_{61} O_1$$

The final output layer does not have any SVM embedded in it. It calculates its results based on maximum of the three outputs of second hidden layer. There is only one neuron in this layer. The final output is one among the three classes (H, E or C), which ever neuron produces the maximum output after multiplying it with appropriate weight with the second hidden layer output is considered as final output.

So the output of the third layer is as follows.

If [Max ($W_{74} O_4$, $W_{75} O_5$, $W_{76} O_6$) = $W_{74} O_4$)]

Then

O_7 = H

Else If [Max ($W_{74} O_4$, $W_{75} O_5$, $W_{76} O_6$ = $W_{75} O_5$)]

Then

O_7 = E

Else

O_7 = C

For optimizing the weights Genetic Algorithm is used. The weight range is selected to be between 0 and 1 so that the architecture performs to its full potential.

5. Granular Decision Tree: SVM_Complete

This is a simple inclusion method, in which all the six binary classifiers are used to form the tertiary classifier. In SVM_Represnt. scheme [7], irrespective of the distance value's sign (positive / negative), the classifier with the absolute maximum distance is chosen as the representative classifier for the final decision of the class. In this paper, we consider that fact that among the three one-versus-one classifier, two classifier try to identify the same class, for example H/E and C/H tries to classify H (only difference is in H/E H is the positive class and in C/H H is the negative class). So we add up the values of one-versus-one classifier which classifies the same class. Then we also add the value of one-versus-rest classifier, to sum up the total strength of the specific class. For example, for calculating the strength of H, we have to:

Step 1: Check if SVM (H/E) is positive, if true

H = absolute value of SVM (H/E)

Step2: Check if SVM (C/H) is negative, if true

H = H + absolute value of SVM (C/H)

Step 3: Add one-versus-rest prediction value

H = H + value of SVM (H/~H). *

* Note here we add the actual value not absolute, since we want to determine H's total strength.

Similarly strength of E and C are calculated and final result is produced depending upon which class has the highest value. Here SVM (H/E) means the exact output the support vector machine gives after classifying the given data.

In SVM VOTE [4], all six binary classifiers are combined by using a simple voting scheme in which the testing sample is predicted to be state i (i is among H, E and C) if the largest number of the six binary classifiers classify it as state i. In case the testing samples have two classifications in each state, it is considered to be a coil. Though all six binary classifiers are considered for tertiary classification, only one constitutes the results. In SVM_Complete all six classifiers are used to calculate individual strengths of each class and finally the one with highest strength is considered as the predicted secondary structure.

6. Simulations and Performance

For comparing the results of this study with previously published results [7], RS 126 data set is used.

There are several standard evaluation methods of secondary structure prediction. Among them, Q_3, Matthew's Correlation Coefficient and Segment Overlap Measure (SOV) are widely used assessing methods. We have simulated results comparing the Q_3 percentage value of different tertiary classifier.

Q_3 is one of the most commonly used performance measures in the protein secondary structure prediction and it refers to the three-state overall percentage of correctly predicted residues. This measure is defined as,

$$Q_3 = \frac{\sum_{i \in \{H,E,C\}} \#of\ residues\ correctly\ predicted_i}{\sum_{i \in \{H,E,C\}} \#of\ residues\ in\ class\ i} \times 100 \quad \textbf{(5)}$$

Based on the above equation, the per-residue accuracy for each type of secondary structure (Q_H, Q_E, Q_C) can be
obtained as:

$$Q_I = \frac{\#of\ residues\ correctly\ predicted\ in\ state\ I}{\#of\ residues\ in\ state\ I} \times 100 \quad \textbf{(6)}$$

$I \in \{H, E, C\}$

The new tertiary classifier (neural network architecture) is compared with other tertiary architectures of former studies. The 7 fold test cases have been performed for a valid comparison of the new tertiary classifier with that of the SVM_Represnt., [7] contributed classifier. The accuracy percentage of the new methods is compared with that of the other methods. The accuracy level of the tertiary classifier is important from research point of view, as the main objective of protein secondary structure prediction is to accurately determine the secondary structure of the protein sequence. The Table 4 gives the accuracy level of all the methods [7] and also the accuracy levels two

new classifiers ENSVM and SVM_Complete (tertiary classifiers of this research). As shown in the Table 4 ENSVM is better than other available methods.

First the accuracy levels of single window encoding scheme is compared with different former methods. As seen in Table 4 the average accuracy of the ENSVM (a new classifier of this study) is better than other available methods and SVM_ Complete shows the best performance. Table 4 shows the accuracy level for window of size 15.

The results in the Table 4 are obtained after 7-fold cross validation for window of size 15. The accuracies are compared with other classifiers that use single window encoding scheme. In Table 4 accuracy levels of SVM_Represnt. [7]) And SVM_VOTE [4] are obtained by simulation after 7-fold cross validation. All other former classifiers accuracies are adopted from [7].

Table 4 Accuracy of tertiary Classifiers on RS 126

Tertiary Classifier	Q_3 (%)	Q_H (%)	Q_E (%)	Q_C (%)
TREE_HEC	63.2	51.0	45.2	79.9
TREE_ECH	62.3	62.4	26.2	79.0
TREE_CHE	61.2	64.8	47.3	65.2
SVM_VOTE	62.0	73.5	34.7	65
SVM_MAX_D	63.2	61.0	40.1	75.5
DAG	63.2	59.2	41.6	76.0
SVM_REPRESNT.	63.2	70.6	35.4	70.5
ENSVM	66.1	68.3	49.8	72.1
SVM_Complete	66.7	64.0	40.8	80.3

Note: The table is adopted from (Hu and Yi, 2004) [7].

Table 5 Q_3 % for Different Window Sizes

Window Size	ENSVM	SVM_Complete	SVM_Represnt.
15	66.10%	66.70%	63.15%
13	63.20%	64.10%	62.43%
11	57.10%	56.82%	55.93%

Closely analyzing the accuracy levels, it is recorded that Neural Network using SVM (ENSVM) has performed equally well in all 3 cases (H, E and C), when compared to other methods that have very high Q_C accuracy and have very low Q_E accuracy. The 'Evolutionary Neural SVM' still has scope of improvement as it is a neural network technique which is optimized using Genetic Algorithms, its potential can be further increased.

The simulations were performed for many window sizes. The accuracy levels of ENSVM, SVM_Complete and SVM_Represnt. is shown inTable 5 for window sizes 15, 13 and 11.

Table 6 Accuracy of Tertiary Classifier Using Multiple Windows Scheme

Tertiary Classifier	Q_3 (%)	Q_H (%)	Q_E (%)	Q_C (%)
SVM_VOTE	62.6	78.9	39.7	62.4
SVM_REPRESNT.	64.8	72.1	41.8	72.0
ENSVM	68.0	73.5	52.3	71.7
SVM_Complete	68.4	69.1	45.0	78.8

The same tertiary classifiers were demonstrated with binary classifiers using multiple windows scheme. This resulted in increase in total accuracy level, which is expected as the binary classifiers formed using multiple window scheme are better when compared to single window encoding scheme. The binary classifier used is constructed using three consecutive windows each of size 5 with gaps between the first and second window as well as between second window and third window. The results of these simulations are shown in Table 6.

7. Conclusions and Future Work

After many demonstrations, it is now established that single window scheme is not the only best method to encode while considering BLOSUM62 and orthogonal matrix. Multiple windows scheme performed better in some cases where the data given to the learning machine (SVM) was less informative than that given in single window scheme. When both were encoded with equal amount of information, multiple window schemes' performance is better than single window scheme in every case.

The tertiary classifiers, ENSVM and SVM_Complete have shown to perform better than other contemporary techniques. ENSVM tertiary classifier has less accuracy when directly compared with the results of former tertiary classifiers belonging to previous studies. Though the method is not better when compared directly to the claimed accuracy levels of the former methods, the encoding scheme of binary classifiers used in those methods is different and better than the one used in this study.

In future more advanced and enhanced techniques like PSSM (position specific scoring matrix) will be used to improve the accuracy level of the new method. The accuracy level of genetic neural network can be further increased, as it makes use of genetic algorithm, its full potential has not reached yet. Also other protein datasets will be considered as prospective research. Also frequency profiling technique used by Hua and Sun [4] will also be tested to see if multiple windows scheme is better than single window scheme. After forming the best binary classifiers, the new tertiary classifiers will be tested to prove that their performance is best among all the current research methods.

The future work primarily deals with using different encoding schemes, which will increase the results of both binary as well tertiary classifier's accuracy levels. More concrete case can be developed if other datasets like CB513 etc. is used to prove the supremacy of these new methods over other contemporary techniques.

With the implementation of these new classification methods, we hope to have shed a new light in the field of using the granular decision tree for accurate protein secondary structure prediction.

Acknowledgements

The first author is supported by MBD (Molecular Basis of Disease) fellowship of Georgia State University.

8. References

1. Rost, B. and Sander, C. "Improved prediction of protein secondary structure by use of sequence profile and neural networks", *Proc Natl Acad Sci U S A* 90, pp. 7558-62, 1993.
2. Kabsch, W. and Sander, C. "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical feature". *Biopolymers* 22, pp. 2577-2637, 1983.
3. Chandonia, J.M. and Karplus, M. "New method for accuracy prediction of protein secondary structure", *Proteins* 35, pp. 293-306, 1999.
4. Hua, S. and Sun, Z., "A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach", *J. Mol. Biol.* 308, pp. 397-407, 2001.
5. Casbon, J. "Protein secondary structure prediction with support vector machines", 2002.
6. Jones D.T. "Protein secondary structure prediction based on position-specific-scoring matrices", *J. Mol. Biol*., 292, pp. 195-202, 1999.
7. Hu, H. and Yi, P. "Improved secondary structure prediction using support vector machines with a new encoding scheme and an advanced tertiary classifier", *IEEE Transaction on Nanobioscience,* vol. 3, no. 4, 2004.
8. Reyaz-Ahmed, A. and Zhang, Y.-Q., "Protein Secondary Structure Prediction Using Genetic Neural Support Vector Machines", *Proc. of IEEE 7th International Conference on BioInformatics and BioEngineering*, pp. 1355-1359, Boston, Oct.14-17, 2007.
9. Reyaz-Ahmed A. and Zhang, Y.-Q., "A New SVM-Based Decision Fusion Method Using Multiple Granular Windows for Protein Secondary Structure Prediction", *RSKT*, 2008
10. Heniko, S. and Heniko, J.G. "Amino acid substitution matrices from protein blocks". *PNAS* 89, 10915-10919 (1992).
11. Tang Y.C., Jin, B. and Zhang Y.-Q. "Granular Support Vector Machines with Association Rules Mining for Protein Homology Prediction", *Artificial Intelligence in Medicine, Special Issue on Computational Intelligence Techniques in Bioinformatics*, vol. 35, no. 1-2, pp. 121-134, Sept.-Oct. 2005.
12. Jin B., Zhang Y.-Q. and Wang B.H., "Granular Kernel Trees with Parallel Genetic Algorithms for Drug Activity Comparisons", *International Journal of Data Mining and Bioinformatics*, vol. 1, no. 3, pp. 270-285, 2007.
13. Kim, H. and Park, "Protein secondary structure prediction based on an improved support vector machines approach", *Protein Eng,* 16, pp. 553-560, 2003
14. Frishman, D. & Argos, P. "75% accuracy in Protein Secondary Structure Prediction", *Proteins*, 27, pp. 329-335, 1997.
15. Richards, F.M. and Kundrot, C.E., "Identification of structural motifs from protein coordinate data: secondary structure and first-level super secondary structure", *Proteins*, vol. 3, pp. 71-84, 1988.
16. Vapnik, V. and Corter, C., "Support vector networks", *machine learning*. vol. 20, pp. 273-293, 1995.
17. Burges, C.J.C. "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, vol. 2, pp. 121-167, June 1998.
18. Christianini, N. and Shawe-Taylor, J. *An introduction to support vector machines*, Cambridge University Press, 2000.
19. Joachims, T. "Making Large-Scale SVM Learning Practical" *Advances in Kernel Methods - Support Vector Learning*, B. Schölkopf and C. Burges and A. Smola (Ed.), MIT Press, 1999.
20. Jang, Sun and Mizutani, *Neuro-Fuzzy and soft computing*, Princeton Hall, Inc., 1997.
21. Zemla, A., Venclovas, C., Fidelis, K. and Rost, B. "A modified definition of sov, a segmentbased measure for protein secondary prediction assessment", *Proteins: Structure, Function, and Genetics*, 34, pp. 220-223. (1999)

22. Rost, B., Sander, C. and Schneider, R. “Redefining the goals of protein secondary structure prediction”, *J. Mol. Biol.*, 235, pp. 13-26. (1994)
23. Heiler, M. *Optimization Criteria and Learning Algorithms of Large Margin Classifiers*, University of Mannheim. (2002)
24. Nguyen, M.N. & Rajapakse, J.C. “Multi-Class Support Vector Machines for Protein Secondary Structure Prediction”, *Genome Informatics*, 14, pp. 218-227.(2003)
25. Weston, J. and Watkins, C. “Multi-class support vector machines”, in: Verleysen, M. (Ed.) *Proceedings of ESANN99*, Brussels, D. Facto Press.(1999)