ENSEMBLE OF NEURAL NETWORKS TO SOLVE CLASS IMBALANCE PROBLEM OF PROTEIN SECONDARY STRUCTURE PREDICTION

Maryam Alirezaee[1], Abdollah Dehzangi[2,3] and Eghbal Mansoori[1]

[1] School of Electrical & Computer Engineering, Shiraz University, Shiraz, Iran
alirezaiee@cse.shirazu.ac.ir
mansoori@shirazu.ac.ir
[2]Institute for Integrated and Intelligent Systems (IIIS), Griffith University, Brisbane, Australia
[3]National ICT Australia (NICTA), Brisbane, Australia
Abdollah.dehzangi@griffithuni.edu.au

ABSTRACT

Protein secondary structures prediction (PSSP) is considered as a challenging task in bioinformatics. Many approaches have been proposed in last few decades in order to solve this problem. Despite the enhancements achieved, the prediction accuracy still remains limited. Accurate prediction of the secondary structure of proteins is a critical step in deducing tertiary structure of proteins and their functions. Among the proposed approaches to tackle this problem, artificial neural networks (ANN) are considered as one of the most successful methods that widely used in the field of PSSP. Recently, many efforts have been devoted to modify, improve and combine this technology with other machine learning methods in order to get better results. In this paper, we have proposed an ensemble method which combines the outputs of four feed-forward neural networks. In each network one of the machine learning approaches has been applied in order to solve the class imbalance problem of protein secondary structure classification. The experimental results on RS126 data set show that our ensemble system has better performance compare to the best individual classifier. The results also reveal that the proposed system yields significant improvement in prediction accuracy of beta-sheet structure and a more balanced classification of three secondary structures.

KEYWORDS

Protein Secondary Structure Prediction (PSSP), Artificial Neural Network (ANN), α-helix, β-sheet, Coil, Position-Specific Scoring Matrix (PSSM) Profile, Genetic Algorithm

1. INTRODUCTION

Proteins are the main building blocks and functional molecules of the cell and play a key role in almost all biological processes. They take part in maintaining the structural integrity of the cell, transport and storage of small molecules, catalysis, regulation, signaling and the immune system [1]. Proteins are large, complex molecules consist of long amino acid chains. Different chemical properties of 20 amino acids cause the protein chains to fold up into complex shapes, and the shape of a protein determines its function.The secondary structure prediction is an intermediate step of deducing the 3D structure of proteins. Secondary structure is the local spatial arrangment of its polypeptide backbone ignoring the conformation of the individual sidechains (R groups). Secondary structures are held together by hydrogen bonds. The PSSP problem aims at predicting

DOI : 10.5121/ijaia.2012.3602

each amino acid in a protein sequence as α-helix (H), β-sheet (E) or neither (C) from its primary structure or indeed, the linear sequence of its amino acid structural units. X-ray crystallography, nuclear magnetic resonance (NMR) spectography and electron microscopy are in vitro methods, used to determine the 3D structure of proteins. Though they increasingly being applied in a high-troughput manner, but these methods are time consuming, expensive and not applicable to all proteins. Instead, comutational methods are widly used to predict the secondary structures as a step toward the prediction of 3D structure of proteins. During the last decades, many computational techniques have been proposed in the litrerature to tackle this problem. The earliest approaches for secondary structure prediction considered just single amino acid statistics and properties, and were limited to a small number of proteins with solved structures. While these early methods are not state-of-the-art, they are the basis of many subsequent approaches [1]. some of the most well-known early secondary structure prediction methods are being the Chou-Fasman method which uses a combination of statistical and heuristic rules [2], GOR method on the basis of information theory framework [3] and Lim method [4] as a stereochemical rule-based approach for predicting secondary structure in globular proteins. Since the secondary structure of each amino acid is affected by its neighbors through the interactions between the constituent amino acids along a protein chain, other methods try to consider higher-order intractions between residues. Information theoretic approaches such as an extension of original GOR method is proposed in [5] which considers the high-order residue interactions. Nearest neighbor approaches which incorporate local dependencies for predicting secondary structure of a residue [6,7], consider a window of residues surrounding it and classify each test residue according to the classification of neighbor residues in training samples. Neural network is another method to consider higher-order interaction between residues [8], [9], [10]. The first attempts to use neural networks for PSSP were made by Qian and Sejnowski [8]. They used a fully connected perceptron with at most one hidden layer and gained the accuracy of 64.3% that was more accurate than other priveous methods. Their work was followed by others in various ways in order to improve the prediction accuracy by applying sophisticated network structures [11], [12], [13], [14]. Further improvement in preformance of PSSP were also acheived by exploiting evolutioary information via multiple sequence alignments (MSAs) profiles [10], position-specific score matrices (PSSM) [15], homology detection using hidden markov models [16] and PSI-BLAST [17]. Other approaches have been applied to PSSP are support vector machines (SVM) [18], [19] and ensemble methods that combine some machine learning approaches [20] via the majority voting or weighted majority voting techniques. These methods could optain up to a 3% improvement in Q_3 accuracy over the best individual method. In this paper, we have used ensemble method which combines the outputs of four feed-forward neural networks. Existing sampling methods and a tree-based tertiary classifier have been used to overcome class imbalance problem of each network. The results of ensemble members are combined using three different methods, simple majority voting (SMV), weighted majority voting (WMV) and using genetic algorithm to find the weights named genetically weighted majority voting (GWMV). The weights of each classifier in WMV method are specified using prediction accuracy of corresponding classifier on a validation set. Each solution in GWMV method is also evaulated with prediction accuracy of ensemble result on a validetion set according to the weights specified with that solution. The weights which are proposed by genetic algorithm will result in higher overal accuracy.

The remainder of this paper is organized as follows: section 2 is about related materials which describes dataset, secondary structure assignment, evaluation methods and PSSM generation. Our proposed method is described in section 3. Section 4, presents and analyzes the exprimental results on RS126 dataset and finally, the conclusions are drawn in section 5.

2. RELATED MATERIALS

In this section we describe RS126 data set, secondary structure assignment methods, Measures of prediction accuracy and PSSM generation details.

2.1. Data Set Description

The dataset we used in this work is the RS126 set which was proposed by Rost and Sander [10]. It contains 126 non-homologous globular proteins which no pairs of proteins in the set have more than 25% similarity over a length of more than 80 residues [10]. RS126 contains 23,346 amino acids with 32% α-helix, 23% β-sheet and 45% coil. Average sequence length of all the proteins in this set is 185. This dataset was used in many researches in protein secondary structure prediction field [14], [21].

2.2. Secondary Structure Assignment

Given the 3D atomic coordinate of a protein structure, there are several methods to assign its secondary structures including dictionary of secondary structure of proteins (DSSP) [22], STRuctural IDEntification (STRIDE) [23] and DEFINE [24]. Since the secondary structure assignment of each residue is not completely well-defined, these methods often disagree on their assignments. For example, DSSP and STRIDE differ on approximately 5% of residues [1]. This inconsistency justifies the need for a certain and standard assignment method that could be used. The method was adopted here is DSSP as the standard algorithm and the most widely used secondary structure definition method. The DSSP specifies eight secondary structure classifications: H (α-helix), E (β- strand), G (3_{10} helix), I (π-helix), B (bridge), T (turn), S (bend) and C (other residues). The eight possible secondary structure specified by DSSP can be reduced to 3 classes [25]:

1. $\{H,I,G\} \rightarrow H(Helix)$, $\{E,B\} \rightarrow B(BetaSheet)$, $\mathrm{Rest}\{S,T,C\} \rightarrow C(Coil)$

2. $\{H,G\} \rightarrow H(Helix)$, $\{E\} \rightarrow B(BetaSheet)$, $\mathrm{Rest}\{S,T,B,I,C\} \rightarrow C(Coil)$

3. $\{H\} \rightarrow H(Helix)$, $\{E\} \rightarrow B(BetaSheet)$, $\mathrm{Rest}\{G,S,T,B,I,C\} \rightarrow C(Coil)$

4. $\{H,G\} \rightarrow H(Helix)$, $\{E,B\} \rightarrow B(BetaSheet)$, $\mathrm{Rest}\{S,T,I,C\} \rightarrow C(Coil)$

In this work we applied method 1.

2.3. Measures of Prediction Accuracy

We have used two measures to evaluate the prediction accuracy of the methods. The three state accuracy (Q_3) is defined as the percent of residues that have been predicted correctly:

$$Q_3 = \frac{n_H + n_E + n_C}{N_T} \quad (1)$$

Where n_H, n_E, n_C are the number of correctly predicted residues of type H, E and C, respectively and N_T is the total number of residues in dataset. Although Q_3 is a concise and useful measure to compare different methods, but for known protein structures it has shown that residues are approximately 30% in helices, 20% in sheets and 50% in coil. Because of this imbalanced characteristic of PSS datasets, Q_3 does not convey useful types of information. For example, a classifier that always predicts C has a Q_3 accuracy of 50%. It does not also indicate whether one type of structure is predicted more successfully than another [1], so we have also used Matthews correlation coefficient (MCC) [26] for each of the three secondary structures to judge the quality of prediction. This measure would be a value between −1 and +1 where +1 represents a perfect

prediction, 0 no better than random prediction and −1 shows total disagreement between prediction and observation. So, coefficients closer to +1 represent better prediction. The Matthews correlation for a particular state $i \in \{H, E, C\}$ is defined as:

$$c_i = \frac{TP_i \times TN_i - FP_i \times FN_i}{\sqrt{(TP_i + FP_i) \times (TP_i + FN_i) \times (TN_i + FP_i) \times (TN_i + FN_i)}} \quad (2)$$

Where TP_i, TN_i, FP_i and FN_i are the number of correctly predicted (true positives), correctly rejected (true negatives), incorrectly predicted (false positives), incorrectly rejected (false negatives) residues, respectively and C_i indicates Matthews correlation coefficient for each three classes H, E and C.

2.4. PSSM Generation

To obtain Position-Specific Scoring Matrix (PSSM) profile for each protein sequence, we performed PSI-BLAST (Position-Specific Iterative Basic Local Alignment Search Tool) [17] against a non-redundant sequence (NR) data base. BLAST is the most widely used sequence similarity tool. PSI-BLAST is an iterative database searching method that uses homologous proteins found in an iteration to build a profile to be used for searching in the next iteration. This profile incorporates sequence weighting so that several closely-related homologs detected in the database do not overwhelm the contribution of more remote homologs [1]. The detected homologous proteins are then used as input into the neural network via the PSSM profile provided by PSI-BLAST. E-value threshold for inclusion is set to 0.001 and number of iteration to 3. The PSSM for each protein sequence has 20×L elements where L is the length of target sequence and each element represents the log-likelihood of particular residue substitution based on a weighted average of BLOSUM62 [27].

3. PROPOSED METHOD

In this section we first describe class imbalance problem which is a serious problem in machine learning and some approaches to tackle this problem. Then each ensemble member and three voting combination methods which we have used in this paper are introduced.

3.1. Class Imbalance Problem

When the number of training samples of one class is significantly fewer than other classes, class imbalance problem occurs [28]. Neural networks and most other machine learning algorithms aim to optimize the overall classification accuracy. So, these methods work well with balanced data sets. But when the data set is imbalanced the learning algorithm tends to be biased towards the majority class and the minority class samples are more likely to be misclassified [28]. In protein secondary structure prediction, this problem is an intrinsic property. For example RS126 data set contains 32% α-helix, 23% β-sheet and 45% coil. So most classifiers used in PSSP have tendency to predict a residue as coil (which is the majority class) to obtain a higher overall accuracy and have weak prediction for beta-sheet structure (which is the minority class). With these classifiers minority class is less important than the majority class.

3.2. Some Approaches to Solve Class Imbalance Problem

In this part some exiting machine learning approaches which we used to solve class imbalance problem are described.

3.2.1. Sampling

Sampling is one of the common approaches to class imbalance problem. In this approach the prior distribution of the minority and majority class are modified in order to obtain a more balanced number of samples in each class [28]. Two basic sampling methods are under-sampling and over-sampling:

Under-sampling extracts a smaller set of majority instances while preserving all the minority instances. A drawback with under-sampling is that by discarding instances of majority class some informative instances may be lost and classifier performance degrades [28]. Over-sampling increases the number of minority instances by replicating them [29], [30]. The advantage is that no information is lost. However it has its own drawbacks. For instance over-sampling leads to a higher computational cost [28].

3.2.2. Tree-based Tertiary Classifier

The distribution of three secondary structure class instances shows that the number of residues with coil structure is approximately equal to the number of residues with non-coil structures (alpha-helix and beta-sheet). For example 45% of RS126 data set residues are coil (C) and 55% are alpha-helix (H) and beta-sheet (E). So using the tree-based tertiary classifier which is shown in Figure 1 can help to have more balanced training set.

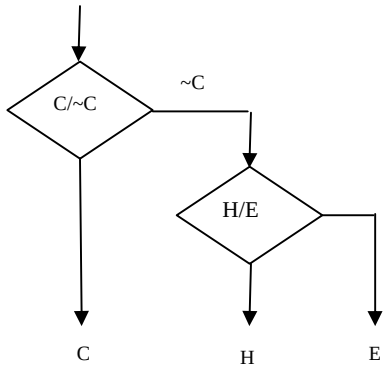

Figure 1. Tree-based tertiary classifier

3.3. Description of Ensemble Members

In this section we describe the structure of four classifiers which are ensemble members. All classifiers are based on a feed-forward neural network with one hidden layer, 25 neurons in hidden layer and PSSM profiles as their input. Since the optimum length of window according to the RS126 set is 13 [10], [31], in each network we used a sliding window of size 13 which moves through the protein sequence and the output of the network is attained for the residue in the middle of the window. As a result, the input layer includes 13×20=260 neurons (i.e., 13 rows of PSSM including 20 elements are concatenated).

- In first network under-sampling has been used. In this method N_H-N_E samples of class H and N_C-N_E samples of class C are selected randomly and discarded from training set in order to have equal number of residues of each class. (N_H, N_E and N_C indicate the number of residues of class H, E and C in training set respectively). So the number of samples in each class got equal to N_E.

- In second network over-sampling has been applied. N_C-N_H samples of class H and N_C-N_E samples of class E are selected randomly and replicated in training set (in the case that N_C-N_E is larger than N_E, all samples of class E are replicated). So the number of samples in each class in this case gets equal to N_C.

- In third classifier under-sampling and over-sampling methods were combined. N_C-N_H randomly selected samples of class H are discarded and N_H-N_E samples of class E are selected randomly and replicated. So the number of samples in each class gets equal to N_H.

- Fourth classifier is a tree-based tertiary classifier. Two binary classifiers which are feed-forward neural networks are cascaded according to Figure 1 in order to have an approximately balanced training set. Unlike previous classifiers the number of samples of three classes is not exactly equal.

3.4. Voting Combination Methods

In this part we introduce three voting combination schemes which we have used in this paper to combine ensemble member outputs.

3.4.1. Simple Majority Voting (SMV)

Suppose we have N samples, C classes and M classifiers. Each classifier will assign a class to each sample according to its prediction. Final prediction for each sample is the class which gains the majority vote with classifiers.

Let V_c indicates the vote is assigned to class c for sample x. we can formulate this as follows:

$$V_c = \sum_{i=1}^{M} p_i(c) \tag{1}$$

Where $p_i(c)$ is 1 if class c is predicted for sample x with classifier i, otherwise it is equal to 0.

Finally the predicted class for sample x is obtained according to the following formula.

$$\begin{aligned} &p(x) = \arg\max(V_c) \\ &c = \{1,\ldots, C\} \end{aligned} \tag{2}$$

3.4.2. Weighted Majority Voting (WMV)

In this method the decision of a more qualified classifier is more important in the vote. For example in this work we have weighted the decision of each classifier by its prediction accuracy on a validation set. So V_c in formula (1) can be redefined as follows:

$$V_c = \sum_{i=1}^{M} w_i \times p_i(c) \tag{3}$$

Where w_i is weight of classifier i. Final prediction for sample x is obtained according to formula (2) with this modification on term V_c. In other words the class which obtains the greatest total weight is assigned to example x.

3.4.3. Genetically Weighted Majority Voting (GWMV)

In this method we have used genetic algorithm to find the weight of each classifier. As shown in Figure 2 Real-valued chromosomes of length 4 (the number of classifiers) are used to represent solutions where each gene of a chromosome corresponds to the weight of a classifier. The fitness is assigned to each solution is the prediction accuracy of ensemble system on a validation set (which is obtained according to the weights of ensemble members in corresponding chromosome). The function for creating initial population is feasible population which creates a random initial population that satisfies all bounds (i.e., held in [0,1]). The fitness scaling function scales the raw scores according to their ranks in the sorted scores. The selection function of genetic algorithm to specify how parents are chosen for the next generation is stochastic uniform. In this method, parents are laid out in a line in which each parent corresponds to a section of the line of length proportional to its scaled value. The algorithm moves along the line in steps of equal size. At each step, the algorithm allocates a parent from the section it lands on. Crossover function is scattered which creates a random binary vector and selects the genes where the vector is a 1 from the first parent, and the genes where the vector is a 0 from the second parent, and combines the genes to form the child. Mutation function is Gaussian which adds a random number is taken from a Gaussian distribution with mean 0 to each entry of the parent vector.

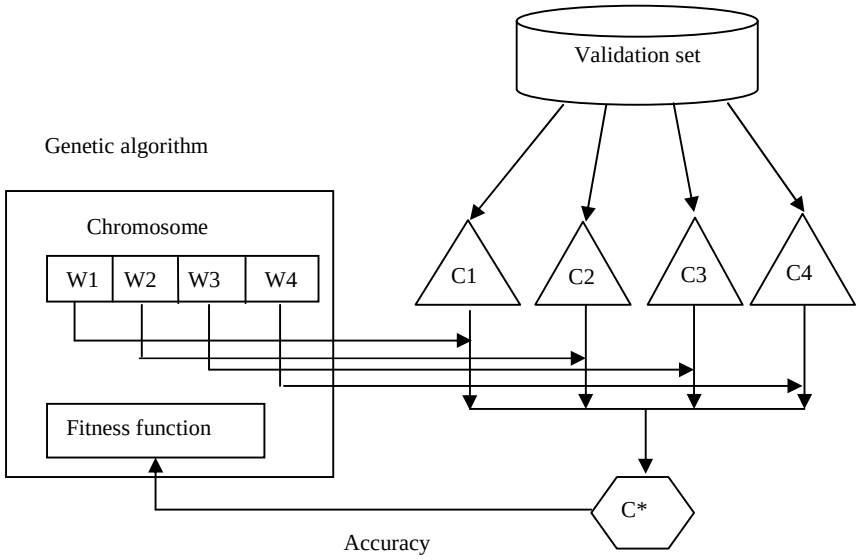

Figure 2. Using genetic algorithm to find the weights of classifiers

Moreover since the length of consecutive helices, H, and strands, E, should be at least three and two, respectively, we also have used the filtering method proposed by Salamov and Solovyev [32] for each individual classifier result and also ensemble results to make the predictions more realistic. So the following modifications are applied to the final output in order to exclude physically unlikely structures:

$$[EHE]\rightarrow[EEE],\ [HEH]\rightarrow[HHH],\ [HCH]\rightarrow[HHH],\ [ECE]\rightarrow[EEE],\ [HEEH]\rightarrow[HHHH]$$

Also, all helices of length one or two and all strands of length 1 are converted to coils, C.

4. EXPERIMENTAL RESULTS AND DISCUSSION

In this section to evaluate our method, we have used seven-fold cross-validation on the RS126 set. In this method the dataset is spitted to seven equal parts A, B, …, G and each part is in turn left out of the training set and used as test set. So, the networks are trained using 108 proteins and the rest 18 proteins are left for test in each turn. 10% of this training set is set aside for validation set which is needed for finding weights of ensemble members in WMV and GWMV voting schemes. First individual classifiers are evaluated and their results compare with the results of a neural network which was trained with imbalanced training set. Then a comparison is made between three combination methods. All results are after applying the filtering method.

4.1. Comparison of Individual Classifiers

Table 1 illustrates the prediction accuracy and MCCs for each ensemble member and a NN with imbalanced training set (the results were averaged on seven test groups). In this table, Q_H, Q_E and Q_C are the percentage of correctly predicted residues observed in class E, H and C, respectively.

Table 1. The average accuracies obtained for each classifier.

Accuracies / Training set	Q_H	Q_E	Q_C	Q_3	C_H	C_E	C_C
NN (Imbalanced training set)	68.07	52.78	87.20	73.33	0.65	0.54	0.51
NN (under-sampling)	69.76	68.07	77.48	73.02	0.64	0.56	0.49
NN (over- sampling)	71.72	68.90	77.44	**73.75**	0.66	0.57	0.51
NN (under-sampling and over-sampling)	71.05	69.38	77.38	73.73	0.65	0.56	0.51
NN (tree-based)	75.78	67.85	74.84	73.51	0.64	0.55	0.52

The results show that the overall prediction accuracy of network is decreased when under-sampling is used. Because in this method some useful information contained in ignored samples of majority class is lost and the performance of the network degrades. But this method has significant improvement in prediction of beta-sheet which is the minority class compare to NN with imbalanced training set. In three other classifiers the performance of the network is improved. Since in over-sampling no information is lost, better performance has been obtained using this method. As shown in Figure 3 all of four methods could decrease misclassification tendency of minority class and prediction ability of the network for three classes is more balanced (when imbalanced training set is used the difference between prediction accuracy of beta-sheet and coil is very high).

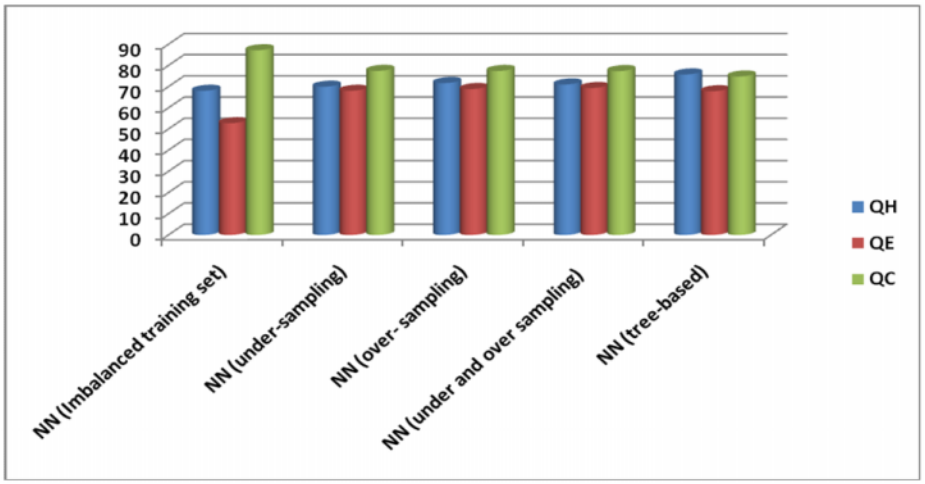

Figure 3. Q_H,Q_E and Q_C of individual neural networks

4.2. Evaluating Ensemble Method Using Three Voting Schemes

Table 2 reports the results of ensemble method using three methods for combining the outputs of ensemble members and compares them with the best ensemble member.

Table 2. The average accuracies obtained for voting schemes.

Accuracies / Voting scheme	Q_H	Q_E	Q_C	Q_3	C_H	C_E	C_C
SMV	74.85	72.78	75.32	74.66	0.67	0.58	0.53
WMV	72.43	69.76	78.43	74.64	0.67	0.58	0.53
GWMV	72.61	70.25	78.56	**74.90**	0.67	0.58	0.53
Best Ensemble member	71.72	68.90	77.44	73.75	0.66	0.57	0.51

We can justify that using genetic algorithm to find the weights of classifiers has the best overall prediction accuracy. Figure 4 visualizes comparison between these three ensembles and the best ensemble member. This figure shows that compare to the best classifier, prediction accuracies for three classes and overall accuracy improved with ensemble method for three combination schemes except for Q_C in simple majority voting. Matthews correlation coefficients (MCCs) for each of the three secondary structures are also better than the best classifier. Simple majority voting has the best accuracy for prediction of alpha-helix and beta-sheet predictions. Another comparison which was made in Figure 5 shows that simple majority voting has the most balanced prediction for three classes H, E and C and the best response to our method for solving class imbalance problem to avoid giving more importance to coil structure compare to alpha-helix and beta-sheet.

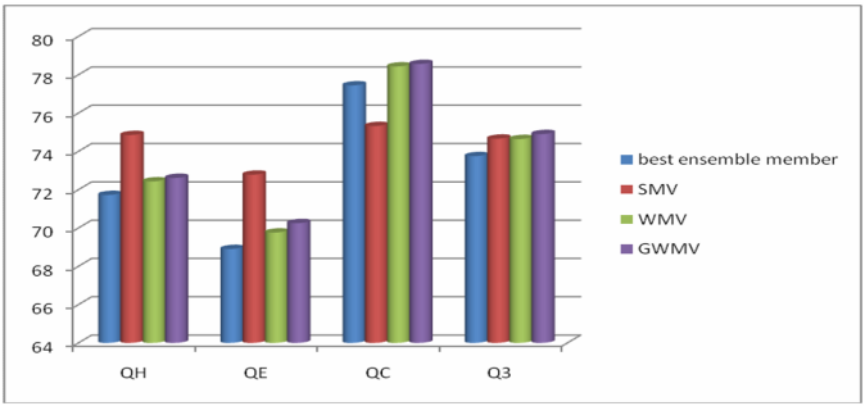

Figure 4. The comparison of ensemble methods and the best ensemble member accuracies

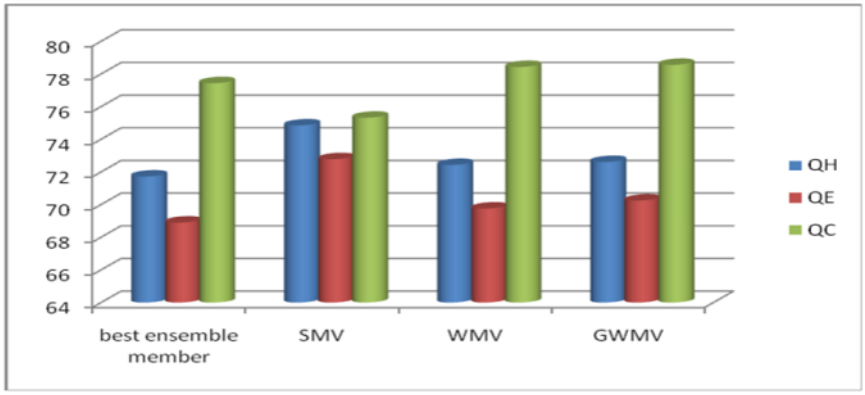

Figure 5. How much prediction accuracy of H,E and C are balanced in each method

5. CONCLUSIONS

Most machine learning algorithms which are used in protein secondary structure prediction suffer from class imbalance problem. In this work we used ensemble method and combined the results of four feed forward neural networks. Three of them trained with balanced training sets which were obtained using sampling methods (under-sampling, over-sampling and a combination of both methods) and in forth classifier a tree-based structure was used in order to have more-balanced training set. Results showed that when a neural-network is trained with imbalanced set, there is large difference between prediction accuracy of coil structure (majority class) and beta-sheet structure (minority class). In other words this network gives more importance to prediction of coil structures compare to alpha-helix and beta-sheet. But we had more balanced classification of these three classes in four classifiers with sampling methods and tree-based structure. The worst and best overall accuracy was obtained by under-sampling and over-sampling method respectively. This is because of discarding some informative samples in under-sampling method, but with over-sampling no information is lost. We used three voting schemes to combine the outputs of ensemble members (simple majority voting, weighted majority voting and using genetic algorithm to find the weights). Highest overall accuracy (Q_3) was obtained using genetic algorithm and simple majority voting had the most balanced classification of H, E and C. So using machine learning approach to solve class imbalanced problem can help to have approximately a fair prediction of three secondary structures and with ensemble of these approaches more accurate prediction is obtained compare to the best classifier.

REFERENCES

[1] M. Singh, "Predicting Protein Secondary and Supersecondary Structure," in Handbook of Computational Molecular Biology by S. Aluru, Chapman & Hall/CRC, 2006.

[2] P. Chou and G. Fasman, "Prediction of protein conformation," Biopolymers, vol. 13, no. 2, pp. 211-215, 1974.

[3]J. Garnier, D. Osguthorpe, and B. Robson, "Analysis and implications of simple methods for predicting the secondary structure of globular proteins," Journal of Molecular Biology, vol. 120, pp. 97-120, 1978.

[4] V. I. Lim, "Algorithms for prediction of alpha helices and structural regions in globular proteins," Journal of Molecular Biology, vol. 88, pp. 873-894, 1974.

[5] J. Gibrat, J. Garnier, and B. Robson, "Further developments of protein secondary structure prediction using information theory: new parameters and consideration of residue pairs," Journal of Molecular Biology, vol. 198, pp. 425-443, 1987.

[6] K. Nishikawa and T. Ooi, "Amino acid sequence homology applied to prediction of protein secondary structure and joint prediction with existing methods," Biochimica et Biophysica Acta, vol. 871, no. 1, pp. 45-54, 1986.

[7] J. Levin, B. Robson, and J. Garnier, "An algorithm for secondary structure determination in proteins based on sequence similarity," FEBS Letters, vol. 205, no. 2, pp. 303-308, 1986.

[8] N. Qian and T. Sejnowski, "Predicting the secondary structure of globular proteins using neural network models," Journal of Molecular Biology, vol. 202, no. 4, pp. 865-884, 1988.

[9] L. H. Holley and M. Karplus, "Protein secondary structure prediction with a neural net," Proceedings of the National Academy of Sciences (USA), vol. 86, pp. 152-156, 1989.

[10] B. Rost and C. Sander, "Prediction of protein secondary structure at better than 70%," Journal of Molecular Biology, vol. 232, pp. 584-599, 1993.

[11] L. Han, 1. Cuil, H. Lin, Z. Ji, Z. Cao,Y. Li, and Y. Chen, "Recent progresses in the application of machine learning approach for predicting protein functional class independent of sequence similarity," Proteomics, vol. 6, pp. 4023-4037, 2006.

[12] A. Ceronia, P. Frasconia, and G. Pollastri, "Learning protein secondary structure from sequential and relational data," Neural Networks, vol. 18, pp. 1029-1039, 2005.

[13] J. Chen and N. Chaudhari, "Cascaded bidirectional recurrent neural networks for protein secondary structure prediction," IEEE Trans. Computational Biology and Bioinformatics, vol. 4, no. 4, 2007.

[14] J. Chen and N. Chaudhari, "Bidirectional segmented-memory recurrent neural network for protein secondary structure prediction," Soft Computing, vol. 10, pp. 315-324, 2006.

[15] D. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," Journal of Molecular Biology, vol. 292, 1999.

[16] K. Karplus, C. Barret, and R. Hughey, "Hidden Markov models for detecting remote protein homologies," Bioinformatics, vol. 14, pp. 846-856, 1998.

[17] S. Altschul, T. Madden, A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs," Nucleic Acids Research, vol. 25, pp. 3389-3402, 1997.

[18] S. Hua and Z. Sun, "A novel method of protein secondary structure prediction with high segment overlap measure: Support vector machine approach," Journal of Molecular Biology, vol. 308, pp. 397-407, 2001.

[19] M. Nguyen, J. Rajapkse, "Two-stage multi-class support vector machines to protein secondary structure prediction," Pac. Symp. Biocomput., vol. 10, pp. 346–57, 2005.

[20] R. King, M. Ouali, A. Strong, A. Aly, A. Elmaghraby, M. Kantardzic, and D. Page, "Is it better to combine predictions?," Protein Engineering, vol. 13, pp. 15-19, 2000.

[21] H. Kim and H. Park, "Protein Secondary Structure Prediction Based on an Improved Support Vector Machines Approach," Protein Engineering, vol. 16, pp. 553-560, 2003.

[22] W. Kabsch and C. Sander, "A dictionary of protein secondary structure," Biopolymers, vol. 22, pp. 2577-2637, 1983.

[23] D. Frishman and P. Argos. "Knowledge-based secondary structure assignment," Proteins: Structure, Function and Genetics, vol. 23, pp. 566-579, 1995.

[24] F. M. Richards and C. E. Kundrot, "Identification of structural motifs from protein coordinate data: secondary structure and first-level supersecondary structure," Proteins, vol. 3, pp. 71–84, 1988.

[25] P. Baldi and S. Brunak, Bioinformatics: The machine learning approach, Cambridge, MA, MIT Press, Second Edt., 2001.
[26] B. Matthews, “Comparison of the predicted and observed secondary structure of T4 phage lysozyme,” Biochimica et Biophysica Acta, vol. 405, pp. 442-451, 1975.
[27] S. Heniko and J. Henikoff, “Amino acid substitution matrices from protein blocks,” Proc. Natl. Acad. Sci. USA, vol. 89, pp. 10915–10919, 1992.
[28] S. L. Phung, A. Bouzerdoum and G. H. Nguyen, “Learning pattern classification tasks with imbalanced data sets”, 2009.
[29] N. V. Chawla, K. Bowyer, L. Hall and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling TEchnique,” Artificial Intelligence Research, pp. 321-357, 2002.
[30] N. Japkowicz and S. Stephen, “The Class Imbalance Problem: A Systemic Study,” Intelligent Data Analysis, pp. 429-449, 2002.
[31] M. Mirto, M. Cafaro, S. Luigi Fiore, D. Tartarini, and G. Aloisio, "A grid-enabled protein secondary structure predictor," IEEE Trans. Nanobioscience, vol. 6, no. 2, 2007.
[32] A. A. Salamov and V. V. Solovyev, “Prediction of protein secondary structure by combining nearest neighbor algorithms and multiple sequence alignments,” Journal of Molecular Biology, vol. 247, pp. 11–15, 1995.