Protein Secondary Structure Classifiers Fusion Using OWA

Majid Kazemian[1], Behzad Moshiri[1], Hamid Nikbakht[2], and Caro Lucas[1]

[1] Control and Intelligent Processing Center of Excellence, Electrical and Computer Eng. Department, University of Tehran, Tehran, Iran
am.kazemian@ece.ut.ac.ir, moshiri@ut.ac.ir
[2] Laboratory of Biophysics and Molecular Biology, Institute of Biochemistry and Biophysics, University of Tehran, Tehran, Iran

Abstract. The combination of classifiers has been proposed as a method to improve the accuracy achieved by a single classifier. In this study, the performances of optimistic and pessimistic ordered weighted averaging[1] operators for protein secondary structure classifiers fusion have been investigated. Each secondary structure classifier outputs a unique structure for each input residue. We used confusion matrix of each secondary structure classifier as a general reusable pattern for converting this unique label to measurement level. The results of optimistic and pessimistic OWA operators have been compared with majority voting and five common classifiers used in the fusion process. Using a benchmark set from the EVA server, the results showed a significant improvement in the average Q3 prediction accuracy up to 1.69% toward the best classifier results.

1 Introduction

There are three main classes of secondary structural elements in proteins named as alpha helices (H), beta strands (E) and irregular structures (turns or coils that are shown as C), so prediction engines can be assumed as structural classifiers. In the process of predicting protein secondary structure classification, it is usual to use a variety of approaches that each of them has its own strengths and weaknesses [1, 2, and 3].

The reasons for combining the outputs of multiple classifiers are compelling, because different classifiers may implicitly represent different useful aspects of the problem, or of the input data, while none of them represents all useful aspects [4]. In this context, the idea of decisions fusion of several classifiers has been well explored. Information fusion techniques have been intensively investigated in recent years and their applications for classification domain have been widely tested [5]. Methods for fusing multiple classifiers can be classified according to the type of information produced by the individual classifiers. Classifiers can differ in both the nature of the measurements used in the classification process and in the type of classification produced. A Type I classification is a simple statement of the class, a Type II classification is a ranked list of probable classes, and a Type III classification assigns probabilities to classes [6]. These three types are known as *Abstract level* outputs, *Rank level* outputs and *Measurement level* outputs respectively. Most of protein secondary struc-

[1] Ordered Weighted Averaging (OWA)

J.L. Oliveira et al. (Eds.): ISBMDA 2005, LNBI 3745, pp. 338–345, 2005.

ture classifiers are type I classifier. The best known approach for consensus of type I classifiers is majority voting. In Majority voting approach there are some problems, for instance, where there is no majority winner, what the majority voter should do? There are some better techniques for classifier's outputs fusion but most of them need the results of type II or type III classifiers [7].

The results of five common protein secondary structure prediction engines of a benchmark dataset have been used for testing this fusion approach. The rest of the paper is organized as follows: the confusion matrix and the algorithm for converting type I classifier to type III classifier have been explained in section 2. Section 3 describes two simple OWA operators and demonstrates the application of these operators in the protein secondary structure classifiers fusion context. The classifiers and test dataset were introduced briefly in section 4. Section 5 presents the criteria of secondary structure prediction accuracy. Section 6 reveals the results of the fusion and finally the conclusion has been posed.

2 Measurement Level from Abstract Level Classifications

In this study, it is suggested that measurement level classifications could be created from the confusion matrix (a posteriori probabilities of each classification) of a Type I classifier. The assumptions are that, first, the behavior of the classifier is known and is characterized in a confusion matrix and, second, that the prior behavior or the classifier is representative of its future behavior. The larger the data set on which the classifier has been tested, the more thoroughly will the second assumption be true.

A confusion matrix is a matrix in which the actual class or a datum under test is represented by the matrix row, and the classification of that particular datum is represented by the confusion matrix column. The element M[i][j] gives the number of times that a class i object was assigned to class j. The diagonal elements indicate correct classifications and, if the matrix is not normalized, the sum of row I is the total number of elements of class I that actually appeared in the data set [8]. The columns of such a matrix can be used to convert Type I to Type III classification, which can then be sorted to yield the ranks (Type II classification).

Consider a classifier that produces only a single output class (Type I) and has been trained and tested on many thousands of data elements. During this process, the following confusion matrix was generated (assume there are three classes): (Table 1)

Table 1. Confusion Matrix for a classifier

	Classified as H	Classified as E	Classified as C
Actual Class H	1600	100	300
Actual Class E	200	1200	600
Actual Class C	100	500	1400

Each row sums to 2000, which was the number of elements of each class in the data set. Note that if there was not the same number of elements in each row, it must be normalize by the number of its elements. Now as an example, presume that this classifier issues a classification of H for a given input datum. From the first column of the confusion matrix, it can be seen that the most likely actual class is H, the second

most likely class is E, followed by C. This is a fair ranking of the possible classes based on the past of the classifier history. In other words, given a classification of H:

1600/1900 will be correct (class H)
200/1900 will actually be class E
100/1900 will actually be class C

This is the scheme suggested for converting abstract level classifications into measurement level.

3 Ordered Weighted Averaging

The Ordered Weighted Averaging Operators (OWA) were originally introduced by Yager to provide a means for aggregating scores associated with the satisfaction of multiple criteria, which unifies in one operator the conjunctive and disjunctive behavior [9]. An OWA operator of dimension n is a mapping $F: R^n \rightarrow R$ and is given by:

$$OWA(x_1, x_2, ..., x_n) = \sum_{j=1}^{n} w_j x_{\sigma(j)} \tag{1}$$

Where σ is a permutation that orders the elements: $x_{\sigma(1)} \leq x_{\sigma(2)} \leq .... x_{\sigma(n)}$. The weights are all non-negative ($w_i \geq 0$) and their sum equals to one ($\sum_{i=1}^{n} w_i = 1$).

This operator has been proved to be very useful, because of its versatility, The OWA operators provide a parameterized family of aggregation operators, which include many of the well-known operators such as the maximum, the minimum, the k-order statistics, the median and the arithmetic mean. In order to obtain these particular operators we should simply choose particular weights. The Ordered Weighted Averaging operators are commutative, monotone, idempotent, they are stable for positive linear transformations, and they have a compensatory behavior. This last property translates the fact that the aggregation done by an OWA operator always is between the maximum and the minimum. It can be seen as a parameterized way to go from the *min* to the *max*. In this context, a degree of maxness (initially called orness) was introduced in [9], defined by:

$$maxness(w_1, w_2, ..., w_n) = \frac{1}{n-1} \sum_{i=1}^{n} (n-i) w_i \tag{2}$$

We see that for the minimum, we have that *maxness*(1,0,…,0)=0 and for the maximum *maxness*(0, …,0,1)=1.

A simple class of OWA operators as exponential class of OWA operators was introduced to generate the OWA weights satisfying a given degree of maxness. The optimistic and pessimistic exponential OWA operators were introduced as follows [9]:

- Optimistic weights:

$$w_1=a;\ w_2=a(1-a);\ w_3=a(1-a)^2;\ ...;\ w_{n-1}=a(1-a)^{n-2};\ w_n=(1-a)^{n-1} \tag{3}$$

- Pessimistic weights:

$$w_1 = a^{n-1};\ w_2=(1-a)\,a^{n-2};\ w_3=(1-a)a^{n-3};\ \ldots;\ w_{n-1}=(1-a)a;\ w_n=(1-a) \quad (4)$$

Where parameter a (alpha) belongs to the unit interval, [0 1] and is related to orness value regarding the n.

Each protein secondary structure classifier outputs a label (H or E or C). Meanwhile a list of measured level classification (MLC) is constituted by using confusion matrix as descried in previous section $MLC(i) = \{W_i(H), W_i(E), W_i(C)\}$. This list shows the confidences of a classifier to its possible outputs. For example, consider the MLC of two classifiers:

	H	E	C
Classifier 1:	**0.7**	**0.2**	**0.1**
Classifier 2:	**0.3**	**0.5**	**0.2**

$W_i(H)$, $W_i(E)$ and $W_i(C)$ are fused by OWA operator separately for all of classifiers. After fusion process, the secondary structure of a certain amino acid is extracted from the Fused MLC, $\{W(H), W(E), W(C)\}$ as below:

$$PC = arg\ max\{W(H), W(E), W(C)\} \quad (5)$$

The general architecture of proposed Meta classifier is shown in Figure 1.

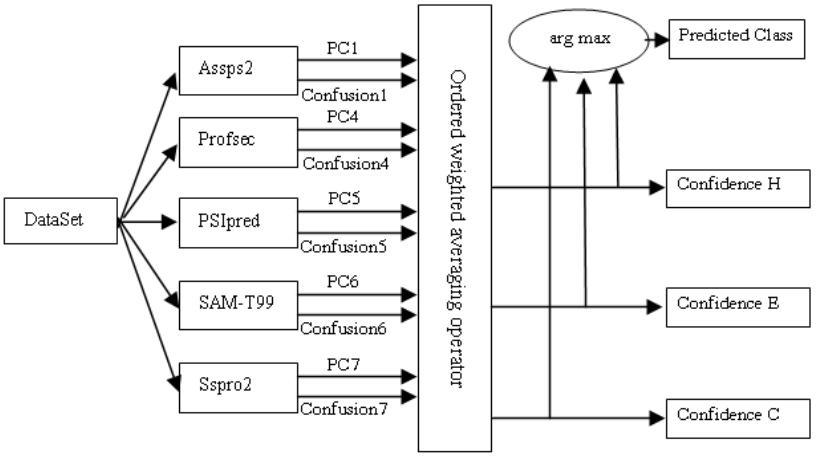

Fig. 1. Meta classifier schema-(PC: predicted class)

4 Experimental Evaluations

An experimental evaluation was carried out on EVA1 dataset. Novel test set which is provided by the datasets available from the real-time evaluation experiment [10], which compares a number of prediction servers on a regular basis using the sequences deposited in the PDB every week. In particular, we have used the dataset labeled "common1" published on 20/10/2002. Some information about the prediction method and location of five used prediction servers are shown in Table 2. For more information about these servers, see the references.

Table 2. Secondary structure prediction servers on Internet

Secondary Structure Prediction Servers		
Server	**Location**	**Prediction method**
Apssp2 [11]	Institute of Microbial Technology, INDIA	EBL* + Neural network
Profsec [12]	Columbia University, USA	Profile-based Neural network
PSIPRED [13]	University College London, UK	Neural network
SAM-T99 [14]	University of California, Santa Cruz, USA	Hidden Markov Model
SSPro2 [15]	University of California, Irvine, USA	Recurrent Neural Network

* Example Based Learning

To choose the parameter *alpha* in optimistic and pessimistic OWA, an iterative approach is used. In this purpose, a dataset is divided into three parts randomly; one of them is assigned for training and the rest for testing purpose. In training set, the *alpha* value was increased from zero to one by step 0.01 and consequently the prediction accuracy was calculated. The *alpha* value, in which the prediction accuracy was the maximum there, is selected as a needed parameter.

5 Accuracy of Predicting Secondary Structure Content

- *Prediction accuracy matrix:*
  M_{ij} = number of residues observed in state I and predicted in state j, with i, j {H, E, C}
  Note: the total number of residues observed in state i is:

$$obs_i = \sum_{j=1}^{3} M_{ij} \quad , \; with \; j \in \{H, E, C\} \tag{6}$$

  Note: the total number of residues predicted in state i is (helix, strand, other)

$$prd_i = \sum_{j=1}^{3} M_{ij} \quad , \; with \; j \in \{H, E, C\} \tag{7}$$

  The total number of residues is simply:

$$N_{res} = \sum_{i} obs_i = \sum_{i} prd_i = \sum_{i,j} M_{ij} \tag{8}$$

- *Three-state prediction accuracy: Q_3*
  The three-state per residue accuracy Q_3 becomes:

$$Q_3 = 100 \times \frac{1}{N_{res}} \times \sum_{i=1}^{3} M_{ij} \tag{9}$$

- *Per-state percentages:*
  To define the accuracy for a particular state (helix, strand, other), two possible variants could be considered. As a result, the following questions could be raised up:
  How many observed helix residues (strand or coil) were correctly predicted? Given are the correctly predicted residues as percentage of all residues OBSERVED in a particular state (% obs).

$$Q_i^{\%obs} = 100 \times \frac{M_{ij}}{obs_i} \tag{10}$$

How many predicted helix (strand or coil) residues were correctly predicted? Given are the correctly predicted residues as percentage of all residues PREDICTED in a particular state (% prd)

$$Q_i^{\%prd} = 100 \times \frac{M_{ij}}{prd_i} \tag{11}$$

6 Results

Statistics of the predictions performed by the five selected servers (described in previous section) are presented in Table3. The results demonstrate that the best classifier for this dataset is PSIPRED. Although its predictions of the secondary structure are of the highest accuracy, it has been further improved by our meta-classifier. Improvements in terms of the accuracy of the OWA based meta-classifier are presented in Table 5. The results show that the OWA based meta-classifier has absolute improvement of 1.69% compared to PSIPRED. The most interesting results have been achieved for β strand prediction. PSIPRED predicts accurately 68.25% of the cases while OWA gets 73.08% giving an improvement of 4.83%. In addition, there is 5.78% improvement in helix structure prediction with -3.75% changes in coil structure classification.

Comparison between MV and OWA shows that the OWA based meta-classifier has improvement of 0.79% compared to MV, which is not very interesting in first look, but with deeper look into the results, we found that the OWA caused an improvement of 6.71% in β strand and 4.14% in helix structure. Recognition rate in helices and strands is more important than coils because due to general definition of protein secondary structures, each residue that is not in helix or strand structure will be posed in coil structure.

Table 3. The results of EVA1 dataset prediction by five common selected engines

	Q_3	$Q_h^{\%obs}$	$Q_e^{\%obs}$	$Q_c^{\%obs}$	$Q_h^{\%prd}$	$Q_e^{\%prd}$	$Q_c^{\%prd}$
apssp2	74.49	78.00	65.65	77.01	79.4	76.38	70.08
Profsec	74.71	75.38	74.48	74.05	82.95	71.29	70.76
Psipred	74.78	78.53	68.25	75.67	79.21	75.32	70.99
samt99_sec	74.63	82.60	63.12	75.06	77.90	79.37	69.74
sspro2	73.58	78.14	62.79	76.45	78.70	76.55	68.35

Table 4. The calculated maxness value and corresponding alpha value (achieved by the Fig.1 of [14]) of OWA

	alpha	maxness
Majority Voting	*	*
OWA-optimistic	0.3	0.6
OWA-pessimistic	0.7	0.4

Table 5. The results of Majority Voting and OWA operators

	Q_3	$Q_h^{\%obs}$	$Q_e^{\%obs}$	$Q_c^{\%obs}$	$Q_h^{\%prd}$	$Q_e^{\%prd}$	$Q_c^{\%prd}$
MV	75.68	80.17	66.37	77.68	80.28	78.94	70.58
OWA-optimistic	76.47	84.31	73.08	71.92	78.63	75.51	75.03
OWA-pessimistic	76.47	84.31	73.08	71.92	78.63	75.51	75.03

7 Conclusions and Future Research

Combining protein secondary structure classifiers requires a uniform representation of their decisions with respect to an observation. Confusion matrix is a well-known evaluator for each type of classifier which is used here as a general reusable pattern for fusion of protein secondary structure classifiers. Such a general assessor could be used in better weighting assignment in all fusion approaches. Moreover, a confusion matrix is used for converting Type I classifier to Type III classifier. In these types of classifiers, heuristic functions or theories for decision fusion may be more applicable.

The performance of a Meta classifier system can be better than each individual classifier; also, such systems can provide a unified access to data for users.

There are still open issues ahead:

- To obtain the number of classifiers those are required to achieve a desired accuracy.
- To obtain better identifiers to convert Type I classification to Type III classification.
- To publish the protein secondary structure meta-classifiers as an open-access web service.

References

1. Argos P, Hanei M, Garavito RM. The Chou-Fasman secondary structure prediction method with an extended database. FEBS Lett. 1978 Sep 1;93(1):19-24.
2. Cai YD, Liu XJ, Chou KC. Prediction of protein secondary structure content by artificial neural network. J Comput Chem. 2003 Apr 30;24(6):727-31.
3. Kim S. Protein beta-turn prediction using nearest-neighbor method. Bioinformatics. 2004 Jan 1;20(1):40-4.
4. T.H. Ho, J. J. Hull, S.N. Srihari, "Decision Combination in Multiple Classifier System", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, pt. 1, pp. 66-75, 1994.
5. Dymitr Ruta and Bogdan Gabrys, An Overview of Classifier Fusion Methods, computing and Information Systems, 7 (2000) p.1-10.
6. L. Xu, A. Krzyzak, and C.Y. Suen, "Methods of Combining Multiple Classifiers and their Applications to Handwriting Recognition", IEEE Trans. SMC, vol. 22, No. 3, 1992. pp 418-435.
7. Robles V, Larranaga P, Pena JM, Menasalvas E, Perez MS, Herves V, Wasilewska A., "Bayesian network multi-classifiers for protein secondary structure prediction.", Artif Intell Med. 2004 Jun;31(2):117-36.
8. J. R. Parker: Rank and response combination from confusion matrix data. Information Fusion 2(2): 113-120 (2001).

9. Yager, R. R., On ordered weighted averaging aggregation operators in multi-criteria decision making, IEEE transactions on Systems, Man and Cybernetics 18, 183-190, 1988.
10. B. Rost and V.A. Eyrich. EVA: large-scale analysis of secondary structure prediction. Proteins, 5:192–199, 2001.
11. G. P. S. Raghava, Protein secondary structure prediction using nearest neighbor and neural network approach. CASP4: 75-76, 2000.
12. B Rost, PROF: predicting one-dimensional protein structure by profile based neural networks. http://cubic.bioc.columbia.edu/predictprotein.
13. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. J. Mol. Biol. 292: 195-202.1999.
14. K Karplus, C Barrett, and R Hughey: Hidden Markov Models for Detecting Remote Protein Homologies, Bioinformatics, 14, 846-856, 1998.
15. G.Pollastri, D.Przybylski, B.Rost, P.Baldi, "Improving the Prediction of Protein Secondary Structure in Three and Eight Classes Using Recurrent Neural Networks and Profiles", Proteins, 47, 228-235, 2002.