A PROBABILISTIC NEURAL NETWORK APPROACH FOR PROTEIN SUPERFAMILY CLASSIFICATION

PV NAGESWARA RAO[1](NAGESH@GITAM.EDU), T UMA DEVI[1], DSVGK KALADHAR[1], GR SRIDHAR[2], ALLAM APPA RAO[3]

[1]GITAM University,[2] Endocrine Society, Visakhapatnam, [3]JNTU, Kakinada, India

ABSTRACT

The protein superfamily classification problem, which consists of determining the superfamily membership of a given unknown protein sequence, is very important for a biologist for many practical reasons, such as drug discovery, prediction of molecular function and medical diagnosis. In this work, we propose a new approach for protein classification based on a Probabilistic Neural Network and feature selection. Our goal is to predict the functional family of novel protein sequences based on the features extracted from the protein's primary structure i.e., sequence only. For this purpose, the datasets are extracted form Protein Data Bank(PDB), a curated protein family database, are used as training datasets. In these conducted experiments, the performance of the classifier is compared to other known data mining approaches / sequence comparison methods. The computational results have shown that the proposed method performs better than the other ones and looks promising for problems with characteristics similar to the problem.

Key words: *Probabilistic Neural Network, Classification, Feature Extraction, Bioinformatics*.

1. INTRODUCTION

Proteins are complex organic macromolecules made up of amino acids. They are fundamental components of all living cells and include many substances, such as enzymes, structural elements and antibodies, which are directly related with the functioning of an organism[1]. Hence the knowledge of the proteins biological actions (functions) is very important. Until recently, the functions of the proteins could be identified only by time-consuming and expensive experiments. However, in the post-genomic era, with the huge amount of available sources of information, new challenges arise in protein function characterization[2]. Moreover, computer based methods to assist in this process are becoming increasingly important. The need for faster sequence classification algorithms has been demonstrated by Cameron G *et al*.[3].

2. RELATED WORKS

Techniques used for biological sequence classification fall into two categories:

Similarity search: This approach is to classify unlabeled test sequences by searching for either global similarities or local similarities in the sequences. Global similarity search involves either pair-wise sequence comparison, or multiple sequence alignment. Local similarity search is to find patterns in sequences[4].

Machine Learning: This approach was surveyed by Haussler D[5]. Various machine learning techniques have been applied to biological sequences classification. For example, hidden Markov Model has been used in gene identification as well as protein family modeling. Neural Networks have been applied to the analysis of biological sequences[6].

3. THE PROPOSED METHOD

Feature Extraction: The majority of real-world classification problems require supervised learning where the underlying class probabilities and class-conditional probabilities are unknown, and each instance is associated with a class label. In real-world situations, relevant features are often unknown a *priori*. Therefore, many candidate

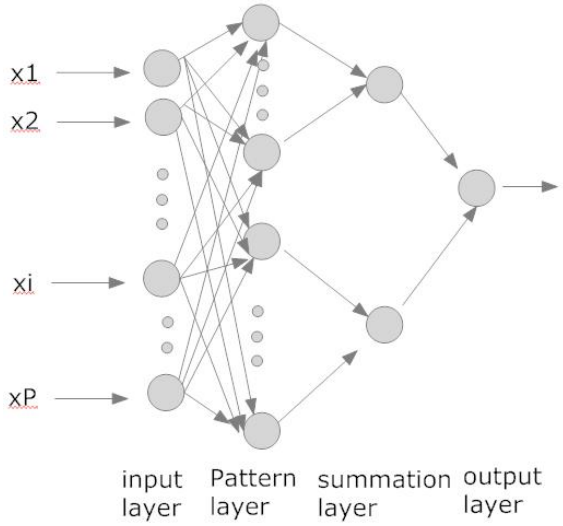
features are introduced to better represent the domain. Sometimes many of these features are either partially or completely irrelevant /redundant to the target concept. A relevant feature is neither irrelevant nor redundant to the target concept. An irrelevant feature does not affect the target concept. A redundant feature does not add anything new to the target concept. In many applications, the size of a dataset is so large that learning might not work as well before removing these unwanted features. Reducing the number of irrelevant/redundant features drastically reduces the running time of a learning algorithm and yields a more general concept. This helps in getting a better insight into the underlying concept of a real-world classification problem. *Feature selection* methods try to pick a subset of features that are relevant to the target concept[7]. The protein is represented as vector according to its features. The details of the extracted features are explained in section 4 of this paper.

Probabilistic Neural Network:

A Probabilistic Neural Network (PNN) is defined as an implementation of statistical algorithm called Kernel discriminate analysis in which the operations are organized into multilayered feed forward network with four layers: input layer, pattern layer, summation layer and output layer. A PNN is predominantly a classifier since it can map any input pattern to a number of classifications. Among the main advantages that discriminate PNN is: Fast training process, an inherently parallel structure, guaranteed to converge to an optimal classifier as the size of the representative training set increases and training samples can be added or removed without extensive retraining. Accordingly, a PNN learns more quickly than many neural networks model and have had success on a variety of applications. Based on these facts and advantages, PNN can be viewed as a supervised neural network that is capable of using it in system classification and pattern recognition[8]. The main objective of this paper is to describe the use of PNN in solving protein classification problem.

The architecture of PNN:

The PNN consists of nodes allocated in three layers after the inputs:



-- *pattern layer/unit:* there is one pattern node for each training example. Each pattern node/unit forms a product of the input pattern vector x(for classification) with a weight vector W_i, $Z_i=x.W_i$, and when perform a non linear operation on Zi before outputting its activation level to the summation node/unit. Here instead of the sigmoid activation function(back propagation algorithm), the nonlinear operation $\exp[(Z_i-1)/\sigma^2]$ is used. Both x and W_i are normalized to unit length, this is equivalent to using $\exp[-(W_i-x)^T(W_i-x)/2\sigma^2)]$.

-- *summation layer/unit*: each summation node/unit receives the outputs from pattern nodes associated with a given class. It simply sums the inputs from the pattern units that correspond to the category from which the training pattern was selected, $\sum_i\exp[-(W_i-x)^T(W_i-x)/2\sigma^2)]$.

-- *output layer/unit:* the output nodes/units are two-input neurons. These units product binary outputs, related to two different categories Ω_r, Ω_s $r\neq s$, r, s =1,2, ……., q, by using the classification criterion:

$\sum_i\exp[-(W_i-x)^T(W_i-x)/2\sigma^2)] > \sum_j\exp[-(W_j-x)^T(W_j-x)/2\sigma^2)]$

These units have only a single corresponding weight C, given by the loss parameters the prior probabilities and the number of training pattern in each category. Concretely, the corresponding weight is the ratio on a priori probabilities, divided by the ratio of samples and multiplied by the ratio of losses, C= $\frac{h_s l_s}{h_r l_r}\frac{n_r}{n_s}$. This ratio can be determined only from the significance of the decision. There were developed non parametric techniques for estimating univariate (or multivariate) probability density function (pdf) from random samples. By using the multivariate Gaussian approximation, that is a sum of multivariate Gaussian distributions centered at each training sample, we obtain the following form of the pdf

$$f_r(x) = \frac{1}{(2\pi)^{\frac{p}{2}}}\frac{1}{m}\sum_{i=1}^{m}\exp\left(-\frac{\left||x - x_i|\right|^2}{2\sigma^2}\right)$$

r = 1,2,…q, and i is the (vector) pattern number, m is the total number of training patterns, x_i is the i^{th} training pattern from category(class), Ω_r, p is the input space dimension and σ is an adjustable *smoothing* parameter using the training procedure. The network is trained by setting the W_i weight vector in one of the pattern units equal to each x pattern in the training set and then connecting the pattern unit's output to the appropriate summation unit[9][10].

4. EXPERIMENTS AND RESULTS

The datasets that were used in our experiments were created from the PDB protein sequence database. All complete non-ambiguous sequences of the three selected superfamily classes were processed.

The three classes collected and their sizes are esterase(195), lipase(155) and cytochrome(140). Each protein sequence is represented as vector of feature value calculated using different formula for each feature type. At a high level, the protein is represented as a sequence of 479 feature values.

Feature description	No. of features
Amino Acid distribution	20
Amino Acid distribution – two grams	400
Exchange group distribution	6
Exchange group distribution – two grams	36
Isoelectric point(pI)	1
Length of sequence	1
Molecular weight	1
Atomic composition	5
Hydrophobicity distribution – two grams	9

The two-gram features account for the majority of the attributes. They represent the frequencies of every consecutive "two-letter" sequence in the protein sequence. Two grams have the advantages of being length invariant, insertion/deletion invariant, not requiring motif finding and allowing classification based on local similarity. Exchange grams are similar but are based on a many-to-one translation of the amino acid alphabet into a six letter alphabet that represents six groups of amino acids, which represent high evolutionary similarity. Exchange groups used for this dataset are: $e1$ ={H,R,K}, $e2$ ={D,E,N,Q}, $e3$ ={C}, $e4$ ={A,G,P,S,T}, e5 ={I,L,M,V} and e6 ={F,Y, W}. The exchange groups are based on information from the point accepted mutations (PAM) matrix [11]. The pI, molecular weight, and atomic composition features are also calculated. These vectors are given as input and training sets for the Probabilistic Neural Network to identify the family membership of the query sequence. The measure used to evaluate the performance of the PNN classifier the *precision* (PR) is calculated as $PR = \frac{NumCorrect}{NumTotal}x100\%$ where *NumCorrect* is the number of test sequences classified correctly and *NumTotal* is the total number of test sequences. In general precision is a comprehensive measure in the sense that it

considers true positives, false positives, true negatives, false negatives and unclassified sequences. The *specificity, sensitivity, unclassified$_p$ and unclassified$_n$* are calculated as:

specificity = $(1-N_{fp}/N_{ng}) \times 100$ %
sensitivity = $(1-N_{fn}/N_{po}) \times 100$ %
unclassified$_p$ = $N_{up}/N_{po} \times 100$ %
unclassified$_n$ = $N_{un}/N_{ng} \times 100$ %

N_{fp} is the number of false positives, N_{fn} is the number of false negatives, N_{up} is the number of undetermined positives, N_{un} is the number of undermined negative test sequences, N_{ng} is the total number of negative test sequences, and N_{po} is the total number of positive test sequences.

Comparison of the studied classifier on the esterase family

	PNN	BLAST	SAM
Precision	98.2%	92.8%	95.2%
Specificity	98.4%	95.6%	99.4%
Sensitivity	98.7%	100.0%	99.6%
Unclassified$_p$	0.0%	0.0%	1.1%
Unclassified$_n$	0.0%	6.7%	6.2%

Comparison of the studied classifier on the lipase family

	PNN	BLAST	SAM
Precision	98.7%	91.0%	95.3%
Specificity	99.3%	95.0%	99.8%
Sensitivity	96.1%	100.0%	100.0%
Unclassified$_p$	0.0%	0.0%	3.1%
Unclassified$_n$	0.0%	6.0%	4.6%

Comparison of the studied classifier on cytochrome family

	PNN	BLAST	SAM
Precision	96.7%	88.5%	99.4%
Specificity	97.2%	92.6%	100.0%
Sensitivity	93.5%	100.0%	100.0%
Unclassified$_p$	0.0%	0.0%	2.2%
Unclassified$_n$	0.0%	6.2%	0.3%

5. CONCLUSION

We have presented a new Probabilistic Neural Network approach for protein classification based on their feature composition. The results have shown that, for problems with characteristic similar to the one addressed in this work, the idea of classifying samples according to the class which is better characterized by one of the subsets of their attributes looks promising.

REFERENCES

[01] A Science Primer: Just the Facts: A Basic Introduction to the Science Underlying NCBI Resources – Molecular Modeling: A Method for Unraveling protein structure and function. http://www.ncbi.nlm.nih.gov/About/primer/molecularmod.html

[02] Lecture Notes in Computer Science, Springer Berlin / Heidelberg 3992 (2006)863-870. www.springerlink.com/content/3l3065848t62jj22/

[03] **T. Hubbard, D. Andrews, M. Caccamo, G. Cameron,** Ensembl 2005, *Nucleic Acids Research* 2005 33(Database Issue):D447-D453

[04] **Q. Ma and J. T. L. Wang**, Biological data mining using Bayesian neural networks: A case study, *International Journal on Artificial Intelligence Tools*, 8(4):433{451, 1999.

[05] **Haussler D,** A brief look at some machine learning problems in genomic. *Proceedings*

of the Tenth Annual Conference on Computational Learning Theory. 1997(109-113).

[06] **Faouzi Mhamdi, Mourad Elloumi, Rakotomalala**, Text mining, feature selection and Data mining for protein classification, *IEEE International Conference on Information & Communication Technologies: From Theory to Applications* (IEEE/ICTTA'04), 2004

[07] M. Dash, H. Liu, **Feature Selection for Classification**, *Intelligent Dana Analysis* (1997)131-156.

[08] Ibrahiem M.M., El Emary, S Ramakrishnan, **On the Application of Various Probabilistic Neural Networks in Solving Different Pattern Classification Problems**, *World Applied Sciences Journal*, 2008 4(6):772-780

[09] **Florin Gorunescu**, Benchmarking Probabilistic Neural Network Algorithms, *Proceedings of International Conference on Artificial Intelligence and Digital Communications, 2006*

[10] **F. Gorunescu, M. Gorunescu, E.El-Darzi, M. Ene, S. Gorunescu**, Statistical Comparison of a Probabilistic Neural network Approcah in Hepatic Cancer diagnosis, *Proceedings Eurocon2005 - IEEE Internation Conference on "Computer as a tool"*, Belgrade, Serbia, November 21-24, 1-4244-0049-X/05/20.00 ©IEEE, 2005, 237-240

[11] **S. Henikoff, J.G. Henikoff**, Amino Acid Substitution Matrices from Protein Blocks, *Proceedings of the National Academy of Sciences 89*, 1992:10915-10919.