Journal of Theoretical Biology 421 (2017) 1–15

Contents lists available at ScienceDirect

Journal of Theoretical Biology

journal homepage: www.elsevier.com/locate/jtbi

Extracting features from protein sequences to improve deep extreme learning machine for protein fold recognition

Wisam Ibrahim, Mohammad Saniee Abadeh*

Faculty of Electrical and Computer Engineering, Tarbiat Modares University, Tehran, Iran

ARTICLE INFO

Article history:
Received 7 January 2017
Revised 5 March 2017
Accepted 24 March 2017
Available online 27 March 2017

Keywords:
Protein fold recognition
Extreme learning machine
Protein descriptor
Feature extraction

ABSTRACT

Protein fold recognition is an important problem in bioinformatics to predict three-dimensional structure of a protein. One of the most challenging tasks in protein fold recognition problem is the extraction of efficient features from the amino-acid sequences to obtain better classifiers. In this paper, we have proposed six descriptors to extract features from protein sequences. These descriptors are applied in the first stage of a three-stage framework PCA-DELM-LDA to extract feature vectors from the amino-acid sequences. Principal Component Analysis PCA has been implemented to reduce the number of extracted features. The extracted feature vectors have been used with original features to improve the performance of the Deep Extreme Learning Machine DELM in the second stage. Four new features have been extracted from the second stage and used in the third stage by Linear Discriminant Analysis LDA to classify the instances into 27 folds. The proposed framework is implemented on the independent and combined feature sets in SCOP datasets. The experimental results show that extracted feature vectors in the first stage could improve the performance of DELM in extracting new useful features in second stage.

1. Introduction

Proteins are the components which play important roles in the organisms' activities. Protein's functions depend on the interactions with other proteins and its folding. Incorrect protein folding usually leads to changing in properties of the protein which causes some diseases (Hashemi et al., 2009).

Each protein macromolecule is built of various units called amino acids which connected together in sequences. The real problem in the genome-sequencing studies is that the known protein sequences are rapidly increasing while the number of the proteins with known tertiary structure is limited (Chmielnicki and Sta, 2012).

Protein folding is the process by which a protein converted from its denatured state to its specific biologically active conformation, and various proteins have significantly various rates of folding (Guo et al., 2011). Protein fold recognition is obtaining the tertiary structure of the proteins from the amino acid sequences without relying on the sequence similarities (Ding and Dubchak, 2001). When the similarity between the input and target sequences is little, this task becomes more challenging. To recognize the fold similarity between the proteins many alignment methods have been employed using sequence information, structural information or both (Cheng and Baldi, 2006).

The traditional methods which have been used in the protein fold recognition such as X-ray crystallography, and Nuclear Magnetic Resonance NMR are expensive and time-consuming. However, the computational methods such as Neural Networks NN and Support Vector Machine SVM have been used for protein fold recognition because they are cheaper and faster than laboratorial methods (Valavanis et al., 2010).

In this paper, a framework of three stages has been proposed to develop a faster and more accurate computational method for protein fold prediction. Six descriptors have been employed in the first stage of the proposed framework to extract feature vectors from the amino acid sequences: Auto covariance (AC), Moran Autocorrelation (MA), Geary Autocorrelation (GA), Moreau Broto Autocorrelation (MBA), Conjoint triad (CT), Local descriptor (LD). The resulting feature vectors are added into the original feature vectors (Ding and Dubchak, 2001) to improve the Deep Extreme Learning Machine DELM performance in the second stage of the proposed framework.

It is important here to point out that many previous researchers have applied many methods to expand the original Ding and Dubchuk dataset. Shen and Chou (2006) have used Chou's pseudo-amino acid composition PseAAC (Chou, 2005) to extract four feature vectors and add them to five feature vectors from Ding and Dubchuk datasets. An ensemble classifier named PFP-Pred has been employed by Shen and Chou (2006) on these nine feature vectors. This classifier uses Evidence-Theoretic K-Nearest Neighbor (ET-KNN) as base classifier. Then Shen and Chou (2009) proposed a

* Corresponding author.
E-mail address: saniee@modares.ac.ir (M.S. Abadeh).

http://dx.doi.org/10.1016/j.jtbi.2017.03.023
0022-5193/

novel approach named PFP-FunDSeqE which is the fusion of functional domain descriptor and Pse-PSSM descriptor. This predictor has been applied on the protein sequences of Ding and Dubchuk dataset. However, other works such as Guo et al. (2011) and Liu et al. (2012) have used Chou's PseAAC to extract feature vectors from protein sequences. Principal component analysis PCA in the first stage of the proposed framework is implemented to reduce the dimensionality of extracted feature vectors. Finally, Linear Discriminant Analysis is implemented in the third stage to classify the proteins of the independent and combined feature sets in 27 folds.

As demonstrated by a series of recent publications (Jia et al., 2015; Liu et al., 2015a; Chen et al., 2016a, Cheng et al., 2016; Jia et al., 2016a,b,c, Liu et al., 2016b, Qiu et al., 2016a,b, Zhang et al., 2016), to establish a really useful sequence-based statistical predictor for a biological system and also to make the presentation logically crystal clear, we need to consider the Chou's 5-step guidelines (Chou, 2011): (i) construct or select a valid benchmark dataset to train and test the predictor; (ii) formulate the biological sequence samples with an effective mathematical expression that can truly reflect their intrinsic correlation with the target to be predicted; (iii) introduce or develop a powerful algorithm (or engine) to operate the prediction; (iv) properly perform an evaluation method to objectively evaluate the anticipated accuracy of the predictor; (v) efforts to establish a user-friendly web-server for the predictor that is accessible to the public. Below, after introducing related works, let us elaborate how to deal with these steps one-by one.

The rest of this paper is organized as follows: In Section 2, the related works about the protein fold recognition are briefly introduced. Section 3 explains the datasets and feature vectors which are used in this paper. Protein descriptors are described in Section 4. In Section 5, the used methods and materials are explained. The proposed framework is described in Section 6. The experimental results are reported in Section 7, and the conclusion is discussed in Section 8.

2. Related works

Many computational methods based on machine learning have been implemented for protein fold recognition (Ding and Dubchak, 2001). Ding and Dubchak (2001) applied unique *one-versus-others* and the *all-versus-all* methods with SVM and multilayer Neural Network on Structural Classification of Protein (SCOP) datasets with 27 folds. Cheng and Baldi (2006) presented an approach of two stages. Pairwise similarity features have been extracted in the first stage, then SVM has been applied in the second stage to predict the folds of protein pairs. A hyper framework is proposed by Abbasi et al. (2013) consists of three main components. These components are Fuzzy Resource-Allocating Network FRAN, RBF networks based on PSO RBF-PSO and KNN which applied on six feature vectors of SCOP. Pal and Chakraborty (2003) proposed five feature vectors based on information of the sequences and the hydrophobicity of amino acids, then used Radial Basis Function RBF networks and Multilayer Perceptron MLP in two levels to classify instances into 27 folds.

However, fusion systems are proposed by other researchers to improve the performance of the classifiers. Combined classifiers can represent all aspects of the problem while single classifier can't do that (Jazebi et al., 2009). A fusion system proposed by Chmielnicki and Sta (2012) to combine SVM as a discriminative classifier and Regularized Discriminant Analysis RDA as a generative classifier. The advantages of both approaches of the classification (generative and discriminative) have been carried out in this fusion system. Hashemi et al. (2009) applied MLP and RBF networks with two ensemble methods. Each classifier has been employed with weighted majority voting method and Bayesian fusion method separately on six feature vectors extracted from protein sequences. The Bayesian ensemble method achieved decreasing of both bias and variance error of the used classifiers. Guo and Gao (2008) proposed a hierarchical fusion system includes two layers and three steps. Firstly, six classifiers GAET-KNN in the first layer define six 27-dimension vectors represent the confidence degree of the related folds. Secondly, classifiers in second layer classify the samples based on the weights in the vectors obtained in the previous layer. Finally, a genetic weighted voting system ensembles outputs from the second layer in a 27-dimension vector as a final classification output.

Structural classes of the proteins (all α, all β, α/β, and $\alpha+\beta$) can be used in hierarchical models to obtain better results in protein fold recognition problem. Huang et al. (2003) proposed a two level framework and employed five independent classifiers of SVM and MLP. In the first level, one classifier is used to classify the protein features into four classes. Then in the second level, four classifiers classify the features resulting from the previous level into 27 folds. Their framework does not contain any voting system. Therefore, bad classification in the first level will not be corrected. However, Aram and Charkari (2015) employed a voting system after the first level to remove the bad results. Three classifiers have been applied in the first level to classify the instances into 4 classes. One feature is extracted as a result form the first layer and added into the basic feature vectors in the next level. Finally, a classifier classifies the instances in 27 folds.

3. Datasets and feature vectors

3.1. Datasets

The main four levels of Structural Classification of Protein (SCOP) datasets are classes, folds, superfamilies and families. The major structural classes which have been widely used for protein structural class prediction by researchers such as Chou (1995), Chou et al. (1998), Li et al. (2009), Sahu and Panda (2010), Kong et al. (2014), Zhang et al. (2014) are all alpha, all beta, alpha+beta and alpha/beta (α, β, $\alpha+\beta$, α/β). However, in this study we have used these four structural classes to extract new features to improve the protein fold recognition as explained in Section 6.

To compare our method with previous works, the dataset that were introduced in Ding and Dubchak (2001) has been used. Due to lack of information on sequence records of two proteins (2SCMC and 2GPS) in the training dataset and two proteins (2YHX_1 and 2YHX_2) in test dataset, we excluded these four proteins from the working dataset. Therefore, the training datasets and test dataset contain 311 and 383 proteins respectively. The sequences in training dataset have similarity less than 35%, while the sequences in test datasets have similarity less than 40%. These datasets contain most populated 27 folds represent all major structural classes (α, β, α/β, $\alpha+\beta$). The proteins and their folds in training and test datasets are described in Table 1.

3.2. Feature vectors

Dubchak et al. (1995) used three descriptors to extract six feature vectors from the protein sequences. The descriptors are Composition, Transition and Distribution. The extracted feature vectors are Amino acids composition (C), Predicted secondary structure (S), Hydrophobicity (H), Polarity (P), Normalized van der Waals volume (V), and Polarizability (Z). Feature vector C consists of 20 features while each vector of the remaining ones consists of 21 features. The independent and combined feature vectors and the number of their features are shown in Table 2.