COM 599100 Deep Learning Final Project Proposal – Protein Family (Group 4)

Po-Yu Chou (team leader), Yi-Yu Zheng, Ya-Ting Yang, Yu-Chia Huang and Yu-Hsiu Huang

Abstract—**In the field of bioinformatics, identifying protein function from amino acid sequence is a fundamental problem. With a thorough understanding of protein structures, the progress of drug design and genetic engineering will be significantly accelerated. Investigating protein functional often involves structural studies (crystallography) or biochemical studies, which require time consuming efforts. In this project, we explore how well we can represent biological function through examination of raw sequence alone. With the emerging study of deep neural networks, various fields have groundbreaking progress by incorporating the novel methods of DNN such as computer vision and natural language processing. Using a large corpus of protein sequences and their annotated protein families, many works have succeed in classifying the structure of protein for several datasets. In this work, we experiment two deep neural network architectures—GRU and 1D-CNN to train classifiers for protein family identification for the Research Collaboratory for Structural Bioinformatics (RCSB) Protein Data Bank (PDB) dataset.**

Index Terms—**classification, deep learning, protein family.**

I. Introduction

With the development of advanced measuring techniques and instruments, we are able to retrieve a myriad of important information about the structure of biological macromolecules using X-ray crystallography, Nuclear Magnetic Resonance (NMR) spectroscopy and cryo-electron microscopy. Accurate identification of protein functions has applications in a wide variety of areas, such as understanding diseases, drug design and genetic engineering for agriculture. Nevertheless, high throughput experiments like the next generation sequencing technologies are resulting in a large number of new protein sequences uncharacterized [1].

Sequenced-based methods for protein fold recognition can be summarized into two categories: the sequence alignment methods and machine learning/deep learning based methods. The former one determines the unknown structure of sequences by calculating the alignment scores between sequences. Despite the success, the sequence alignment methods are essentially an indirect means of nearest neighbor methods, which cannot give an insightful explanation about the sequence-structure relationship. Consequently, we are motivated to propose a deep learning-based end-to-end protein structure classifier. We can expect our model not only have a decent performance in terms of classification accuracy but also obtain meaningful features extracted automatically from the neural networks without the bioinformatics expertise.

II. Material and Method

To create a distributed representation of our protein sequences, we represent each sequence as a series of trigrams (a block of 3 amino acids) and create a distributed representation of each trigram using Global Vectors for Word Representation (GloVe). To train models for protein family classification, we limited ourselves to sequences of less than 1000 overlapping trigrams. The data was then split into training/validation/test folds at a 70/15/15 ratio preserving class stratification.

A. GRU Approach

From the reference paper [2], Protein Family Classification with Neural Networks, Gated Recurrent Neural Networks perform pretty well on the protein family classification, and thus we would apply this method to complete our task and compare it with other methods.

Gated Recurrent Neural Networks extend recurrent neural networks (RNNs) by using gated recurrent units (GRUs). GRUs consist of two additional gates, an update gate and a reset gate. The reset gate determines how much of the previous hidden state is used before the nonlinear activation. The update gate determines how much of the new memory content is used with the previous hidden state to determine the new hidden state. Together, these two gates allow long or short term dependencies to be expressed.

We used overlapping trigrams in sequence as the inputs to the neural networks and initialized our inputs with GloVe embeddings and allowed them to be trained.

When training the model, we would apply maxpooling over all hidden layers' outputs of the forward net and backward net, respectively, stack the maxpooled outputs before feeding into the softmax layer, and experiment with a number of learning rates and method of dropout.

B. 1D-CNN Approach

The second method directly classifies protein sequences into folds by 1D-convolutional neural network (DeepSF) [3]. According to Hou et al. 2018, this neural network contains 15 layers in total, including the input layer, 10 convolutional layers, one 30-max pooling layer, one flattering layer, one

Po-Yu Chou (105061110) [†] Yi-Yu Zheng (105061108) [†]
Ya-Ting Yang (105061210) [†] Yu-Chia Huang (105061236) [†]
Yu-Hsiu Huang (104061249) [†]
[†]Department of Electrical Engineering, National Tsing Hua University

fully-connected hidden layer, and the output layer. The whole architecture is illustrated in Figure 1. Between the input layer and the first convolutional layer, the ReLU activation function is applied as non-linear transformation. Each convolutional layer has 10 filters and the size of each filter is chosen to be 10 and 6. The fully-connected hidden layer contains 500 nodes and the dropout techniques is applied to prevent over-fitting. As for the output layer, the softmax activation function is used in the output node.

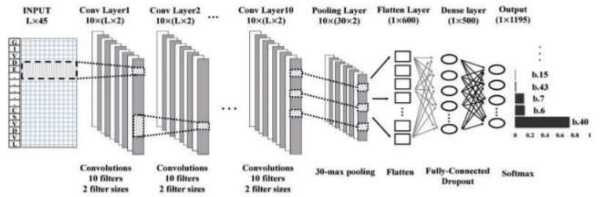

Figure 1. The architecture of DeepSF for fold classification. The L denotes the variable length.

The overall procedure remains the same as Hou et al. 2018 while in our case we do not classify the protein sequence into folds. Instead, we classify them into the sub-structure called families. In this case, the number of output layer should increase since the number of family is higher than the number of folds.

III. Evaluation

For evaluation of our model, we calculate several performance measures, such as: precision, recall, accuracy, and F1 score defined as:

$$precision = \frac{tp}{tp + fp}$$

$$recall = \frac{tp}{tp + fn}$$

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

$$F1\ score = 2 \times \frac{precision \times recall}{precision + recall}$$

Where tp are the number of true positives, fp are the number of false positives, tn are the number of true negatives and fn are the number of false negatives

According to “Protein Family Collection with Neural Network, the F1-score of this project can reach 0.948452 (using GRU); according to “DeepSF: deep convolutional neural network for mapping protein sequences to folds”, for some specific folds, the accuracy can reach 97.5%. However the aforementioned results are evaluated on different datasets, so the actual performance of our work could potentially be better after delicate fine tuning with respect to the PDB dataset.

IV. Discussion

The first difficulty we may encounter is sequence encoding (choosing between different embedding methods) because different numeric representation of the amino acid sequence may influence our model performance. The second difficulty is that there may be a lot of information and classes of the protein families, so we need to decide which to keep and which to delete. Last but not least, no matter which model we use (GRU, 1dCNN,……), we do require a lot of computation resources to finish the training part.

References

[1] Nauman, M., Rehman, H. U., Politano, G., & Benso, A. (2018). Beyond Homology Transfer: Deep Learning for Automated Annotation of Proteins. Journal of Grid Computing. doi:10.1007/s10723-018-9450-6

[2] Timothy K. Lee, Tuan Nguyen, "Protein Family Classification with Neural Networks", 2016.

[3] Hou J, Adhikari B, Cheng J. DeepSF: deep convolutional neural network for mapping protein sequences to folds. Bioinformatics. 2017;34(8):1295-1303.