ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20200417091724
Original scientific paper

Distributed Representation of Protein Sequence Based on Multi-Alignment Results

Siqi WANG, Liu HE, Shi CHENG, Xiaohu SHI*

Abstract: Protein sequence representation is a key problem for protein studies, especially for those sequence-based models. In this paper, a distributed representation model of protein sequence is proposed, which involves evolutionary information by introducing multi-alignment results. Firstly, we construct a non-redundancy protein dataset and perform multi-alignment for each protein. Then k-mer amino acids "biology corpus" was abstracted from the alignment results which are "evolutionary information" enriched. Using the "biology corpus", k-mer amino acids distributed embedding vectors could be trained according to word2vec method. We compared the amino acid pair distance derived from our produced 1-mer amino acids distributed embedding vectors with that derived from BLOSUM62; it was found that their Pearson coefficient is 0.937, showing they have strong correlation. Then we applied the obtained amino acids distributed embedding representation to protein secondary structure recognition and solubility prediction. For both of the experiments, our proposed alignment results based amino acid distributed representation outperforms that derived directly from protein sequences. Moreover, compared to those existing up-to-date algorithms, our method could get better or comparative results, on condition of only using the feature of our produced amino acid distributed vectors.

Keywords: distributed representation; embedding; protein sequence; word2vec

1 INTRODUCTION

Proteins are polypeptide chains composed of amino acid sequences, which are one of the most important components of living organisms. Proteins are involved in almost all the processes within organisms, including catalysing metabolic reactions, DNA replication, responding to stimuli, providing structure to cells and organisms, and transporting molecules from one location to another [1]. Proteins function biologically depending on their tertiary structures, which are generally believed as being determined by their primary structures, i.e., amino acid sequences [2]. However, experiment techniques for protein structure determination, such as X-ray crystal diffraction and NMR, are extremely expensive and time consuming, resulting in the knowledge gap between obtained protein structures and protein sequence. Therefore, sequence analysis of proteins is of great significance.

The first problem of sequence analysis is how to represent the sequence. Traditional protein sequence representation methods mainly include orthogonal coding and profile method. Orthogonal coding, also known as one hot representation method, represents each amino acid by using a 21-bit binary vector, of which only one bit is '1', and its position corresponds to the amino acid type. It is natural and simple, but could not grasp the relationships between different amino acids. Position-Specific Scoring Matrix (PSSM) is a convinced and popular used profile method which contains the evolutionary information. However, it is computationally expensive because multi-alignment is needed for each protein sequence. With the development of deep learning, distributed representation method has made great achievements in the Natural Language Processing field, such as word2vec [3], and GloVe [4], which has received great attention and been widely used [5, 6]. Naturally, the protein sequence was considered as biology "sentence" and "amino acid vectors" were obtained by word2vec, which could represent the protein sequence for further protein studies [7, 8]. However, the evolutionary information was not included in the existing methods, though it is very important for protein analysis. Aiming at the abovementioned problems, this paper developed an amino acid distributed representation model, which involves evolutionary information by introducing multi-alignment results.

In order to test the effectiveness of our proposed method, protein secondary structure recognition and solubility prediction were applied. Numerical results show that our method outperforms the existing representation methods.

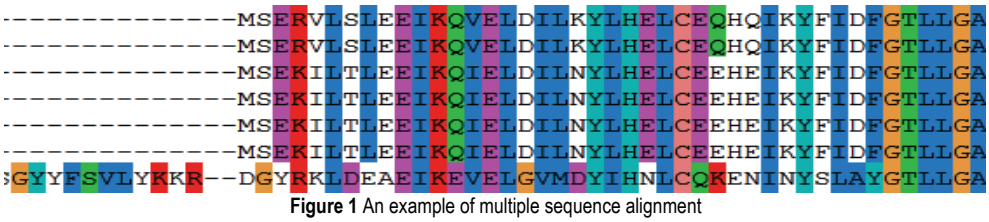


Figure 1 An example of multiple sequence alignment

2 METHODS

Focused on protein sequence analysis, we proposed a new embedding representation method based on the distributed embedding method word2vec, which introduces the evolutionary information by using the multi-alignment results to construct the "biology corpus". In this section, data processing and the method will be described in detail.

2.1 Data Retrieval and Pre-Processing

To construct the protein dataset, we downloaded 160,000 non-redundant protein data from NCBI's FTP site. After that, multi-alignment of each protein sequence was

performed using the package of BLASTp [9], which could be accessed at https://blast.ncbi.nlm.nih.gov/Blast.cgi. We set the parameters of the BLASTp to find only the 20 most similar protein sequences for each protein sequence as output. Then, for each sequence in the dataset, a multi-alignment profile was obtained (an example is shown in Fig. 1). For the representative reason, those profiles with less than 10 sequences are discarded, and finally, 40968 multiple-sequence alignment profiles remained.

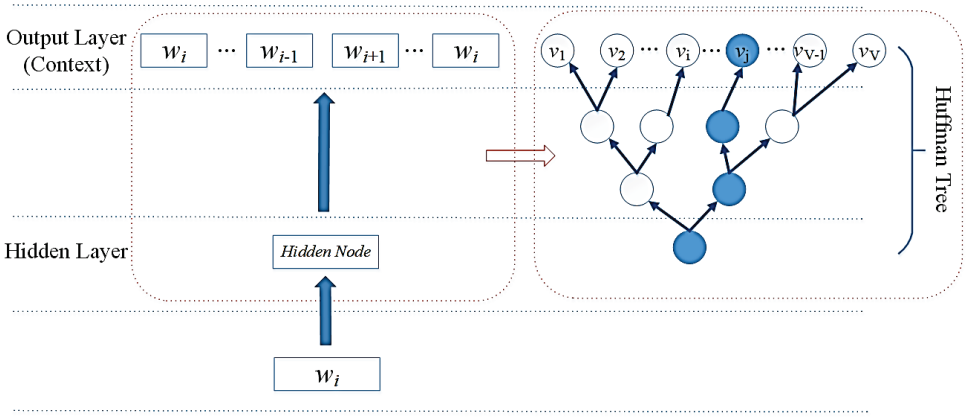
2.2 Word2vec Distributed Representation Method

2.2.1 Word2vec Introduction

The term of "word embedding" was originally developed by Bengio et al. in 2003 [10]. Different from former vector space models, word embedding is trained in a neural language model together with the model's parameters. However, it was not until word2vec was proposed by Google in 2013 that the word embedding model has attracted numerous researchers' attention and thus got developed promptly. For example, a year later, Stanford developed GloVe [4], Facebook proposed FastText in 2016 [11] and Peters et. al. introduced ELMo in 2018 [12]. These models have been widely used in various natural language processing problems and have become standard word representation methods. In this paper, we chose word2vec as the amino acid embedding model, so it will be described in detail below in this section.

Word2vec is a shallow two-layered neural network which produces word embedding for better word representation. It is based on the hypothesis of "a word is its own context", meaning that a word could be described by its own context, or vice visa, which derived two types of network frameworks: CBOW and Skip gram. In CBOW, the inputs are the context word embedding vectors and the output is the central word, while it is the contrary in Skip gram. Taking Skip gram as an example, suppose we have a sentence of $\{w_1, w_2, \ldots, w_{i-1}, w_i, w_{i+1}, \ldots, w_n\}$, where w_i means the i-th word in the sentence. Then for each word, saying about w_i, a sample could be abstracted by setting it as the central word, and the neighbors in t-window as its context, namely that $\{w_i, w_{i-t}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+t}\}$. Skip gram model takes input as w_i, and output as $\{w_{i-t}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+t}\}$ (See Fig. 2a).

The above described samples consisting of a pair of a central word and its context words could be considered as "positive samples", and the pairs 5 that do not exactly match all are "negative samples". Obviously, the numbers of "positive samples" and "negative samples" are severely unbalanced. Word2vec uses two kinds of sampling methods to solve this problem, namely hierarchical softmax and negative sampling [13, 14]. In negative sampling method, only a smaller number of "negative" samples are randomly selected to train the model instead of using the whole "negative" sample set. In hierarchical softmax method, the output layer uses a Huffman tree to represent the whole vocabulary, with each word corresponding to a leaf node of the tree, and the path linking the leaf node and root representing the word (See Fig. 2b). In this paper, Skip gram framework and hierarchical softmax method is used, the detail information about word2vec could refer to [3].



(a) Skip-gram framework (b) Hierarchical softmax structure

Figure 2 Skip-gram model with hierarchical softmax structure

2.2.2 K-mer Amino Acids Content Corpus Construction

Unlike the existing method that directly uses the protein amino acid sequences as the "biology sentence" [15], we construct the "biology corpus" from the multi-alignment results, which includes rich evolutionary information. We just consider the normal 20 types of amino acids in this paper. As the alignment results include the symbol of "indel", the generalized amino acids have 21 types. Then for k-mer amino acids, the size of vocabulary should be 21^k. Tab. 1 is an example of multi-alignment results. The target protein sequence is $A_1^{(0)}, A_2^{(0)}, A_3^{(0)}, A_4^{(0)}$, the first obtained similar protein sequence is $A_1^{(1)}$, $A_2^{(1)}$, $A_3^{(1)}$, $A_4^{(1)}$, and so on.

Table 1 Sequence alignment grouping

Target sequence	$A_1^{(0)}$	$A_2^{(0)}$	$A_3^{(0)}$	$A_4^{(0)}$	…
Similar sequence 1	$A_1^{(1)}$	$A_2^{(1)}$	$A_3^{(1)}$	$A_4^{(1)}$	…
Similar sequence 2	$A_1^{(2)}$	$A_2^{(2)}$	$A_3^{(2)}$	$A_4^{(2)}$	…
Similar sequence 3	$A_1^{(3)}$	$A_2^{(3)}$	$A_3^{(3)}$	$A_4^{(3)}$	…
⋮	⋮	⋮	⋮	⋮	…
Similar sequence n	$A_1^{(n)}$	$A_2^{(n)}$	$A_3^{(n)}$	$A_4^{(n)}$	…

In the alignments, the rich evolutionary information is included in the columns of the table. Naturally, we use the column of alignments to construct "biology sentences". However, the aligned sequences are highly homologous and many columns are highly identical. Therefore, we should filter the alignment results by removing those

columns with too high "information entropy". For column i, the "information entropy" is defined by:

$$H_i = -\sum_{j=1}^{21} p_j^{(i)} \ln p_j^{(i)} \quad (1)$$

where $p_j^{(i)}$ means the j-th type of amino acids frequency in column i.

As we want to get the embedding of the k-mer amino acid vocabulary, the "biology sentences" should consist of k-mer amino acid. So we define the "information entropy" of a k consecutive columns as:

$$H^{(k)} = \prod_{i=1}^{k} H_i \quad (2)$$

where H_i is "information entropy" of the ith column within the k consecutive columns. If the "information entropy" of the k consecutive columns is less than the threshold, some samples can be generated as follows: randomly select a k-mer amino acid as the target "word" from all the n similar sequences of this k column, and randomly select other $2tk$-mers as its "context", this process could be repeated several times. Finally, a "biology corpus" for k-mer amino acid can be derived from multi-alignment profiles.

3 FRAMEWORK OF ALIGNMENT BASED PROTEIN SEQUENCE DISTRIBUTED REPRESENTATION

The framework of the proposed method is shown as Fig. 3. We first download the protein data from NCBI, then use the blast program to analyze each protein's similarities within the download data. By removing those sequences with high redundancy, we get the non-redundant data set of protein sequence. Next, we perform multi-alignment for each protein sequence, and construct the "biology corpus" of k-mer amino acids from the alignment results. Then we use the word2vec method to train k-mer amino acids distributed embedding vectors. In the paper, CBOW framework and hierarchical softmax method are used in word2vec.

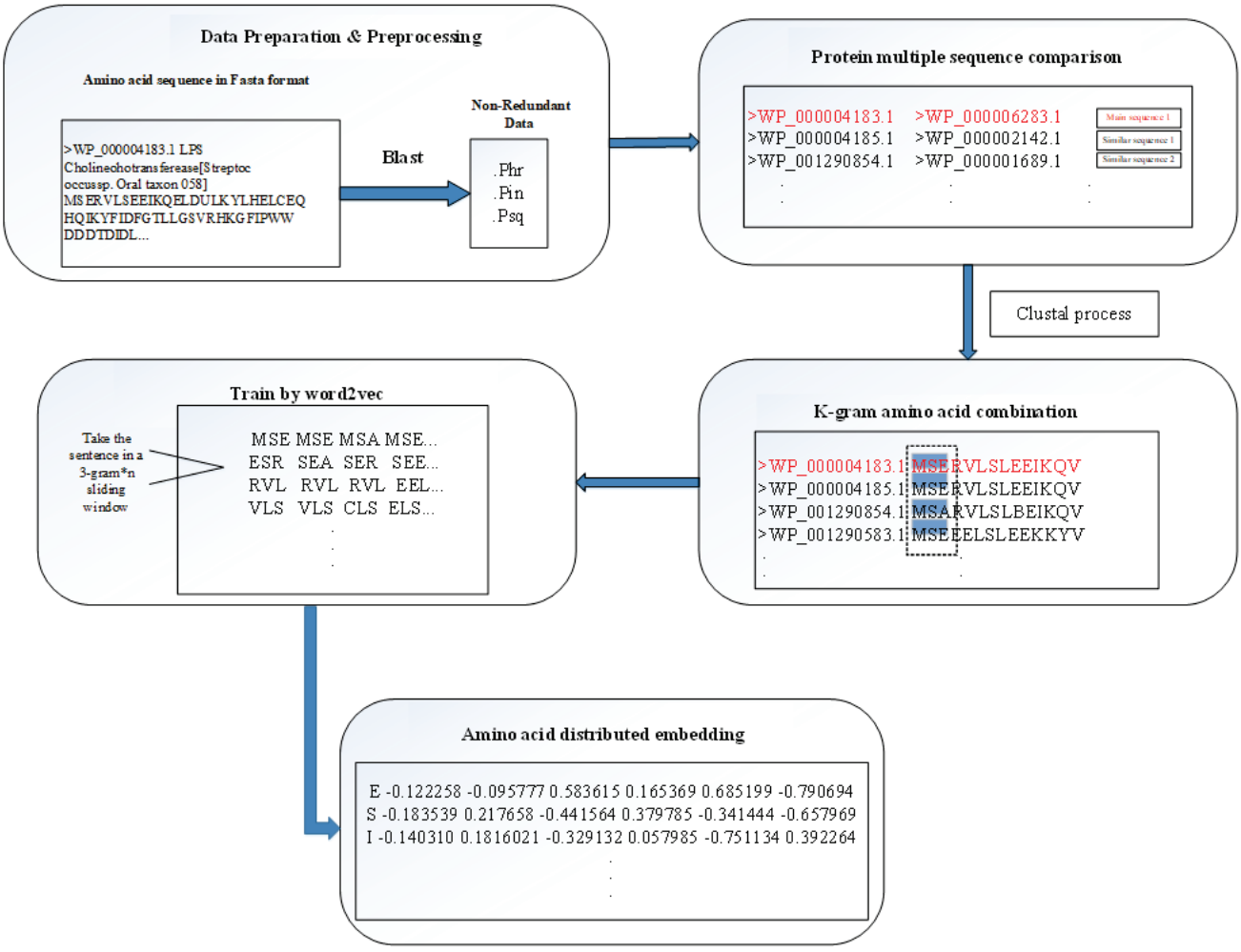


Figure 3 The framework of the proposed method

4 RESULTS

Firstly, we compare our obtained 1-mer amino acids embedding vectors with BLOSUM62 to testify the rationality of the proposed embedding method. Then two designed experiments are executed to verify the performance of our method, namely, the secondary structure recognition and the solubility prediction from protein sequence. Some existing popular used methods are selected for comparison.

4.1 Comparison of 1-mer Amino Acids Distributed Representation with BLOSUM62

A common method to measure the similarity between different types of amino acids is to use the so called "substitution matrix", of which PAM and BLOSUM are the most popular [16]. BLOSUM was originally introduced by Steven Henikoff et al. in 1992 [17], the main idea of which is to compare the likelihood of amino acid substitutions in homologous protein sequences compared with the substitutions in the background. For example, BLOSUM62 is a score matrix by constructing homologous protein sequences alignment dataset with similarities greater than 62%.

To verify the rationality of our proposed amino acids embedding method, we compared the result of our 1-mer amino acids distributed embedding representation with BLOSUM62. We calculate the similarity of each pair of amino acids by using dot product of their 1-mer amino acids embedding vectors, and the resulting similarity matrix has the same shape as BLOSUM62 (See Fig. 4). Both of them are stretched to a vector with dimension210. Fig. 5 shows the comparison result of these two vectors after normalization. Their Pearson coefficient is 0.937, and it means that they have strong correlation, showing that our proposed amino acid distributed embedding vector model is convincing.

A	1.0																			
R	−0.02	1.0																		
N	−0.25	0.203	1.0																	
D	−0.17	0.079	0.752	1.0																
C	0.646	0.060	0.238	−0.15	1.0															
Q	0.306	0.586	0.110	0.313	−0.02	1.0														
E	0.121	0.174	0.271	0.725	−0.22	0.738	1.0													
G	0.390	0.234	0.479	0.306	0.462	0.080	−0.05	1.0												
H	0.154	0.467	0.599	0.542	0.416	0.488	0.455	0.193	1.0											
I	0.28	−0.25	0.003	−0.34	0.210	−0.22	−0.25	−0.61	0.049	1.0										
L	−0.15	−0.02	0.072	−0.39	0.370	−0.08	−0.37	−0.35	0.250	0.868	1.0									
K	-0.37	0.864	0.375	0.345	−0.29	0.592	0.390	0.023	0.405	−0.16	−0.07	1.0								
M	0.132	0.014	−0.09	−0.33	0.220	0.397	−0.02	−0.16	0.094	0.525	0.708	−0.04	1.0							
F	0.151	−0.27	0.081	−0.07	0.298	0.000	-0.10	−0.24	0.410	0.476	0.671	−0.29	0.462	1.0						
P	0.226	−0.04	0.103	0.121	0.035	0.116	0.153	−0.07	0.125	−0.04	−0.07	0.079	−0.19	0.399	1.0					
S	0.565	−0.01	0.430	0.342	0.576	0.314	0.380	0.547	0.213	−0.12	−0.19	−0.05	0.062	−0.14	0.207	1.0				
T	0.144	−0.03	0.373	0.126	0.376	0.256	0.275	0.056	0.076	0.422	0.231	0.083	0.364	−0.05	0.154	0.772	1.0			
W	0.221	0.446	0.312	−0.15	0.607	0.098	−0.35	0.322	0.563	0.152	0.547	0.192	0.268	0.557	0.270	0.007	−0.05	1.0		
Y	0.318	−0.01	0.101	0.106	0.257	0.379	0.164	0.036	571	0.097	0.438	−0.11	0.544	0.828	0.150	−0.07	−0.19	0.529	1.0	
V	0.117	−0.21	−0.19	−0.52	0.448	−0.18	−0.29	−0.52	0.036	0.878	0.769	−0.28	0.444	0.514	0.211	0.042	0.438	0.282	0.109	1.0
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V

Figure 4 Similarity matrix according to 1-mer distributed amino acid vectors

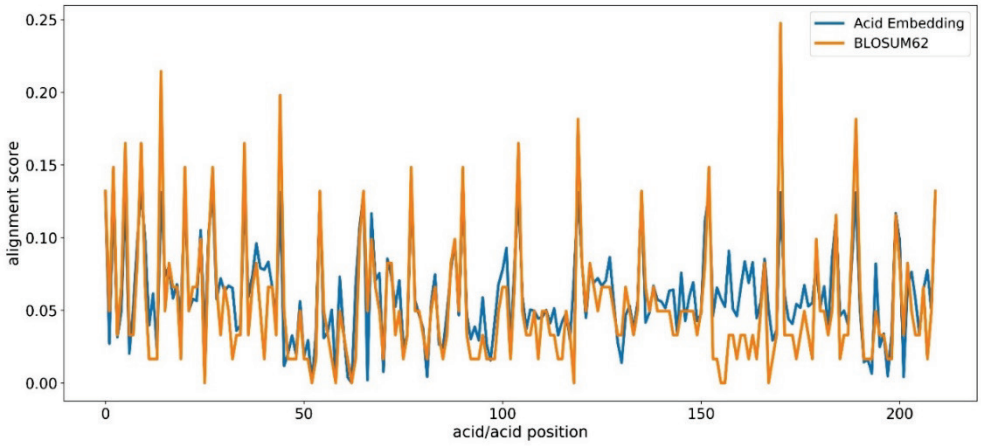


Figure 5 Comparison between the amino acid pair distance from BLOSUM62 and 1-mer distributed amino acid vectors

4.2 Performance on Protein Secondary Structure Recognition

4.2.1 Model Introduction

Long Short-Term Memory (LSTM) is a very successful algorithm in the field of natural language processing, which can effectively extract remote-related information. Naturally, LSTM is extended to amino acid sequence analysis to take full advantage of the correlation features between remote amino acids [18, 19]. Therefore, a variant of LSTM, Bi-directional LSTM, is used as the recognition model for protein secondary structure, as well as the protein solubility prediction model in the next subsection. In the model, it takes our produced distributed amino acid vectors as input, and outputs the types of protein secondary structure or protein solubility accordingly.

The framework of the model is shown in Fig. 6. The input of the model is the k-mer amino acid sequence of the protein (to clarify, 1-mer is shown in Fig. 6). By retrieving the pretrained k-mer amino acid distributed embedding vectors, the *k*-mers of the inputs could be represented by a low dimension condense embedding vector, and then be input into Bi-directional LSTM model. The Bi-directional

LSTM model links to a fully connected layer and then the final softmax layer which outputs the protein secondary structure type. ReLU (Rectified Linear Unit) is used as the active function of the hidden layers in the model.

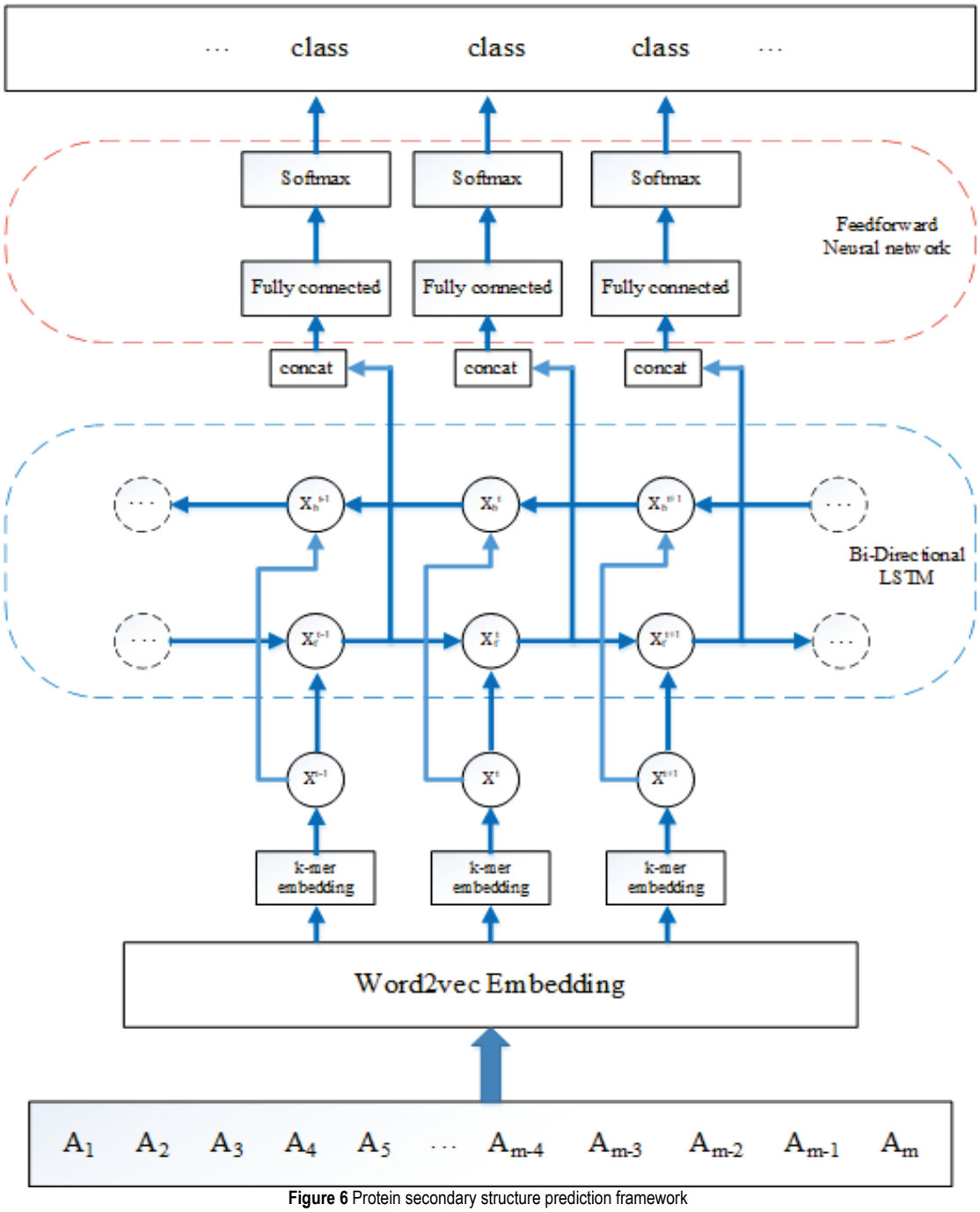


Figure 6 Protein secondary structure prediction framework

4.2.2 Experiment Setup

We select the Cull PDB data set used in the literature [20] as the training and validation data, and CB513 data set as the test data. Cull PDB contains 6128 non-homologous protein sequences after filtering. This data set was generated by the PISCES Cull PDB server, which is typically used in protein structural prediction. To filter the raw data, we set the constraint condition as the protein resolution below 2.5 A, the sequence identity less than 30%, and the protein sequence length is between 50 and 700. Moreover, to avoid the bias of the training data, the sequences in Cull PDB with more than 25% identity of those in CB513 were deleted. Finally, 5.534 sequences of Cull PDB remained, 5278 of which are set as training data and the other 256 ones are set as the validation data.

The secondary structure is classified into 8 types, which could be computed from the PDB files by DSSP program [21]. The input length is set as 700, the max sequence length, those proteins with length shorter than 700 will be padded with zeros to fulfil the input length.

In experiment, our produced 1-mer and 3-mer distributed vectors are respectively set as the input of Bi-directional LSTM model. For comparison, the distributed embedding vectors directly derived from protein sequence are also set as the input of the same model. Moreover, the existing popular algorithms are also applied on this data set, including SSpro8, RaptorX-SS8, SC-GSN, and LSTM

large. RaptorX-SS8 uses a conditional neural field model [22], the SC-GSN uses a convolutional random network [23], and the LSTM large uses a Bi-directional LSTM structure [24]. All of these methods use many other features, such as position-specific scoring matrix (PSSM).

4.2.3 Comparison Results

Comparison results of protein secondary structure recognition on CB513 dataset are listed in Tab. 2. It can be seen that our proposed methods are superior to the other five algorithms, especially the 1-mer based method. Compared with the result of the distributed embedding vectors directly derived from protein sequence (Amino acid embedded based on sequential sequence in table), 1-mer and 3-mer based methods respectively improve by 8.35% and 5.98%, which show that our proposed multi-alignment results based distributed embedding method is more convincing than the one derived from protein sequence. Compared with other 4 existing methods, 1-mer based method is better than all of them, while 3-mer based method is a little lower than that of LSTM large. Noting that our methods just use only one type of features, while the other existing methods utilize many other types of features (for example, RaptorX-SS8 uses PSSM feature combined with the physicochemical properties of proteins), our produced distributed vectors are pretty good. Compared with our two produced vectors, it could be found that 1-mer based method is superior to 3-mer based. As there are so many "indels" existing in the alignment results, the frequencies of some types of 3-mer amino acids are very low, and some types even did not appear. In our experiment, a total amount of 7.785.773 3-mer amino acids were produced, while there are 985 types that were not covered. It is possibly the reason why 1-mer based method beats 3-mer based one.

Table 2 Comparison results for protein secondary structure recognition

Algorithm Name	Accuracy (%)
SSpro8	63.4
RaptorX-SS8	64.9
SC-GSN	66.4
LSTM large	67.4
Amino acid embedded based on sequential sequence	63.5
1-mer amino acid based on Multi-Alignment	**68.8**
3-mer amino acids based on Multi-Alignment	**67.3**

4.3 Performance on Protein Solubility Prediction

4.3.1 Model Introduction

In this experiment, the model is almost the same as that in the above experiment, the only difference is that the output soft max layer has 2 nodes indicating soluble or insoluble classes, while that in the above experiment has 8 nodes indicating 8 secondary structure types.

4.3.2 Experiment Setup

The data set used in this experiment is derived from the SOLP which is used in [25]. It contains a total of 8704 soluble proteins and 8704 insoluble proteins.

Similar to above, our produced 1-mer and 3-mer distributed vectors are respectively set as the input of Bi-directional LSTM model, and the distributed embedding vectors directly derived from protein sequence are also set as the input of the same model for comparison. Moreover, two existing up-to-date methods, namely PROSO [26] and SOL-Pro [25] are selected for further comparison. PROSO is a two-layered structure of logistic regression classifiers, while SOL-Pro is a two-stage support vector machine (SVM) architecture.

4.3.3 Comparison Results of Protein Solubility Prediction

Comparison results of protein solubility prediction are listed in Tab. 3. Compared with the result of the distributed embedding vectors directly derived from protein sequence, 1-mer and 3-mer based methods respectively improve by 6.08% and 3.76%, showing that our proposed multi-alignment results based distributed embedding method is also superior to the one derived from protein sequence. Compared with the other 2 existing methods, our methods are far better than that of PROSO, while they are a little less good than SOL-Pro method. However, considering the large number of sequence-based calculation results and prediction results used in the SOL-Pro, our method could obtain similar results on condition that it only uses one kind of features. This implies the effectiveness of our proposed amino acid embedding method. The comparison between 1-mer based method and 3-mer based one is similar to the former experiments.

Table 3 Comparison results for protein solubility prediction

Algorithm Name	Accuracy (%)
PROSO	59.28
SOL-Pro	74.2
Amino acid embedded based on sequential sequence	69.1
1-mer amino acid based on Multi-Alignment	**73.3**
3-mer amino acid based on Multi-Alignment	**71.7**

5 CONCLUSION

In this paper, we proposed a *k*-mer amino acid sequence distributed representation model based on multiple sequence alignment results. Constructing the "biology corpus" from the alignment profiles, we trained the distributed represented vectors based on word2vec method, thus integrating the evolutionary information into the embedding method. Experimental results show that our proposed model is much better than that directly derived from protein sequence. Only using this one type of features, we could obtain similar or even better results than those of existing popular methods on both protein secondary structure recognition and protein solubility prediction experiments. It implies our proposed model is convincing and effective.

Acknowledgements

The authors are grateful to the support of the National Natural Science Foundation of China (61972174).

5 REFERENCES

[1] https://en.wikipedia.org/wiki/Protein

[2] Serafin, F. (1982). Theoretical Prediction of Protein Antigenic Determinants from Amino Acid Sequences. *Canadian Journal of Chemistry*, *60*(20), 2606-2610. https://doi.org/10.1139/v82-374

[3] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Estimation of Word Representations in Vector Space. *International Conference on Learning Representations*. https://arxiv.org/abs/1301.3781
[4] Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. *In Proceedings of EMNLP*, 1532-1543. https://doi.org/10.3115/v1/D14-1162
[5] Mikolov, T., Sutskever, I., Chen, K.,vCorrado, G., & Dean,vJ. (2013). Distributed Representations of Words and Phrases and their Compositionality. *Advances in Neural Information Processing Systems*, *26*, 3111-3119.
[6] Ou, X. H., Cao, Y., & Mu, X. W. (2015). Classification of Sentiment Sentences Based on Naive Bayesian Classifier. *Journal of Logistics, Informatics and Service Science*, *2*(1), 48-57.
[7] Wu, C., Gao, R., Zhang, Y., & De, M. Y. (2019). PTPD: Predicting therapeutic peptides by deep learning and word2vec. *BMC Bioinformatics*, *20*(1). https://doi.org/10.1186/s12859-019-3006-z
[8] Sun, T., Lai, L., & Pei, J. (2018). Analysis of protein features and machine learning algorithms for prediction of druggable proteins. *Quantitative Biology*, *6*(4), 334-343. https://doi.org/10.1007/s40484-018-0157-2
[9] Altschul, S. F. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res,Sep 1*, *25*(17), 3389-3402. https://doi.org/10.1093/nar/25.17.3389
[10] Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C.(2003). A Neural Probabilistic Language Model. *Journal of Machine Learning Research*, *3*(6), 1137-1155.
[11] Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of Tricks for Efficient Text Classification. *CoRR 2016*, abs/1607.01759.
[12] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *CoRR 2018*, abs/1802.05365. https://doi.org/10.18653/v1/N18-1202
[13] Goldberg, Y. & Levy, O. (2014). Word2vec Explained: Deriving Mikolov et al's Negative-Sampling Word-Embedding Method. *CoRR 2014*, abs/1402.3722.
[14] Mnih, A. & Hinton, G. A. (2008). Scalable Hierarchical Distributed Language Model. *International Conference on Neural Information*, 1081-1088.
[15] Heffernan, R., Paliwal, K., Lyons, J., Singh, J., Yang, Y. D., & Zhou, Y. Q. (2018). Single-Sequence-Based Prediction of Protein Secondary Structures and Solvent Accessibility by Deep Whole Sequence Learning. *Journal of Computational Chemistry*, *39*(26), 2210 -2216. https://doi.org/10.1002/jcc.25534
[16] Styczynski, M. P., Kyle, J. L., Isidore, R., & Stephanopoulos, G.(2008). BLOSUM62 Miscalculations Improve Search Performance. *Nature Biotechnology*, *26*(3), 274-275. https://doi.org/10.1038/nbt0308-274
[17] Henikoff, S. & Henikoff, J. G. (1992). Amino Acid Substitution Matrices from Protein Blocks. *Proceedings of the National Academy of Science of the United States of America*, *89*(22), 10915-10919. https://doi.org/10.1073/pnas.89.22.10915
[18] Singh, S. M. & Panda, A. K. (2005). Solubilization and Refolding of Bacterial Inclusion Body Proteins. *Journal of Fermentation and Bioengineering*, *99*(4), 303-310. https://doi.org/10.1263/jbb.99.303
[19] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. Computer Science. *Proceedings of the 27th International Conference on Neural Information Processing Systems*, *2*, 3104-3112.
[20] Pollastri, G., Przybylski, D., Rost, B., & Baldi, P. (2002). Improving the Prediction of Protein Secondary Structure in Three and Eight Classes Using Recurrent Neural Networks and Profiles. *Proteins*, *47*(2), 228-235. https://doi.org/10.1002/prot.10082
[21] Touw, W. G., Baakman, C., Black, J., Beek, T. A. H., Krieger, E., Joosten, R. P., & Vriend, G. (2015). A series of PDB-related databanks for everyday needs. *Nucleic Acids Research 43*: D364-D368. https://doi.org/10.1093/nar/gku1028
[22] Wang, Z., Zhao, F., Peng, J., & Xu, J. B. (2011). Protein 8-class Secondary Structure Prediction Using Conditional Neural Fields. *Proteomics*, *11*(19), 3786-3792. https://doi.org/10.1002/pmic.201100196
[23] Zhou, J. & Troyanskaya, O. G. (2014). Deep Supervised and Convolutional Generative Stochastic Network for Protein Secondary Structure Prediction. Quantitative Biology. *Proceedings of 31*st *International Conference of Machine Learning*, *32*, 1745-1753.
[24] Sønderby, K. S. & Winther, O. (2015). Protein Secondary Structure Prediction with Long Short Term Memory Networks. https://arxiv.org/abs/1412.7828
[25] Christophe, M. N., Arlo, R., & Pierre, B. (2009). SOLpro: Accurate Sequence-Based Prediction of Protein Solubility. *Bioinformatics*, *25*(17), 2200-2207. https://doi.org/10.1093/bioinformatics/btp386
[26] Pawel, S., Martin-Galiano, A. J., Aleksandra, M., Girschick, T., Holak, T. A., & Frishman, D. (2006). Protein solubility: sequence based prediction and experimental verification. *Bioinformatics*, *23*(19), 2536-2542. https://doi.org/10.1093/bioinformatics/btl623

Contact information:

Siqi WAMG, B.S,
The key laboratory of symbolic computation and knowledge engineering of ministry of education, Jilin University, Changchun, China 130012
College of Computer Science and Technology, Jilin University,
City Changchun, China 130012
E-mail: siqiw17@mails.jlu.edu.cn

Liu HE, M.E,
Intelligent Connected Vehicle Development Institute of China FAW Group Corporation, Changchun, China 130000
E-mail: h_l_heliu@163.com

Shi CHENG, B.S,
The key laboratory of symbolic computation and knowledge engineering of ministry of education, Jilin University, Changchun, China 130012
College of Computer Science and Technology, Jilin University,
City Changchun, China 130012
E-mail: chengshi.jl@foxmail.com

Xiaohu SHI, PhD,
(Corresponding author)
The key laboratory of symbolic computation and knowledge engineering of ministry of education, Jilin University, Changchun, China 130012
College of Computer Science and Technology, Jilin University,
City Changchun, China 130012
Zhuhai Laboratory of Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Zhuhai College of Jilin University, Zhuhai, China 519041
E-mail: shixh@jlu.edu.cn