Journal of Information Systems Engineering and Management
2025, 10(9s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A Novel Framework for Protein Sequence Classification using LSTM and CNN

Prativesh Pawar*[1], Dr. Pinaki Ghosh[2]
[1]PhD scholar
SAGE University Bhopal, India
prativesh.sait@gmail.com
[2]Dept. Of CSE
SAGE university Bhopal, India
pinaki.g@sageuniversity.edu.in

ARTICLE INFO

Received: 22 Oct 2024

Revised: 18 Dec 2024

Accepted: 30 Dec 2024

ABSTRACT

Protein sequence classification has retained its status as one of the biggest challenges in the pharmaceutical industry due to the involved complexities of the subject matter. Technological developments have justified that there needs to be (some) knowledge of identifying and classifying the protein families. On the other hand, and rather cursorily, current methodologies consider a meagre set of protein sequence descriptors. The authors set forth here the amalgamation of BLSTM and BTCN wherein the deeper features within the protein sequence shall be explored. The thesis seeks to divide protein sequences by the sliding duration of the sliding window applied. Local dependencies within these segments may be explored with the proposed convolutional network, capturing interactions between global residues involving BLSTM network. Hence, a BLSTM has been taken as needed in the whole creation, because there is a dependency between amino acid classification and past and future secondary features, and method is that it achieves bidirectional properties avoiding any knowledge of the past and future information for gaining a necessary amount of additional insight. The proposed ensemble model has been concluded as more suitable for protein structure prediction research

Keywords: Protein sequence; Amino acid; Bidirectional long short-term memory (BLSTM); bidirectional temporal Convolutional network (BTCN)

I.INTRODUCTION

All biological tissues and cells are composed of the basic structural molecule known as protein. And it's the primary means via which all of life's pursuits are conveyed. In the fields of biology, medicine, and pharmacy, an understanding of protein function is fundamental. For example, learning how a protein works can direct genetic engineering efforts and provide a firm basis for developing novel proteins or modifying existing ones. Therefore, accurately labeling protein activity is a critical and significant task. Traditional experimental approaches for assessing protein function are accurate and reliable, but they are resource-intensive and time-consuming. Despite the exponential growth in protein sequences brought about by genomics and high-throughput sequencing, only a fraction of the total number of known and predicted protein sequences have had their functions fully defined. Experimental annotations cover less than 0.1% of the more than 179 million proteins in UniProtKB. [1-3]

The primary determinant of a protein's function is its structure, not the order of its amino acids. Therefore, learning more about the structure of a protein can help us understand how it functions. Because experimental procedures are costly and difficult, it is frequently only feasible to obtain structural information about a protein using computational methods. Protein structure prediction advances have allowed for the quick generation of many models for a single protein. Therefore, one of the most crucial jobs in evaluating the accuracy of any computer method to protein structure prediction is to compare the predicted protein models to the naturally determined structure as determined by experiment. Two primary challenges currently facing methods for assessing model quality are (1) the time and effort involved in choosing the most accurate models from a vast protein structure database in an effective manner, and (2) the difficulty of doing so. (2) When comparing two protein structures, there isn't a similarity measure that currently accounts for side chain orientation in addition to main chain carbon alpha (Cα) and side chain (SC) atom orientation.[4]

Given their close relationship, understanding protein architectures can teach us a lot about proteins and their roles. The same primary obstacle faces bioinformatics, drug research, and enzyme discovery: precisely predicting protein structures. The practice of trying to infer a protein's three-dimensional structure from its amino acid sequence is known as "protein structure prediction." There is a significant discrepancy between the sequences that are currently available and the structures that have been found experimentally because of the enormous amounts of protein sequence data that next-generation sequencing technologies are producing. The current experimental methods for predicting protein structures are costly and labor-intensive. Since computational approaches to protein structure prediction are more economical and efficient than experimental procedures, there is a growing focus on their development [5]. The majority of machine learning techniques for assessing quality do not make use of pair wise features, which are composed of several parameters and the majority of the data that is not input. Their sole concentration is on combining the output from other simple estimators.

One of the main bottlenecks in the way of better protein structure analysis methods certainly comes from the nature of its protein names. That happens due to a great extent because designing fusion proteins and optimizing crystallization intermittently makes the quantity of information one seeks to reduce and quality a concern. There is no argument with the masses that helped us describe the better structures in general. In a time of two or three times greater speed, many of the techniques in structural mass spectrometry can offer limited measures. Protein structure research benefits greatly from this experimental approach. On the other hand, structural mass spectrometry is unable to definitively determine the structure of a protein. Using a labeling chemical called covalent labeling, proteins can be permanently and irreversibly altered in the structural mass spectrometry method. This allows for the inference of protein regions exposed to the solvent. Computational techniques combined with structural mass spectrometry can provide a better understanding of the three-dimensional structure of proteins. [6-9]

Therefore, it is significant to develop computational methods to assist in the process of experimentally handling such enormous amounts of protein sequence data, since scaling up the strategy isn't easy. A small number of organizations and individuals have been working on algorithms, methods, and systems for predicting protein functions using advance computer technologies. Taking everything into account, these highly competitive models and algorithms are continuously being optimized and have proven to be incredibly effective in predicting protein function. When dealing with proteins from other species, it is crucial to comprehend and research the amino acid sequence. This proves that understanding the protein sequence is critical for predicting protein function.[10]

An ensemble model combining BTCN and BLSTM has been proposed in this study to enhance the efficiency of protein sequence classification. The BTCN module in the proposed architecture utilizes a sliding window method to capture the deep local dependencies in protein sequences, whereas the BLSTM module enables capturing the global interactions between amino acid residues. The BLSTM module also captures bidirectional deep long-range interactions between residues, which contribute to enhancing feature fusion and optimization. The BLSTM module also captures bidirectional deep long-range interactions between residues, which enhance feature fusion and optimization. With our method, complicated sequence-structure relationships can be effectively simulated by using longer-term bidirectional feature information. Our findings show that our methods perform better and could potentially mitigate the shortcomings of insufficient and partial feature extraction. In this research, we aim to build a DBN- and BLSTM-based protein sequence classification framework, using natural language processing (NLP) to derive contextual cues from the given dataset. From this point forward, this paper is organized into several parts: Section II presents a subsection on findings which are pertinent to a specific period of time covered by the latest pertinent review presented in several academic contributions in this area. Section III deals with the proposed protein classification system using a BTCN-FBLSTM combination over the given dataset. Section V presents the experimental setup conducted on analyzing the efficacy of the proposed classification method. Section V presents the conclusion of research work.

II.RECENT WORK IN THE FIELD OF PROTEIN SEQUENCE CLASSIFICATION

A crucial phase in the pharmaceutical industry is drug development. The cost and duration of creating new medications have been significantly lowered by computational methods. In order to address obstacles of all kinds, we will need to employ a range of drug screening and design techniques. This section focuses mostly on machine learning

and deep learning techniques that transcend the limitations of earlier research. Wei-Li et al. [11] combined loop-based resampling, near-native sampling, loop-based crossover, and stochastic rank-based selection with Rama torsion angle sampling to construct multi-objective evolutionary approaches. The energy function's error may be remedied by applying the secondary structural similarity criterion. Zhou et al. [12] used reinforcement learning based CNN framework which made protein secondary structure prediction possible. On top of CNN's abstraction capabilities and LSTM's sequence data analysis abilities, CDNN has a strong classification capability. Cross-entropy error has been used as a feature for the effective training of the model. The effectiveness of the approach is showcased through empirical validation on two separate datasets. Nevertheless, in spite of the anomalies, the projection remains credible. As a result, forecasts for the future are less reliable.
Deep ResNet was developed by you and others [13] in order to predict template-free protein folding and protein contact/distance. Recent advances in deep ResNet have been made in two areas: tertiary structure prediction and protein-protein interaction. In terms of utilizing inter-residue orientation data, the suggested 3D modeling method remains more basic and less sophisticated. The proposed deep ResNet can accurately fold the great majority of proteins made by humans.

According to Xu et al. [14], template-based structural modeling may be augmented with deep learning to derive a deep structural inference to forecast protein residue/residue interactions. They have used a tertiary structure prediction using huge set of single-domain proteins. A unique recurrent geometric network (RGN) was created by Du et al. [15] that can be used to predict protein structure from sequences. This method was not only efficient computationally but also offers several advantages when multiple sequence alignment isn't possible. RGN does this by describing the geometry of the C backbone using a straightforward method. This approach is limited to taking into account just local interactions (curvature and torsion angles) between C atoms in order to progressively reconstruct the structure of the backbone.
Guo et al. [16] improved protein sequence estimation by creating a multi-advanced deep belief network-based technique. Together, they were able to increase forecast accuracy by more than 80%. The outcomes also showed how well secondary structure could be predicted using hidden Markov model profiles that were created using emission/transition probabilities. The network will have uneven features, though. By computing the network parameters for the particular approach, the optimal values for the other parameters, such as the LR, network width, and depth, were found. To make it easier to compare the DNN model with the recommended Work, the author independently trained it using AC, CT, and LD.

Li et al. [17] has presented an auto-feature engineering based technique sequence prediction using DNs. As NN architecture can learn only through numerical input, the authors have altered the protein sequence by randomly allocating natural numbers to every amino acid. Gonzalez-Lopez et al. [18] presented a tokenization based method which involved assigning an integer token to each sequence triplet to represent it numerically. Two branches that resembled each other were fed and analyzed the pair representation of each protein in the NN. The FC layer of the design, along with the embedding and recurrent layers, each had a distinct function. We also employed Dropout and Branch normalization to guarantee consistent input and prevent over-fitting.

A recent problem for the bioinformatics community has been defining a protein's class only from its discovery, which presents numerous challenges. This enzyme class prediction provides researchers with a high probability when more proteins are added [19]. The primary goal of this work was to identify and execute the most effective machine learning technique for feature selection and prediction. Seven different procedures were tested and compared in order to determine the best categorization strategy. Vani and Bhavani [20] employed the SMOTE (Synthetic Minority Over-Sampling Technique) method to increase accuracy and balancing the data. In addition, SMOTE has proven to be quite effective in tackling protein categorization difficulties. It does have some downsides, such as a tendency to overfit and overgeneralize. To address this issue, the authors created and implemented a sampling approach.

Wang et. al. [21] looked into a variety of feature selection procedures to improve protein classification accuracy. These new sampling strategies improved accuracy while lowering the problem of unbalanced data, even if they were unable to balance the dataset. They asserted that feature selection is an important phase in the protein classification process since it helps extract the most relevant attributes from a dataset while also reducing the dataset's dimensionality. Selecting crucial characteristics simplifies protein classification. Unlike the prior approach, this work treated each protein as a point in feature vector space and used voxel-based descriptors to extract features. These solutions hold a lot of potential for solving problems in this critical field by combining novel methodologies and detailed designs.

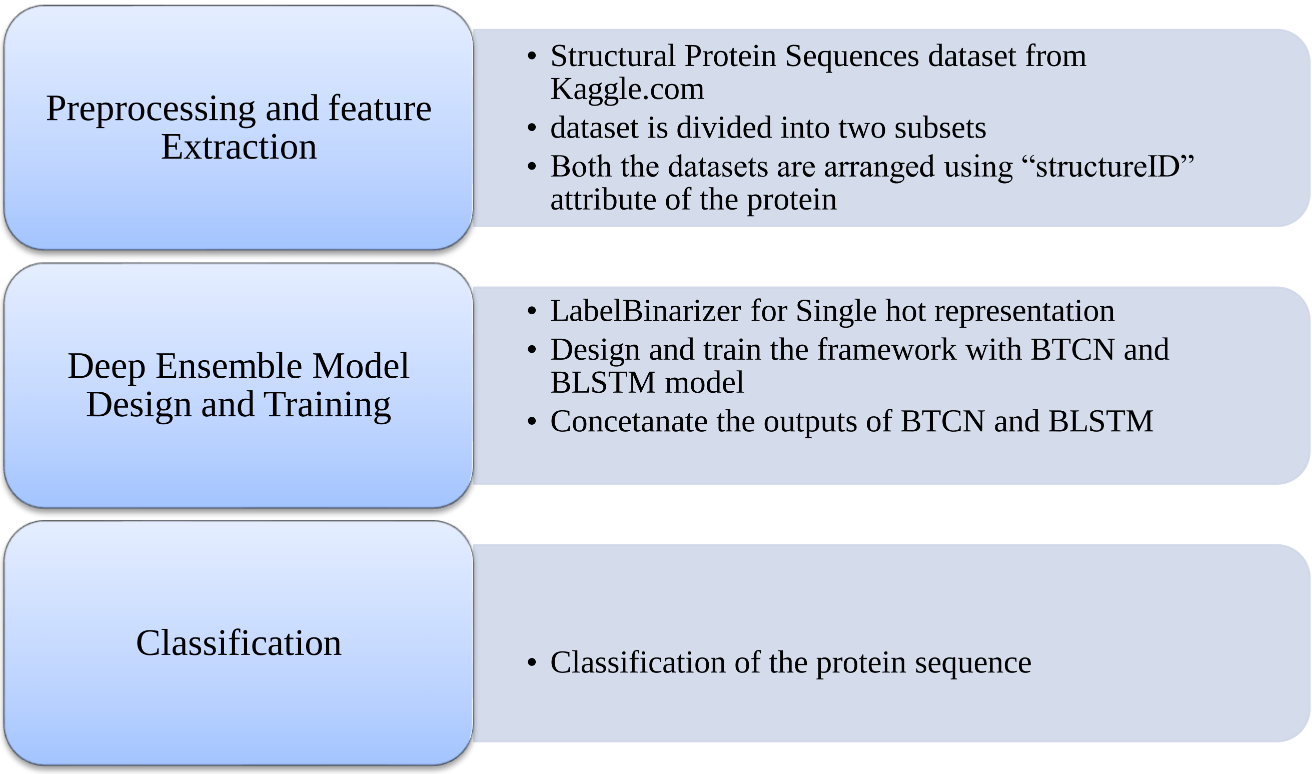
However, the accuracy of the classification depends solely on the depth of feature extraction, quality of dataset, feature dependencies and the respective training of the intelligent model.

III.PROPOSED FRAMEWORK FOR PROTEIN SEQUENCE CLASSIFICATION

For effective classification of various proteins into different types, this study describes methods such as data pre-processing, feature extraction, model training and validation, and classification. Generally, the entire work is divided into four steps as shown in figure 1 and elaborated as follows:

a) Dataset definition

In this particular work, the Protein Data Bank (PDB) has been used for the structural configuration data. It was developed at the RCSB's Research Collaboratory [22]. The first part of the collection is comprised of protein Meta data, which includes topics such as protein classification and extraction methods. The other half of the collection is made up of protein structural sequences. Both databases are organized according to the "structureID" property of the proteins that they hold as their guiding principle. In contrast to the dimensions of second data set which is 4,67,000-by-5, the first data set has the dimension of 1,41,000-by-14. Protein sequence datasets abound on Kaggle, with structural details included in a few of them. To fully grasp the relationships, functions, and roles that proteins play in biological processes, one must have a firm grasp of their three-dimensional structure. Such details are supplied by structural protein sequences. Protein structures found in experiments are part of the Protein Data Bank's collection of structural data that is accessible on Kaggle. Protein structures with exact 3D coordinates and associated sequence data are available in this set. Information included: protein sequences, annotations for secondary structures, three-dimensional coordinates (e.g., in PDB format), and related metadata (e.g., resolution and experimental methods).



Preprocessing and feature Extraction

- Structural Protein Sequences dataset from Kaggle.com
- dataset is divided into two subsets
- Both the datasets are arranged using “structureID” attribute of the protein

Deep Ensemble Model Design and Training

- LabelBinarizer for Single hot representation
- Design and train the framework with BTCN and BLSTM model
- Concetanate the outputs of BTCN and BLSTM

Classification

- Classification of the protein sequence

Fig.1 Proposed model for protein sequence classification

b) Data Preprocessing and Feature Extraction

Protein sequence analysis requires data pretreatment because it prepares the raw biological data for computational modeling and analysis. Unfortunately, direct analysis is difficult since raw sequence data is frequently noisy, fragmentary, and high-dimensional. By altering and purifying the data, data preparation allays these worries and guarantees that the information is prepared for additional processing, including statistical analysis and machine learning. Long segments of amino acids called protein sequences hold a variety of intricate biological information. Unprocessed protein sequence data, which is frequently obtained from PDB databases, may include mistakes, missing values, or unrelated information. Furthermore, the length, makeup, and structure of protein sequences might differ significantly, which makes direct comparison and analysis challenging. Data preparation is necessary for raising consistency, lowering noise, and improving quality of the data.

Researchers can employ preprocessing techniques to ensure that their models are trained on coherent, consistent, and pertinent data. Predictions will become more trustworthy and accurate as a result [23]. Working with big datasets and complex models is made feasible through preprocessing, which lowers the computing cost of the study. Protein sequence estimation involves a number of data preparation procedures, each of which aims to address a distinct problem with the raw data. Data transformation, cleansing, standardization, and dimensionality reduction are a few of these processes. The first stage of data preparation, referred to as "data cleaning," involves finding and fixing errors, conflicts, and missing numbers. Protein sequences may contain incorrect or unclear amino acids, denoted by "X" and other symbols. Two approaches to resolve these problems include removing undesirable sequences from the dataset or altering the sequence in light of biological knowledge. Furthermore, duplicate sequences may be included in huge protein databases, which, if not adequately managed, could produce biased outcomes. Each protein sequence has exactly one representation in the dataset now that duplication has been either eliminated or combined.

Sequence alignment is a crucial preprocessing step in protein sequence comparison. Sequences are aligned to maximize similarity and identify conserved regions using alignment techniques including pair wise alignment and multiple sequence alignment (MSA) [24]. Creating features for machine learning models requires first identifying valuable patterns and domains within the sequences. Sequence alignment is helpful for finding gaps and insertions during preprocessing on sequences so that they can be treated properly. Data is resized to fit inside a predefined range when it is normalized. One such way to guarantee consistency of features throughout the collection and eliminate bias is to standardize physicochemical characteristics (like hydrophobicity or charge) amongst sequences. Protein sequences are commonly represented as strings of amino acid symbols, which require transformation into a numerical representation in order to facilitate computational analysis. Protein sequences can be encoded using a variety of techniques, such as one-hot encoding, which assigns numerical values based on the characteristics of the amino acids, and physicochemical property encoding, which encodes each amino acid as a binary vector.

Protein sequence analysis relies heavily on feature extraction to clean up raw data and prepare it for tasks like function prediction, protein categorization, and interaction analysis. Computational techniques are necessary to comprehend the complex biological information that proteins, which are made up of amino acid sequences, carry. Feature extraction is performed to identify the most important traits from these sequences so that protein attributes may be efficiently and accurately estimated. Protein sequence estimate relies on feature extraction, which is discussed in this essay along with the various techniques used and the difficulties faced. Since protein structures are essential to biological processes, it is vital to comprehend their function and behavior to make progress in areas such as medicine development, bioengineering, and illness therapy.

Here, the categorical values are obtained for all 10 labels which are then transformed into a single hot representation using LabelBinarizer. Values are assigned a 1 in a single hot representation if they exist, and a 0 otherwise. Sequences are further pre-processed by using the Keras library's Tokenizer method, which converts each character in the sequence to a number. The length of every sequence is also uniformized for precise processing. In this case, there is a 256 character limit.

c) Model Training and Testing

This stage presents the training and testing of the proposed framework for the Protein sequence classification. Here we have used an ensemble technique of BTCN and BLSTM to exploit the local features and global features. BTCN is the modified version of TCN which has been proposed to overcome the issues encountered in the application of gradient descent over the sequence processing. It has also presented the advantages of lower memory requirement and faster computational due to parallel processing over the conventional techniques. It is a feedforward multilayered hierarchical network in which a pool of convolutional kernels is used by each layer to perform a distinct transformation. The convolution technique aids in the extraction of a significant feature from locally related data points. After receiving the convolutional kernels' output, the activation function uses it to aid with abstraction learning and incorporate non-linearity into the feature space. This non-linearity facilitates the learning of semantic differences between segmented sequences by producing distinct activation patterns for distinct responses. Subsampling is often used to shield the input from geometric distortions and summarize the results after the nonlinear activation function output. The most distinctive features of TCN are weight sharing, parallel processing, automatic feature extraction, and hierarchical learning. The models use input from a condensed portion of the layer above each hidden node. By leveraging on these local connections, convolutional layers are better than fully-linked ones in handling spatial reliance in data and extracting valuable high-level properties. TCN has the same structure as a 1 dimensional fully connected CNN network. The length of the hidden layers is kept equal to that of input layer. Furthermore, zero padding ensures the exact one-to-one mapping of input sequences to the output sequences. Dilated convolutions have been used here to extract the long term past information and to increase the receptive field, which could not be done effectively in casual convolutions. However, this network can only pass the information from past to future, which cannot be suitable for the amino acid sequence estimation problem where the comprehensive feature extraction is depending upon both past and future positions. To overcome this issue, we have used a bidirectional framework of TCN known as BTCN which consists of two TCNs: forward and backward. The forward TCN is similar to the conventional TCN, but the backward TCN is created by reversing the sequence at the input layer.

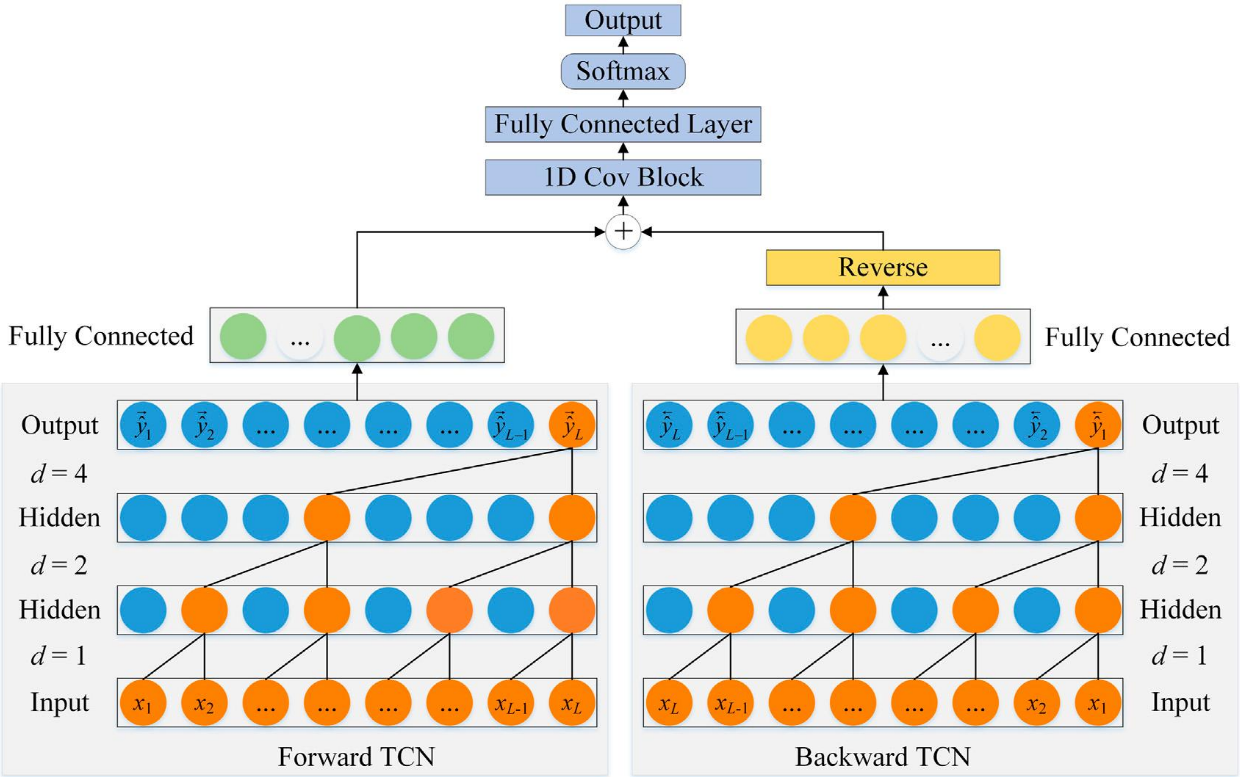

Output
Softmax
Fully Connected Layer
1D Cov Block
+
Reverse
Fully Connected
...
Fully Connected
Output
$d = 4$
Hidden
$d = 2$
Hidden
$d = 1$
Input
$\hat{\vec{y}}_1$ $\hat{\vec{y}}_2$ ... ... $\hat{\vec{y}}_{L-1}$ $\hat{\vec{y}}_L$
x_1 x_2 ... ... x_{L-1} x_L
Forward TCN
$\hat{\overleftarrow{y}}_L$ $\hat{\overleftarrow{y}}_{L-1}$ ... ... $\hat{\overleftarrow{y}}_2$ $\hat{\overleftarrow{y}}_1$
x_L x_{L-1} ... ... x_2 x_1
Backward TCN

Fig 2. The architecture of BTCN.

The framework of BTCN can be mathematically represented in terms of forward TCN $\vec{T}\left(\vec{x}\right)$ and backward TCN $\overleftarrow{\mathrm{T}}\left(\overleftarrow{x}\right)$ as

$$\hat{Z} = soft\max\left(f\left(W\left(Conv\left(\vec{T}\left(\vec{x}\right)\oplus\overleftarrow{\mathrm{T}}\left(\overleftarrow{x}\right)\right)\right)+b\right)\right)$$

Here, $\left(\vec{x}\right)$ is the forward input sequence and $\left(\overleftarrow{x}\right)$ The input sequences run in reverse. The symbols W and b are used to denote the weighting matrix and the bias term, respectively. Here, we are using the 11-dimensional PCscores descriptor as one amino acid feature on the input protein sequences of 300 amino acids. The eleven columns of an 11 x 300 matrix represent one input feature of that single protein. The max-pooling layer then down-samples along the sequence dimension of those feature maps to reduce dimensionality after the convolutional layers apply them on grouped input from nearby locations. In each of the convolutional layers, a set of filters works like a sliding window taking a distinct feature set from the data channels of the neighboring layer. This proposed research work ensembles the BTCN network with a recursive type of framework called BLSTM, composed of forward and backward LSTM. The architectural framework of the BLSTM unit is shown in fig 3. The global features of the amino acid sequence can be efficiently extracted using this bidirectional variant of LSTM. The bidirectionality of this model also affords it increased ability to capture deeper long-range dependencies among the amino acid sequences, thereby further enhancing its ability to model the complex relationship between the protein sequence and its structure. It could assign importance to what information to retain and what to discard automatically. A typical LSTM cell represents the input feature at some time t by xt, the output by ht, and the cell state by ct. The LSTM unit calculates input gate (i), forget gate (f), and output gate (o) as follows: [25]

$$f_t = \sigma\left(W_{xf} \times x_t + W_{hf} \times h_{t-1} + b_f\right)$$

$$i_t = \sigma\left(W_{xi} \times x_t + W_{hi} \times h_{t-1} + b_i\right)$$

$$o_t = \sigma\left(W_{xo} \times x_t + W_{ho} \times h_{t-1} + b_o\right)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tanh\left(W_{xc} \times x_t + W_{hc} \times h_{t-1} + b_c\right)$$

$$h_t = o_t \circ \tanh\left(c_t\right)$$

where the weight and bias terms are represented by W and b respectively. σ is the sigmoid function, element-wise multiplication is represented by $\circ$. Two dropout layers are added in this model to ensure the gradient stabilization at the time of training.

The outputs of both the networks, BTCN and BLSTM are concatenated in the end and are used for the classification of protein sequence. The extracted features are processed and oputimized over the residual block and the combination of a fully connected layer and softmax function is used to perform the classification.

d) Classification

Basically, to categorize the protein sequences, one has to combine the outputs of the two different networks BTCN and BLSTM. Here, the classification is performed with the help of a fully connected layer with softmax. Most proprietary efforts in protein engineering are directed toward finding protein sequences that mediate the highest substrate exchange. There are twenty standard types of amino acids in the protein sequences database (A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, and Y) and six non-standard amino acid types (B, X, and Z). These non-standard types can all be clumped together due to their very low occurrence. We assumed amino acid types in protein sequences to be of 21 kinds and formulated the problem as a multi-class classification problem.

IV.IMPLEMENTATION AND RESULTS

Intelligent prediction and classification models make use of the Protein Data Bank (PDB) dataset to assess the accuracy in prediction of proposed methods for protein attributes. The collection includes 14,991 proteins selected by us from the PDB. Proteins that were duplicated in the training set and test set were thus discarded. In addition, proteins longer than 800 or shorter than 40 were discarded. The final dataset has a total of 14,562 protein chains. Random splitting of the dataset was done into three sections for better evaluation- the test set (1,456), the validation set (1,456), and the training set (11,650). All results for all experiments are derived from averaging over three separate, independent experiments. The feature matrix under consideration with sequence length L for 21 classes could be derived from semantic one-hot encoding, as stated above. Because of the independent characteristics of the amino acid structures, one-hot encoding could also be called orthogonal coding. Hyperparameters used for the training of the proposed ensemble model are presented in Table 1.

Table 1. Hyperparameters

Parameters	Description/Values		Parameters	Description/Values
Learning rates	0.001, 0.005, 0.01		Number of filters	512

Optimization technique	Gradient Descent		Objective	mse
Number of Fully connected layers in BTCN	20		Initialization	HeNormal
Number of hidden layers in LSTM	2500		Number of residual blocks	6
Sliding window size	13, 15, 17		Size of filter	5

The technique is put to questionnaire during its evaluation assessing accuracy as well as a comparison with some conventional techniques. The corresponding results of comparison are listed in table 3.

Table 2. Comparison of techniques

Technique	**Accuracy**
CNN	82.35
BTCN	85.71
BTCN+BLSTM	88.59

It is clearly shown that the performance of the proposed ensemble technique is better than the conventional techniques i.e. CNN based and BTCN based classification. The training performance is shown in figure 4 and 5 which display the variation of cost function with respect to number of epochs and variation of training and validation accuracies with respect to number of iterations.

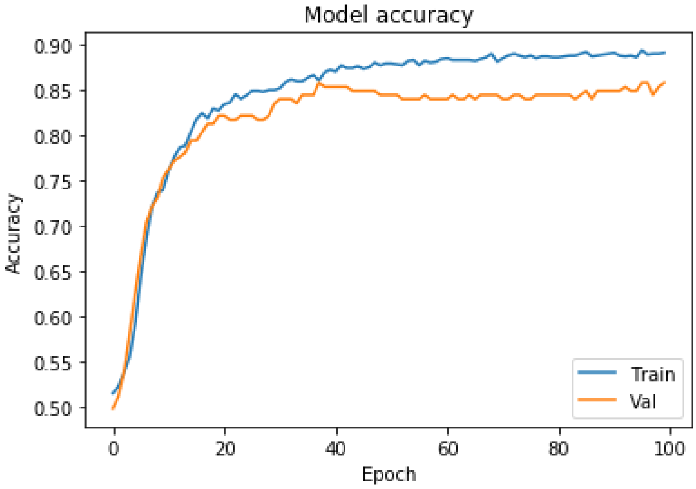
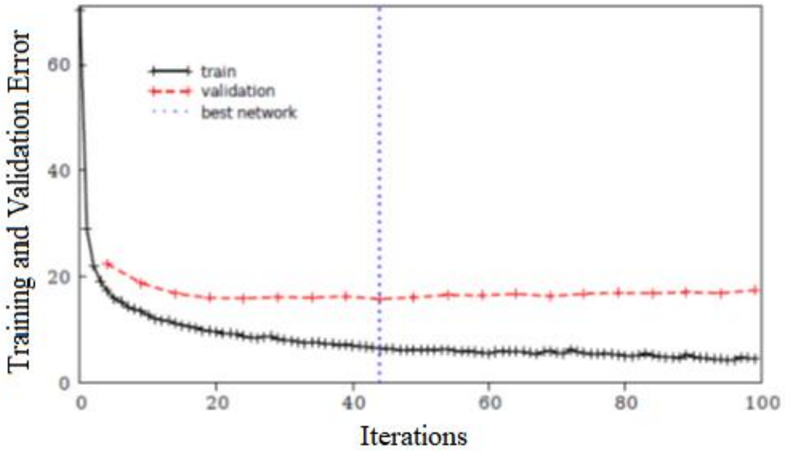


Figure 4 Modeling Accuracy wrt number of iterations



b

Figure 5 Errors wrt number of epochs

The convergence of the error graph relates the training performance of its model. Cumulative evaluations of the figures give a clearer idea of the overall system performance. The existence of a large flexible receptive field allows the proposed model to be practically used for long sequences of higher dimensions. It helps explore the complex relationships between the sequence and the structure of the protein by capturing longer-term dependencies among the residues.

V.CONCLUSION

We are introducing a novel AI-based method in this article for protein sequence classification. To delve deeper into protein-chain intricacy, this study will assemble both BTCN and BLSTM networks. In order to dissect protein sequences, the sliding window technique was used. The convolutional network possibly can focus on local dependencies within these segments, while the BLSTM network could possibly extract global relationships amid residues. The BLSTM was selected for the task, considering that amino acid classification is dependent upon the previous and following secondary features. We applied the bidirectional network to maintain the feature information of intervening residual blocks in addition to pulling bidirectional features, which is in contrast to traditional temporal convolutional networks pulling unidirectional features. The evaluation shows that the ensemble model gives a slightly lower Accuracy on predicting Protein Structures, even though MOECNN received the highest Accuracy.

Competing Interests: *The authors declare that they have no conflict of interest.*

Funding Information: *The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.*

Author contribution: *All the authors have contributed equally.*

Data Availability Statement: *Not Applicable*

Research Involving Human and /or Animals: *Not Applicable*

Informed Consent: *Not Applicable*

REFERENCES

[1] R. C. Tillquist, "Low-dimensional representation of biological sequence data," in *Proc. 10th ACM Int. Conf. Bioinformatics, Computational Biology and Health Informatics*, New York, NY, USA, 2019, pp. 555.

[2] T. Villmann, F.-M. Schleif, M. Kostrzewa, A. Walch, and B. Hammer, "Classification of mass-spectrometric data in clinical proteomics using learning vector quantization methods," *Brief. Bioinform.*, vol. 9, no. 2, pp. 129-143, 2008.

[3] E. C. Alley, G. Khimulya, S. Biswas, M. AlQuraishi, and G. M. Church, "Unified rational protein engineering with sequence-based deep representation learning," *Nat. Methods*, vol. 16, no. 12, pp. 1315-1322, 2019.

[4] C. Fang, Y. Shang, and D. Xu, "Mufold-ss: New deep inception-insideinception networks for protein secondary structure prediction," *Proteins Struct. Funct. Bioinforma.*, vol. 86, pp. 592–598, 2018.

[5] A. Kryshtafovych, T. Schwede, M. Topf, K. Fidelis, and J. Moult, "Critical assessment of methods of protein structure prediction (CASP)—Round XIV," *Proteins Struct. Funct. Bioinforma.*, vol. 89, pp. 1607–1617, 2021.

[6] P. Kumar, S. Bankapur, and N. Patil, "An enhanced protein secondary structure prediction using deep learning framework on hybrid profile-based features," *Appl. Soft Comput.*, vol. 86, p. 105926, 2020.

[7] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 156-165.

[8] A. Nambiar, M. Heflin, S. Liu, S. Maslov, M. Hopkins, and A. Ritz, "Transforming the language of life: transformer neural networks for protein prediction tasks," in *Proc. 11th ACM Int. Conf. Bioinformatics, Computational Biology and Health Informatics*, 2020, pp. 1-8.

[9] M. Heinzinger *et al.*, "Modeling aspects of the language of life through transfer-learning protein sequences," *BMC Bioinform.*, vol. 20, no. 1, pp. 1-17, 2019.

[10] A. Madani *et al.*, "Progen: Language modeling for protein generation," *arXiv preprint arXiv:2004.03497*, 2020.

[11] L. Wei *et al.*, "Improved prediction of protein–protein interactions using novel negative samples, features, and an ensemble classifier," *Artif. Intell. Med.*, vol. 83, pp. 67-74, 2017.

[12] Y. Z. Zhou, Y. Gao, and Y. Y. Zheng, "Prediction of protein-protein interactions using local description of amino acid sequence," in *Advances in Computer Science and Education Applications*, M. Zhou and H. Tan, Eds. Berlin, Germany: Springer, 2011, pp. 254-262.

[13] Z. H. You *et al.*, "Prediction of protein-protein interactions from amino acid sequences using a novel multi-scale continuous and discontinuous feature set," *BMC Bioinform.*, vol. 15, no. 15, pp. 1-9, 2014.

[14] H. Xu, D. Xu, N. Zhang, Y. Zhang, and R. Gao, "Protein-protein interaction prediction based on spectral radius and general regression neural network," *J. Proteome Res.*, vol. 20, no. 3, pp. 1657-1665, 2021.

[15] X. Du *et al.*, "DeepPPI: boosting prediction of protein–protein interactions with deep neural networks," *J. Chem. Inf. Model.*, vol. 57, no. 6, pp. 1499-1510, 2017.

[16] Y. Guo and X. Chen, "A deep learning framework for improving protein interaction prediction using sequence properties," *bioRxiv*, 2019.

[17] H. Li, X. J. Gong, H. Yu, and C. Zhou, "Deep neural network based predictions of protein interactions using primary sequences," *Molecules*, vol. 23, no. 8, p. 1923, 2018.

[18] F. Gonzalez-Lopez *et al.*, "End-to-end prediction of protein-protein interaction based on embedding and recurrent neural networks," in *2018 IEEE Int. Conf. Bioinformatics Biomed. (BIBM)*, 2018, pp. 2344-2350.

[19] A. Rives *et al.*, "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 118, no. 15, 2021.

[20] K. S. Vani and S. D. Bhavani, "SMOTE based protein fold prediction classification," in *Advances in Computing and Information Technology: Proceedings of the Second International Conference on Advances in Computing and Information Technology (ACITY) July 13-15, 2012, Chennai, India-Volume 2*, Berlin, Heidelberg: Springer, 2013, pp. 541-550.

[21] X. Wang, Y. Wu, R. Wang, Y. Wei, and Y. Gui, "A novel matrix of sequence descriptors for predicting protein-protein interactions from amino acid sequences," *PLoS ONE*, vol. 14, no. 6, p. e0217312, 2019.

[22] G. Wang and R. L. Dunbrack, "Pisces: Recent improvements to a PDB sequence culling server," *Nucleic Acids Res.*, vol. 33, pp. W94-W98, 2005.

[23] Z. Wu, S. J. Kan, R. D. Lewis, B. J. Wittmann, and F. H. Arnold, "Machine learning-assisted directed protein evolution with combinatorial libraries," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 116, pp. 8852-8858, 2019.

[24] M. Chatzou *et al.*, "Multiple sequence alignment modeling: methods and applications," *Briefings in Bioinformatics*, vol. 17, no. 6, pp. 1009–1023, 2016.

[25] L. Yuan, Y. Ma, and Y. Liu, "Ensemble deep learning models for protein secondary structure prediction using bidirectional temporal convolution and bidirectional long short-term memory," *Front. Bioeng. Biotechnol.*, vol. 11, p. 1051268, 2023.