WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

An Unknown Protein Sequence-Tuned Human Protein Function Classification Using A Novel Ln-Px-Rnn and Egk-Fuzzy

Dr. G Janakasudha[1], Mithun S[2], Aswin V V[3], Pratyush Ganesan Iyer[4], Arihant A M[5], Ajay Sabari S B[6]

[1]Associate Professor, Computer Science and Engineering, Sri Venkateswara College of Engineering, Sriperumbudur
[2]Artificial Intelligence and Data Science, Computer Science and Engineering,Sri Venkateswara College of Engineering.Sriperumbudur
[3]Artificial Intelligence and Data Science, Computer Science and Engineering, Sri Venkateswara College of Engineering. Sriperumbudur
[4]Artificial Intelligence and Data Science, Computer Science and Engineering, Sri Venkateswara College of Engineering. Sriperumbudur
[5]Computer Science and Engineering, Sri Venkateswara College of Engineering, Sriperumbudur
[6]Computer Science and Engineering, Sri Venkateswara College of Engineering, Sriperumbudur

Abstract:

Basically, protein plays a pivotal role in biological processes like cellular structure, function, and regulation. Accurate protein function identification helps to develop innovative therapeutic strategies, drug design, and understanding of disease mechanisms. However, the existing works struggled to handle the unknown protein sequence. Therefore, this paper proposes an unknown protein sequence-aware human protein function classification using LN-PX-RNN and EGK-Fuzzy. Firstly, the historical dataset is gathered. Then, the protein sequences are extracted, followed by pre-processing. Next, the sequence is translated into codon form. Also, the domain and motif are identified from the pre-processed data. Furthermore, the macromolecular separation is done via D(DB)2SCAN from the translated sequences. Then, one-hot encoding is carried out. In addition, the semantic features are obtained by using the D-ProtBERT. Also, the evolutionary information and essential features are extracted from the macromolecular separated data. Finally, the proposed LN-PX-RNN is employed to classify the function of the protein sequence. Here, the function name and unknown function are classified from the protein sequence by using the LN-PX-RNN. From the unknown function, the sequence is split, followed by similarity evaluation. Next, the location and sequence variation are analyzed through the EGK-Fuzzy algorithm. Then, the function for the unknown sequence is identified by considering the past sequences. Finally, the proposed work obtains high reliability with a 99.12% f-measure.

Keywords: Protein Sequence (PS), Protein Function (PF), Large Language Model (LLM), Macromolecular Separation (MS), Layer Normalization-Poly Exponential-Recurrent Neural Network (LN-PX-RNN), Exponential Gustafson-Kessel–Fuzzy (EGK-Fuzzy), and Deep Learning (DL).

1. INTRODUCTION

Nowadays, the demand for advancement in protein-related research areas has exponentially increased to obtain precise and timely disease treatment (Sunny et al., 2023) (Wu et al., 2023). Basically, proteins are one of the fundamental macro-molecules in nature (Jisna& Jayaraj, 2021). According to the homologous structures present in the Protein Data Band (PDB), the protein structure models are commonly classified as Template-Based Modelling (TBM) and Template-Free Modelling (TFM) (Nallasamy&Seshiah, 2023). In essence, proteins perform crucial functions in numerous biological processes. Therefore, it is essential to understand protein structure, function, and interactions for improving drug discovery and molecular biology (Tasnim et al., 2024) (Jang et al., 2024). Over the past decade, enormous researchers conducted numerous studies to predict protein structures. However, functional classification based on structural information of protein sequences is a foremost research problem (Hegedűs et al., 2022) (Chauhan et al., 2022).

Recently, Natural Language Processing (NLP) and computer vision technique were coupled to learn crucial patterns from growing biological databases (Ferruz et al., 2023). Also, Artificial Intelligence (AI) methodologies like Machine Learning (ML) and DL are promising tools in computer vision techniques, such as speech recognition and protein classification (Pearce & Zhang, 2021). An automated protein function classification model encompasses key processes like dataset collection, pre-processing, feature extraction, and classification. Firstly, the protein sequence datasets were gathered and then pre-processed for data cleaning and duplication removal. Then, the features were extracted and classified via the neural network (Khattak et al., 2023). Furthermore, ML techniques like Support Vector Machine (SVM), Random Forest (RF), and Decision Tree (DT) were established in the conventional works to perform protein-protein interaction prediction (Cunningham et al., 2023) (Cao et al., 2024). Likewise, the existing methods employed DL techniques like Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), and Deep Neural Network (DNN) to classify the protein function of the protein sequences (Soleymani et al., 2022). Nevertheless, the existing methodologies struggled to annotate the protein function (Zheng et al., 2024). Also, none of the traditional works focused on protein function classification regarding unknown protein

sequences. To address this issue, this work proposes a novel architecture named unknown protein sequence-aware human protein function classification using LN-PX-RNN and EGK-Fuzzy.

1.1 Problem statement

The drawbacks of the conventional frameworks are given below,

- None of the traditional methodologies was generalized well enough to estimate the function of unknown sequences that occurred in protein sequences.
- The existing work (Chu et al., 2022) failed to focus on the semantic and structural features of proteins, thus affecting the performance of the accurate function annotation.
- The prevailing framework (Jiang et al., 2021) doesn't concentrate on the macromolecule type, which mainly affects the model's efficacy.
- The existing approach (Li et al., 2022) failed to consider the motif and domain of the protein, thereby leading to missing out on the key interaction sites.
- Some of the conventional methodologies consumed more time to process the large-scale protein sequence data.

1.2 Objectives

The major objectives of the proposed methodology are discussed further,

- A novel EGK-Fuzzy helps to identify the function of the classified unknown sequence.
- An effective D-ProtBERT is employed to extract the semantic and structural features, thus increasing the system's performance.
- In the proposed work, the sequences are grouped according to the macromolecular type via $D(DB)^2SCAN$.
- The proposed work identifies the motif and domain from the protein sequences to enhance the framework's reliability.
- In the research methodology, a sequence is translated into codon form to improve computational efficiency.

The paper is organized as: Section 2 discusses the related frameworks, Section 3 elucidates the proposed mechanism, Section 4 illustrates the results and discussion, and Section 5 concludes the paper with future ideas.

2. LITERATURE SURVEY

(Jiang et al., 2021) implemented a DL-based protein subcellular and suborganellar localization prediction framework along with residue-level interpretation. Primarily, the numerous datasets were gathered from the publically available resources. Thereafter, Multiple Localization-based Deep learning (MULocDeep) was established to estimate the 10 main subcellular localizations and 44 suborganellar localizations. This framework had better outcomes in protein localization. However, this model failed to focus on the macromolecular type of the protein sequence.

(Ahsan et al., 2022) demonstrated a model called imbalanced protein sequences classification using DL approaches. Initially, the polynomial and time matrix datasets were collected. Here, the Decision Tree (DT), K-Nearest Neighbour (KNN), and Long Short Term Memory (LSTM) were utilized to predict the imbalanced protein sequence based on feature selection, distance, and time series, respectively. Thus, the experimental results proved that the model was highly accurate and feasible. However, this approach was less effective owing to the limited data representation.

(Min et al., 2021) investigated pre-trained model-based deep bidirectional protein sequence representations using structural information. Here, masked language modelling was established to extract the structural information from the protein sequences. Also, the pre-trained bidirectional RNN was introduced to classify the function of the protein structure. This model efficiently analyzed the protein structure via structural information. But, this model was inadequate to handle long protein sequences.

(Chu et al., 2022) investigated machine learning-based liquid-liquid phase separating protein prediction. Initially, the componential and sequential information of the protein sequence were computed during the protein embedding phase. Thereafter, the ensemble-ML algorithms were used to predict the function of the protein sequence effectively. However, this framework failed to consider the semantic and structural features of the protein, thereby degrading the model's significance.

(Shin et al., 2023) examined a deep neural network-based aptamer-protein interaction prediction via pre-trained encoders. Here, the pre-trained encoders were utilized to represent the structural information of the protein sequences. Furthermore, the deep transformer-encoder was employed to predict the aptamer and protein sequences effectively. This model significantly estimated the protein function. However, this model struggled to reveal the affinity of aptamer interaction via biological experiments.

(Li et al., 2022) propounded an innovative model-named self-attention with deep neural network-based protein-protein interaction prediction. Primarily, the global and local features of the protein sequences were extracted from the dataset. Then, the features were fed into the self-attention neural network, where the protein function was obtained. This framework provided insight into significant drug design and disease prevention schemes. However, this model failed to identify the motif and domain of the protein sequences.

(Zhong & Gu, 2022) investigated a clustering deep recurrent neural network-based local protein 3D structure prediction. Here, the entire dataset was split into multiple cluster sub-trees. Then, the RNN was trained for each cluster in the sub-trees. Now, the RNN effectively predicted the distance matrices, torsion angles, and secondary structures of protein sequence segments. Nevertheless, this approach had high computation costs and training time.

(Zeng et al., 2024) demonstrated a graph neural network-based multi-category prediction of protein-protein interactions. Here, a graph isomorphism network was employed to extract the global graph features. Likewise, to extract the local subgraph features, a graph isomorphism network with kernel was utilized. Finally, the features were inputted into the graph neural network, thus predicting the protein functions. This framework had high reliability in protein-protein interaction prediction. But, it struggled to handle the sequences with high-level similarity.

(Dang & Vu, 2024) examined deep learning-based protein-protein interaction prediction using a protein language model. Here, the multi-kernel pooling convolutional neural networks were utilized to predict the protein-protein interaction. This approach had low computational cost and better efficiency. Nevertheless, this model had overfitting issues, thus causing poor performance on new protein sequences.

(Dhusia& Wu, 2021) employed a protein-protein association rate classification based on biophysical informatics. Here, the physics-based coarse-grained simulations were integrated with the neural network to compute the associate rate of the protein-protein interaction. This framework provided significant insight into protein interactions. However, this framework had biases and uncertain outcomes during protein-protein association rate prediction.

3. PROPOSED METHODOLOGY FOR PROTEIN FUNCTION CLASSIFICATION USING LN-PX-RNN AND EGK-FUZZY

In the research framework, an unknown protein sequence-tuned human protein function classification is implemented by using the LN-PX-RNN and EGK-Fuzzy. The proposed work is generalized well enough to classify the function of the protein, thus improving the model's trustworthiness. The diagrammatic design of the proposed approach is illustrated in Figure 1.

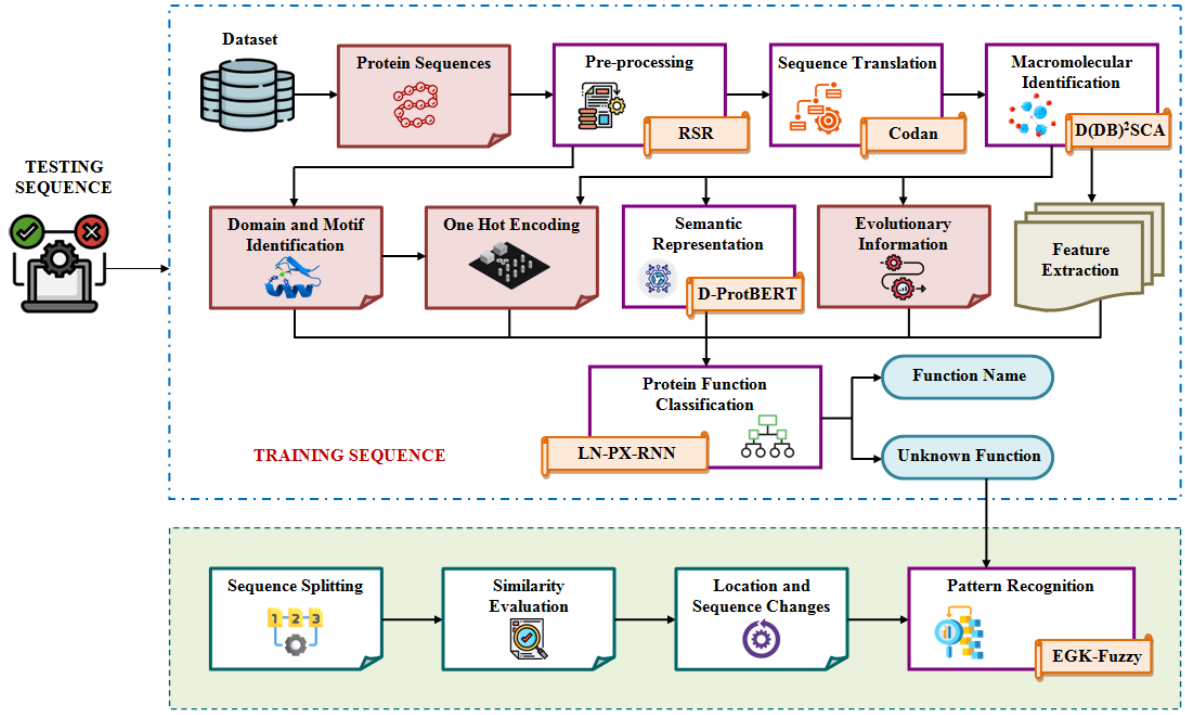

Figure 1: The structural illustration of the research methodology

Furthermore, the proposed approach involves several key processes like sequence translation, macromolecular separation, semantic representation, protein classification, and pattern recognition. The mathematical modeling of the proposed model is derived further,

3.1 Dataset

Initially, the historical dataset named Structural Protein Sequences (SPS) dataset is gathered from the publically available resources to pre-train the proposed protein function classification model. The input data $\left(\Im_n\right)$ is mentioned as,

$$\Im_n \rightarrow \left\langle \Im_1, \Im_2, \ldots\ldots \Im_N \right\rangle \text{Here, } n = 1 to N \tag{1}$$

Where, N denotes the number of collected input data.

3.2 Protein sequences

From the gathered data $\Im_n$, the protein sequences are extracted to perform further computation processes. The protein sequence provides detailed information about the biological macromolecules. The extracted protein sequences are depicted as $\left(\wp_\infty\right)$.

$$\wp_\infty = \left(\wp_1, \wp_2, \ldots\ldots \wp_F\right) \tag{2}$$

Where, F demonstrates the number of $\wp_{\infty}$.

3.3 Pre-processing

In this phase, the $\wp_{\infty}$ is subjected to pre-processing, where the data quality is considerably enhanced. In common, the raw historical dataset contains redundant information, thus leading to biased outcomes in protein function classification. Therefore, the redundant protein sequences $(\Re)$ are eliminated from the $\wp_{\infty}$ to improve the model's outcomes. The Redundant Sequence Reduction (RSR) is done as shown below,

$$\zeta_{data} = \wp_{\infty} - \Re \tag{3}$$

Here, ζ_{data} illustrates the pre-processed data.

3.4 Sequence translation

Thereafter, the pre-processed data ζ_{data} is fed into the sequence translation to increase the computational efficiency. In basic, codon-based sequence translation is the task of transforming a DNA or RNA sequence into its corresponding amino acid sequence concerning the genetic code. Here, the sequence translation is done via a codon scheme, which is discussed below,

- Commonly, the genetic code involves several codons, which are sequences of three nucleotides that correspond to particular amino acids during protein synthesis.
- During sequence translation, the DNA sequence is initially converted into an m-RNA sequence. Thereafter, the m-RNA sequence (m_{RNA}) is partitioned into codons.

$$\zeta_{data} \xrightarrow{transform} m_{RNA} \tag{4}$$

- Afterward, the amino acid sequence (α_{seq}) is generated according to the generated codons.

$$\lambda_{codon} = \sum_{z=1}^{Z} \langle C_1 \ldots C_z \rangle \xrightarrow{generate} \alpha_{seq} \tag{5}$$

Where, Z specifies the number of divided codons C_z . Thus, the translated sequence is represented as (λ_{codon}) .

$$\lambda_{codon_o} = \left(\lambda_{codon_1}, \lambda_{codon_2}, \ldots\ldots \lambda_{codon_O}\right) \tag{6}$$

Here, O illustrate the number of translated sequences.

3.5 Macromolecular separation

Next, the translated sequence λ_{codon} is inputted to the macromolecular separation. Generally, macromolecular separation aids in improving the purity of the target protein because it isolates the specific proteins from a complex mixture in a precise manner. Here, the proposed DunnDavies-Bouldin Density-Based Spatial Clustering of Applications with Noise (D(DB)^{2}SCAN) is introduced to group the translated sequence into macromolecular sequences like DNA, RNA, or protein based on their sequence similarity. The DBSCAN is more robust to noise and outliers. Also, it doesn't require a pre-defined number of clusters. However, it was sensitive to epsilon and minpts values, and the improper epsilon and minpts values could not support clusters with different densities. To improve the clustering quality, the proposed method employs the Dunn Davies-Bouldin (DDB) index to compute the minpts-eps points with respect to the average between the data points. The steps involved in the proposed D(DB)^{2}SCAN are explained below,

Step 1: Primarily, the input parameters, such as epsilon and minimum points are initialized. The epsilon refers to the maximum distance between two points. It aids in identifying the neighbours. Likewise, the minimum points are called the minimum number of points needed to create a cluster. The proposed method establishes the DDB index $(\partial^2 \beta)$ to estimate the epsilon (ψ) and minpts (Φ).

$$\partial^2 \beta = \frac{1}{O} \sum_{o=1}^{O} \max_{k \neq o} \left(\frac{\lambda_{codon_o} + \lambda_{codon_k}}{Dis(o,k)} \right) \tag{7}$$

Where, k depicts the constant value, and Dis illustrates the distance among the data points. Then, the clustering process is initiated by considering all the data points as unvisited. For each point, the following process is carried out.

Step 2: Thereafter, the Euclidean distance is calculated among the data points to identify the neighbour point regarding epsilon.

$$\chi\left(\lambda_{codon_1},\lambda_{codon_o}\right)=\sqrt{\sum_{o=1}^{O}\left(\lambda_{codon_1}-\lambda_{codon_o}\right)^2} \tag{8}$$

Step 3: Subsequently, the density of the cluster is estimated by checking if the epsilon neighbourhood of the point has at least a minimum number of points. If the point has a minimum number of points, then the point is marked as the core point and the cluster is created according to the core point. But, if the point has few minimum numbers of points, then the point is known as noise.

$$\begin{cases} IF\left(\lambda_{codon_o}>\Phi\right), & core\,point \\ IF\left(\lambda_{codon_o}<\Phi\right), & noise \end{cases} \tag{9}$$

Step 4: Similarly, the cluster is expanded by verifying whether the core point has been visited or not. If it is unvisited, then mark it as visited. Likewise, if any of the points is not part of any cluster, then the point is included in the current cluster.

$$\begin{cases} IF\left(unvisited\right) & THEN\ visited \\ IF\left(idol\right) & THEN\ add\,currentcluster \end{cases} \tag{10}$$

Finally, the above-mentioned steps are repeated continuously until all the data points are marked as visited. Thus, the macromolecular separated data is exhibited as $\left(\mathrm{M}_{\nabla}\right)$.

$$\mathrm{M}_{\nabla}=\left\langle \mathrm{M}_1,\mathrm{M}_b,\ldots\ldots\mathrm{M}_B\right\rangle \tag{11}$$

Here, $b=1to\,B$ demonstrates the number of grouped data M_{∇}.

3.6 Domain and motif identification

In the same manner, the domain and motif are identified from the pre-processed data ζ_{data}. In essence, domain and motif identifications are promising tools in computational biology, especially in the study of protein sequences and biological molecules. In the proposed work, the domains and motifs are identified from ζ_{data} by using the PROSITE database.

- ***Domains:*** The domains refer to the unique functional and structural units of a protein. A single domain involves various proteins having similar functions across biological backgrounds.

$$\partial om=\xi_{data}\left(PROSITE.com\right) \tag{12}$$

- ***Motifs:*** Basically, motifs are tiny and conserved sequences within proteins or nucleic acids that are related to particular biological functions.

$$\mu ot=\xi_{data}-\left\{\iota\right\}-\left\{\beta\chi\right\}-\left\{\varpi\right\} \tag{13}$$

Where, ι, β, χ and ϖ denote the pre-defined parameters. Thus, the identified domains and motifs are illustrated as $\left(\partial om\right)$ and $\left(\mu ot\right)$, respectively.

3.7 One-hot encoding

In this step, the macromolecular separated data M_{∇}, domains ∂om, and motifs μot are inputted to the one-hot encoding phase. The one-hot encoding is the task of converting the string characters into numerical form.

$$\eta_{\Xi}=\left(\mathrm{M}_{\nabla},\partial om,\mu ot\right)\xrightarrow{encode}\left\|0,1\right\| \tag{14}$$

Lastly, the numeralized data is defined as $\left(\eta_{\Xi}\right)$.

3.8 Semantic representation

Here, the semantic representation is carried out in M_{∇} based on the proposed Dropout ProtBidirectional Encoder Representations from Transformers (D-ProtBERT). The LLM named protBERT significantly learns hierarchical features and patterns from a broad range of proteins. The protBERT provides a more unique amino acid representation for each protein sequence, thereby enhancing the model's reliability. Yet, it had overfitting issues when it dealt with small datasets. To address this issue, the dropout layer is included in the proposed work to prevent dead neurons. Thus, the process involved in the proposed D-ProtBERT is discussed below,

❖ ***Tokenization:***

In tokenization, the amino acids in M_{∇} are initially converted into the token, which holds the unique integer ID (υid). Additionally, the special tokens $(\delta_{\oplus})$ are included at the beginning and end of sequences. Thus, the tokenization (Tok) is done as below,

$$Tok = \mathrm{M}_{\nabla} \xrightarrow{convert} \upsilon id \,\&\, (\delta_{\oplus} + \mathrm{M}_{\nabla})(\mathrm{M}_{\nabla} + \delta_{\oplus}) \tag{15}$$

❖ ***Embedding layer:***

In the embedding layer (ζmd), the token embedding and positional embedding are carried out. The token IDs are converted into dense vectors (∂v) named embedding via the embedding layer. Likewise, the positional embedding (Ξ_{seq}) is added to capture the order of tokens in the sequence.

$$\zeta md = \upsilon id \xrightarrow{convert} \partial v \,\&\, (\Xi_{seq} \oplus Tok) \tag{16}$$

❖ ***Transformer encoder:***

Now, the embedded tokens ζmd are passed through the multiple layers of the transformed encoder. In the transformer encoder, each layer involves the multi-head self-attention mechanisms and feed-forward network. The self-attention mechanism allows each token to focus on various parts of the sequence to understand the features. Likewise, the feed-forward network is used to apply the non-linear transformations to the token representations. Moreover, the residual connections and layer normalization are established to improve the model's performance. Here, the proposed method employs the dropout layer (σ_{drop}) to mitigate the overfitting issues.

$$\sigma_{drop} = \zeta md \bullet \iota en \tag{17}$$

Where, ιen depicts the mask tensor.

❖ ***Representation extraction:***

Finally, the contextual embedding for each token is extracted to capture the semantic and structural information of each amino acid within the sequence.

$$\delta^{\circ}{}_{v} = (\delta^{\circ}{}_{1}, \delta^{\circ}{}_{2}, \ldots\ldots \delta^{\circ}{}_{V}) \tag{18}$$

Where, $v = 1 to V$ denotes the number of semantic features. Thus, the semantic features are implied as (δ°).

3.9 Evolutionary information

Additionally, the evolutionary information is estimated from M_{∇} by analyzing the correlation among each feature. Here, the evolutionary data is identified to improve the performance of the protein structure prediction. The proposed method employs the Pearson correlation coefficient to analyze the correlation among the features in the M_{∇}.

$$\Theta_{PCC} = \frac{\sum_{b=1}^{B}\left(\mathrm{M}_{\nabla_1} - mn\right)\left(\mathrm{M}_{\nabla_2} - mn\right)}{\sqrt{\sum_{b=1}^{B}\left(\mathrm{M}_{\nabla_1} - mn\right)^2 \sum_{b=1}^{B}\left(\mathrm{M}_{\nabla_2} - mn\right)^2}} \tag{19}$$

$$mn = \frac{sum\left(\mathrm{M}_{\nabla_b}\right)}{B} \tag{20}$$

Where, Θ_{PCC} indicates the evolutionary information and mn illustrates the mean value.

3.10 Feature extraction

Also, the features, such as structure ID, experimental technique, residue count, structure molecular weight, and density Matthews are extracted from the M_{∇} to elevate the prediction accuracy. Thus, the extracted features are represented as,

$$\mathrm{I}_j = \left\langle \mathrm{I}_1, \mathrm{I}_2, \ldots\ldots \mathrm{I}_J \right\rangle \tag{21}$$

Where, $i = 1,2,\ldots I$ depicts the number of extracted features I_j.

3.11 Protein function classification

Subsequently, the ∂om, μot, η_{Ξ}, δ°, Θ_{PCC}, and I_j are fed into the proposed LN-PX-RNN classifier, where the protein function is obtained. The RNN effectively captures the temporal dynamics and long-term dependencies to understand the complex patterns of the protein sequences, thus aiding in improving the model's capacity. However, the RNN had vanishing gradient issues due to improper regularization. Additionally, the RNN had dead neurons, which mainly affected the training efficiency. Therefore, the proposed methodology establishes the layer normalization and poly exponential activation function to mitigate the vanishing gradient issue and improve the learning process, respectively. The structure of the proposed LN-PX-RNN is given in Figure 2.

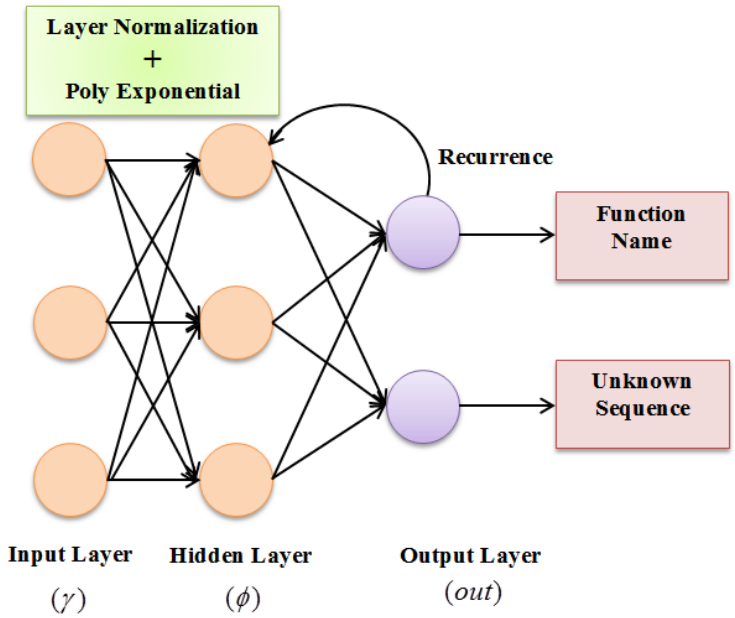

Figure 2: The pictorial illustration of the proposed LN-PX-RNN

- ***Input layer:*** In the first step, the input layer holds the input and is then transferred to the succeeding hidden layers. The inputs (γ) are represented below,

$$\gamma = \left\| \partial om, \mu ot, \eta_{\Xi}, \delta^{\circ}, \Theta_{PCC}, \mathrm{I}_{j} \right\| \tag{22}$$

- ***Poly exponential activation:*** The proposed method employs the poly exponential activation function to upgrade the neuron's learning process, thereby reducing the dead neuron effect. The poly exponential activation function $\left(X_{poly}\right)$ is formulated below,

$$X_{poly}(\gamma) = \left(q\gamma^{2} + p\gamma + s\right)\exp\left(-t\gamma^{2}\right) \tag{23}$$

Here, q, p, s, and t denote the fixed parameters.

- ***Layer normalization:*** Likewise, the proposed work introduces layer normalization to mitigate the vanishing gradient problem. The layer normalization $\left(\mathrm{Y}_{norm}\right)$ is expressed as,

$$\mathrm{Y}_{norm} = \frac{1}{h, \varpi} \sum_{h=1}^{H} \sum_{\varpi=1}^{W} \left(\gamma_{Z,C,h,\varpi} - X_{poly}\right)^{2} \tag{24}$$

Where, h and ϖ depict the height and width of the activation map, respectively, Z denotes the input size, and C indicates the number of channels in the activation function's map.

- ***Hidden layer:*** Thereafter, the hidden layer (φ) is used to identify the protein function by performing the element-wise multiplication. The input is processed in the hidden layer by sharing their weight (wit) and bias (τs).

$$\varphi = X_{poly} \otimes \left(\gamma \cdot wit\right) + \tau s \tag{25}$$

Furthermore, the recurrence function is enabled to handle the sequential data effectively.

- ***Output layer:*** Lastly, the output layer classifies the protein function into function name (Fun) or unknown sequence $\left(\Omega_{seq}\right)$.

$$Out = \left(Fun, \Omega_{seq}\right) \tag{26}$$

The pseudo-code of the proposed LN-PX-RNN is presented further,

---

Input: ∂om, μot, η_{Ξ}, δ°, Θ_{PCC}, and I_{j}

Output: Protein function Out

Begin

Initialize γ, X_{poly}, Y_{norm}, φ and Out

For 1 to each input do,

Assume input layer,

$$\gamma = \left\| \partial om, \mu ot, \eta_{\Xi}, \delta^{\circ}, \Theta_{PCC}, \mathrm{I}_{j} \right\|$$

Apply poly exponential activation,

$$X_{poly}(\gamma) = \left(q\gamma^{2} + p\gamma + s\right)\exp\left(-t\gamma^{2}\right)$$

Execute layer normalization

$$\mathrm{Y}_{norm} = \frac{1}{h, \varpi} \sum_{h=1}^{H} \sum_{\varpi=1}^{W} \left(\gamma_{Z,C,h,\varpi} - X_{poly}\right)^{2}$$

Perform hidden layer,

$$\varphi = X_{poly} \otimes \left(\gamma \cdot wit\right) + \tau s$$

Evaluate output layer

IF (actual == target)

{

Terminate
}
Else
{
Back propagation
}
End IF
End For
Return $Out = \left(Fun, \Omega_{seq}\right)$
End

The above-mentioned phases depict the training stage of the proposed approach. In the same way, the testing sequence is inputted into the proposed model in real-time to identify the protein function. Moreover, to predict the function of the unknown sequence, the proposed work performs the following processes,

3.12 Sequence splitting

Now, the unknown sequence Ω_{seq} is subjected to sequence splitting. Here, the unknown sequence is split into various small manageable parts.

$$\lambda_{\Delta} = \Omega_{seq} \cdot \left\langle \lambda_1, \lambda_2, \ldots \lambda_U \right\rangle \text{ Here, } \Delta = 1 to U \tag{27}$$

Where, U denotes the number of divided sequences. Thus, the divided sequences are mentioned as $\left(\lambda_{\Delta}\right)$.

3.13 Similarity evaluation

Thereafter, the similarity between the λ_{Δ} and remaining sequences in the dataset is evaluated to identify the function of the unknown sequence. The unknown sequence occurs when any of the additional functions are included in the sequence or else any of the functions are excluded from the sequence. Thus, the evaluated similarity is illustrated as $\left(\vartheta\right)$.

$$\vartheta = \left(\tau ax, \nu ar\right) \tag{28}$$

Where, τax denotes the high similarity and νar indicates the low similarity.

3.14 Location and sequence changes

Now, the similarity ϑ helps to identify the variations in the sequences. The obtained variations (low similarity) in the specific location of the sequence are used to identify the pattern of the unknown sequence.

3.15 Pattern recognition

Finally, the νar is inputted to the pattern recognition. Here, the pattern (protein function) is recognized by checking the high similarity value with the remaining sequences in the dataset. The proposed method establishes the EGK-Fuzzy to identify the protein function of the unknown sequences. The fuzzy is chosen because it proficiently handles complex or nuanced scenarios. Yet, it had tuning issues of the membership function. Hence, the proposed method employs the Exponential Gustafson-Kessel (EGK) membership function to improve the model's flexibility. The proposed EGK-Fuzzy is derived further,

- Initially, the crisp data $\left(C_{data}\right)$ is converted into fuzzy data $\left(L_{data}\right)$ in the fuzzification unit $\left(FU\right)$. Here, the low similarity is considered as the crisp data.

$$FU = C_{data} \xrightarrow{convert} L_{data} \tag{29}$$

- Next, the fuzzy if-then rules $\left(\lambda ule\right)$ are generated to identify the protein functions of the unknown sequences.

$$\lambda ule = \{IF\left(operation = ins \,||\, del \,\&\, location\left(a,b\right) \,\&\, \vartheta == high\right), \quad THEN\ \kappa_{\infty} \tag{30}$$

The sequence $\left(\kappa_{\infty}\right)$ is obtained by considering the operation, such as insertion or deletion, location $\left(a,b\right)$, and similarity.

➔ Furthermore, the proposed work employs the EGK membership function to upgrade the fuzzy results.

$$Gs_{kessel}\left(L_{data}\right)_e = \frac{1}{\sum_{y=1}^{Y}\exp\left(\frac{dis^2\left(L_{data}\right)_{e,r}}{dis^2\left(L_{data}\right)_{e,y}}\right)^{\frac{1}{\phi-1}}} \tag{31}$$

$$dis = \sqrt{\sum_{e=1}^{E}\left(L_{data_e} - r\right)^2} \tag{32}$$

Where, ϕ denotes the fuzzifier parameter, $e = 1 to E$ depicts the number of data points, r indicates the cluster centroid, y illustrates the index number, and dis represents the distance parameter.

➔ Also, the decision-making operator $(\mp)$ is used to perform fuzzy operations, which are defined in the fuzzy rules. The decision-making unit (O) is given as,

$$\mathrm{O} = (\text{ƛ}ule)\cdot\mp \tag{33}$$

➔ Similarly, the fuzzy data is transformed into crisp data in the de-fuzzification unit (DU).

$$DU = L_{data} \xrightarrow{transform} C_{data} \tag{34}$$

$$\kappa_{seq} = \left(\kappa_1, \kappa_w, \ldots \kappa_W\right) \tag{35}$$

Here, $w = 1 to W$ depicts the number of recognized sequences κ_{seq} .The pseudo-code of the proposed EGK-Fuzzy is given below,

Input: Low similarity var

Output: Recognized sequence κ_∞

Begin

Initialize C_{data} , L_{data} , Gs_{kessel} and O

For 1 to each var do,

Perform fuzzification

$$FU = C_{data} \xrightarrow{convert} L_{data}$$

Generate fuzzy rules $\text{ƛ}ule$

Apply EGK membership function,

$$Gs_{kessel}\left(L_{data}\right)_e = \frac{1}{\sum_{y=1}^{Y}\exp\left(\frac{dis^2\left(L_{data}\right)_{e,r}}{dis^2\left(L_{data}\right)_{e,y}}\right)^{\frac{1}{\phi-1}}}$$

Execute decision making unit $\mathrm{O} = (\text{ƛ}ule)\cdot\mp$

Transform fuzzy data to crisp data

$$DU = L_{data} \xrightarrow{transform} C_{data}$$

End For

Return sequence κ_∞

End

Thus, the proposed methodology significantly predicts the protein function using the semantic and structural information. Also, the proposed work efficiently recognizes the protein function for unknown protein sequences, improving the disease treatment schemes.

4. RESULTS AND DISCUSSION

In this section, the performance assessment is done to showcase the model's trustworthiness in protein function classification. The proposed work is implemented in the working platform of PYTHON.

4.1 Dataset description

The proposed model is assessed by using the SPS dataset, which is mentioned in the reference section. The SPS dataset comprises crucial information related to proteins and other biological macromolecules. This dataset contains a total of 141402 data. Furthermore, the dataset is split into 80:20 ratios to perform training and testing. Here, 80% of the data is allocated for training the model, and the remaining data is used to validate the system. The specification of the dataset is given in Table 1.

Table 1: SPS dataset characteristics

Training (%)	Testing (%)	Total
113122	28280	141402

Table 1 illustrates the samples of the SPS dataset.

4.2 Performance analysis for protein function classification

To showcase the trustworthiness of the research methodology in protein function classification, the performance assessment is done in this section. The performance of the proposed LN-PX-RNN is validated by comparing it with different kinds of existing algorithms with respect to various quality metrics.

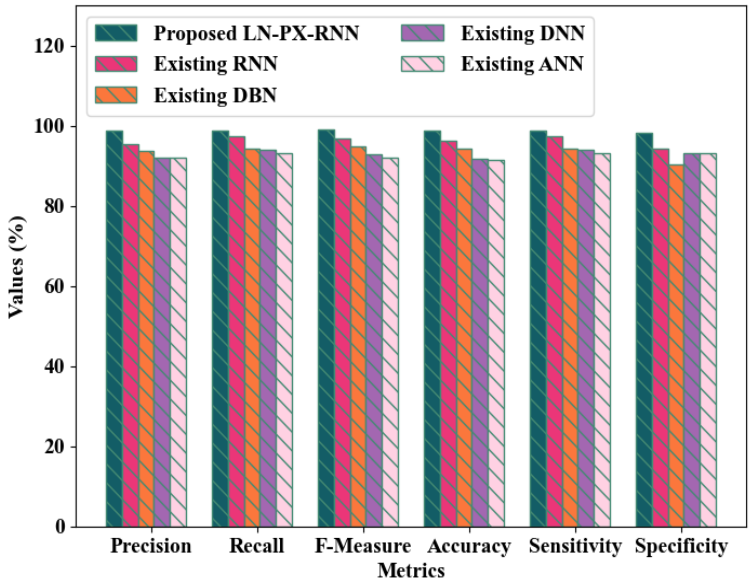

Figure 3: Performance assessment of the proposed LN-PX-RNN

The proposed LN-PX-RNN effectively classifies the protein function due to the assistance of layer normalization and polyexponential activation. Figure 3 exhibits the performance analysis of the proposed LN-PX-RNN and existing algorithms like RNN, Deep Belief Neural network (DBN), Deep Neural Network (DNN), and Artificial Neural Network (ANN) regarding precision, recall, f-measure, accuracy, sensitivity, and specificity. The proposed LN-PX-RNN acquired precision, recall, f-measure, accuracy, sensitivity, and specificity of 98.64%, 98.69%, 99.12%, 98.62%, 98.69%, and 98.19%, respectively. Likewise, the conventional classifiers attained a mean precision, recall, f-measure, accuracy, sensitivity, and specificity of 92.29%, 93.97%, 93.36%, 92.75%, 93.97%, and 91.05%, correspondingly. Thus, the experimental results showed that the proposed approach had higher superiority in protein function classification.

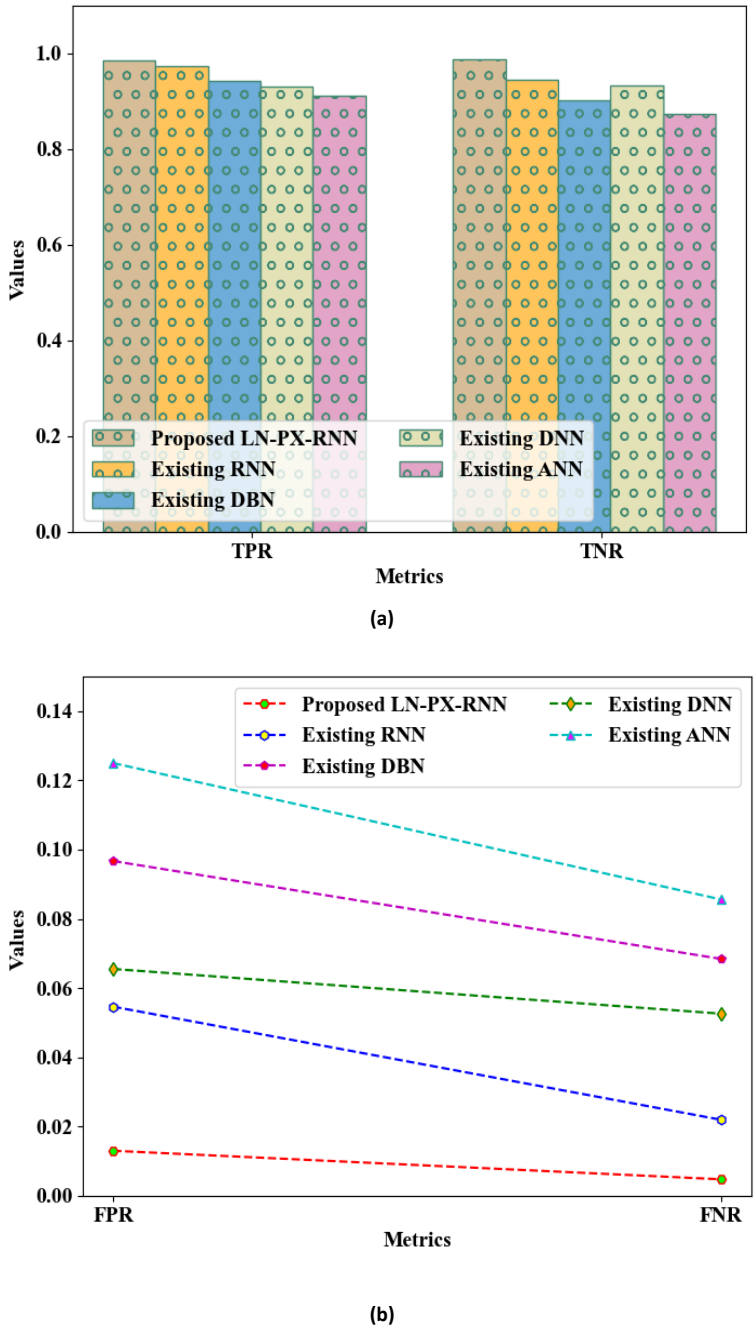

Figure 4: Performance analysis of the proposed LN-PX-RNN with respect to (a) TPR and TNR and (b) FPR and FNR

The performance of the proposed LN-PX-RNN is analyzed in Figure 4 by comparing it with several traditional classifiers regarding True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), and False Negative Rate (FNR). The proposed LN-PX-RNN obtained TPR, TNR, FPR, and FNR of 0.98, 0.98, 0.013, and 0.004, respectively. Yet, the traditional ANN attained TPR, TNR, FPR, and FNR of 0.91, 0.87, 0.125, and 0.085, respectively. Therefore, it is absolutely clear that the proposed method attains better outcomes with fewer errors.

4.3 Performance validation for macromolecular separation

Similarly, the performance of the proposed D(DB)^{2}SCAN is validated to prove the model's efficacy in macromolecular separation. Quality factors like Clustering Time (CT) and Silhouette Score (SS) are used to evaluate the model's reliability.

Table 2: Clustering time validation

Technique	Clustering time (ms)
Proposed D(DB)^{2}SCAN	15358
DBSCAN	22798
FCM	29575
Kmeans	34063
CLARA	37847

Table 3: Silhouette score

Technique	Silhouette score
Proposed D(DB)^{2}SCAN	0.92143
DBSCAN	0.84253
FCM	0.79542
Kmeans	0.74256
CLARA	0.69325

Tables 2 and 3 compare the performance of the proposed D(DB)^{2}SCAN and traditional techniques like DBSCAN, Fuzzy C Means (FCM), Kmeans, and Clustering Large Applications (CLARA) with respect to CC and SS. The proposed D(DB)^{2}SCAN achieved CC and SS of 15358ms and 0.92143, respectively. However, the traditional methods attained limited outcomes owing to the improper epsilon and minimum points. Thus, the proposed work achieved impressive results due to the DDB index.

4.4 Performance assessment for pattern recognition

Here, the performance of the proposed EGK-Fuzzy is evaluated to prove the model's consistency in pattern recognition. Also, the existing techniques, such as Sigmoid Fuzzy (SF), Trapezoidal Fuzzy (TF), Rule-Based Prediction (RBP), and Adaptive Neuro-Fuzzy Inference System (ANFIS) are compared with the proposed model.

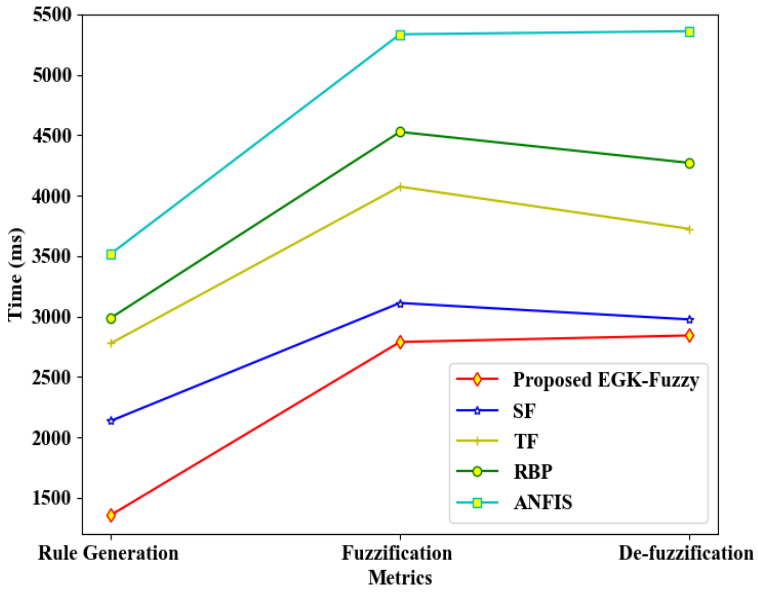

Figure 5: Performance analysis of the proposed EGK-Fuzzy

In Figure 5, the performance of the proposed EGK-Fuzzy and conventional approaches is evaluated based on Rule Generation Time (RGT), Fuzzification Time (FT), and De-fuzzification Time (DT). The proposed EGK-Fuzzy acquired RGT, FT, and DT of 1356ms, 2789ms, and 2843ms, respectively. However, the conventional algorithms attained a mean RGT, FT, and DT of 2855ms, 4262ms, and 4084ms, respectively. Hence, it concludes that the proposed approach obtains low time complexity.

4.5 Performance evaluation for semantic information extraction

Furthermore, the performance assessment of the proposed D-ProtBERT is carried out to reveal the model's prominence in semantic information extraction.

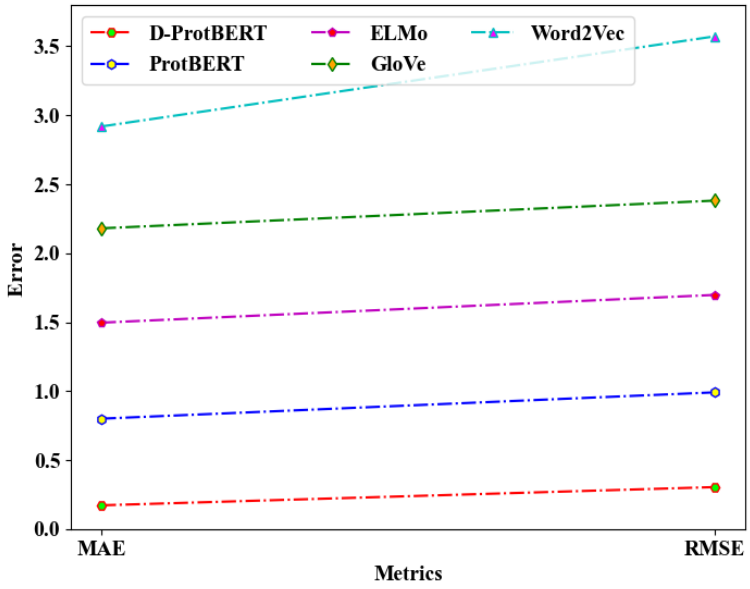

Figure 6: MAE and RMSE analysis

The Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) values of the proposed D-ProtBERT and traditional methods like ProtBERT, Embeddings from Language Models (ELMo), Global Vectors for Word Representation (GloVe), and Word2Vec are analyzed in Figure 6. The proposed work had superior outcomes due to the utilization of the dropout layer. The proposed D-ProtBERT had 0.1735 MAE and 0.3055 RMSE. However, the traditional approaches achieved an average MAE and RMSE of 1.8498 and 2.1611, respectively. Thus, the proposed work attained high robustness and efficacy when compared to the prevailing approaches.

4.6 Comparative validation of the research methodology

In addition, the comparative assessment is done to showcase the reliability and consistency of the framework.

Table 4: Comparative analysis of the proposed methodology

Author's name	Algorithm	Accuracy (%)	Recall (%)	Precision (%)
Proposed model	LN-PX-RNN	98.62	98.69	98.64
(Qian et al., 2022)	Deep neural networks	-	94.2	95.3
(Cui et al., 2022)	CNN-LSTM	-	72.5	83.5
(Giri et al., 2021)	Multi-modal CNN	-	34.39	32.57
(Mohamed Mufassirin et al., 2023)	CNN-LSTM	82.56	-	-
(Gao et al., 2020)	CNN and bidirectional-LSTM	92.84	-	94.36

Table 4 presents the comparative assessment of the proposed work and some relevant frameworks. The proposed LN-PX-RNN proficiently classifies the protein function. Moreover, the proposed method is generalized well enough to recognize the patterns of the unknown protein sequence. The proposed LN-PX-RNN obtained accuracy, recall, and precision of 98.62%, 98.69%, and 98.64%, respectively. The proposed work had better performance owing to the presence of layer normalization and polyexponential activation. Furthermore, the conventional works established techniques like neural networks, CNN, LSMT, and bidirectional-LSTM to perform protein function classification. The existing (Gao et al., 2020) established the combined form of CNN and bidirectional-LSTM to classify the function of the protein. Yet, the traditional (Gao et al., 2020) attained accuracy and precision of 92.84% and 94.36%, respectively. Thus, the comparative assessment stated that the proposed method was more effective and reliable than existing methodologies.

5. CONCLUSION

This paper proposed a well-ordered framework named unknown protein sequence-aware human protein function classification using LN-PX-RNN and EGK-Fuzzy. The proposed LN-PX-RNN was utilized to classify the protein function effectively. Likewise, the proposed EGK-Fuzzy significantly labeled the function of the unknown sequences in the dataset. Furthermore, key processes like macromolecular separation, semantic feature extraction, and domain and motif identification were done to elevate the model's consistency. Moreover, the proposed work was implemented using the SPS dataset. Thus, the experimental findings stated that the proposed model had 98.62% accuracy and 98.64% precision, thus showing the model's trustworthiness. Similarly, the proposed EGK-Fuzzy obtained RGT and FT of 1356ms and 2789ms, respectively, thereby illustrating the low time complexity. For all the quality metrics, the proposed method had superior outcomes than existing methodologies. The proposed work had higher dominance in protein function classification. However, this method only concentrated on the protein sequence dataset.

Future scope:In the future, numerous omics data, including genomics and transcriptomics will be integrated to upgrade the prediction outcomes.

REFERENCES:

Dataset: https://www.kaggle.com/datasets/shahir/protein-data-set?select=pdb_data_no_dups.csv

1. Ahsan, R., Ebrahimi, F., &Ebrahimi, M. (2022). Classification of imbalanced protein sequences with deep-learning approaches; application on influenza A imbalanced virus classes. *Informatics in Medicine Unlocked*, *29*, 1–7. https://doi.org/10.1016/j.imu.2022.100860
2. Cao, M. Y., Zainudin, S., &Daud, K. M. (2024). Protein features fusion using attributed network embedding for predicting protein-protein interaction. *BMC Genomics*, *25*(1), 1–15. https://doi.org/10.1186/s12864-024-10361-8
3. Chauhan, V., Tiwari, A., Joshi, N., &Khandelwal, S. (2022). Multi-label classifier for protein sequence using heuristic-based deep convolution neural network. *Applied Intelligence*, *52*(3), 2820–2837.https://doi.org/10.1007/s10489-021-02529-6
4. Chu, X., Sun, T., Li, Q., Xu, Y., Zhang, Z., Lai, L., & Pei, J. (2022).Prediction of liquid–liquid phase separating proteins using machine learning.*BMC Bioinformatics*, *23*(1), 1–13. https://doi.org/10.1186/s12859-022-04599-w
5. Cui, F., Li, S., Zhang, Z., Sui, M., Cao, C., El-LatifHesham, A., &Zou, Q. (2022). DeepMC-iNABP: Deep learning for multiclass identification and classification of nucleic acid-binding proteins. *Computational and Structural Biotechnology Journal*, *20*, 2020–2028. https://doi.org/10.1016/j.csbj.2022.04.029
6. Cunningham, M., Pins, D., Dezső, Z., Torrent, M., Vasanthakumar, A., &Pandey, A. (2023). PINNED: identifying characteristics of druggable human proteins using an interpretable neural network. *Journal of Cheminformatics*, *15*(1), 1–13. https://doi.org/10.1186/s13321-023-00735-7
7. Dang, T. H., & Vu, T. A. (2024). xCAPT5: protein–protein interaction prediction using deep and wide multi-kernel pooling convolutional neural networks with protein language model. *BMC Bioinformatics*, *25*(1), 1–20. https://doi.org/10.1186/s12859-024-05725-6
8. Dhusia, K., & Wu, Y. (2021). Classification of protein–protein association rates based on biophysical informatics. *BMC bioinformatics*, *22*, 1-20.https://doi.org/10.1186/s12859-021-04323-0
9. Ferruz, N., Heinzinger, M., Akdel, M., Goncearenco, A., Naef, L., &Dallago, C. (2023). From sequence to function through structure: Deep learning for protein design. *Computational and Structural Biotechnology Journal*, *21*, 238–250. https://doi.org/10.1016/j.csbj.2022.11.014
10. Gao, R., Yang, T., Shen, Y., Rong, Y., Ye, K., &Nie, J. (2020). RPI-MCNNBLSTm: BLSTM networks combining with multiple convolutional neural network models to predict RNA-protein interactions using multiple biometric features codes. *IEEE Access*, *8*, 189869–189877. https://doi.org/10.1109/ACCESS.2020.3031301
11. Giri, S. J., Dutta, P., Halani, P., &Saha, S. (2021). MultiPredGO: Deep Multi-Modal Protein Function Prediction by Amalgamating Protein Structure, Sequence, and Interaction Information. *IEEE Journal of Biomedical and Health Informatics*, *25*(5), 1832–1838. https://doi.org/10.1109/JBHI.2020.3022806
12. Hegedűs, T., Geisler, M., Lukács, G. L., &Farkas, B. (2022). Ins and outs of AlphaFold2 transmembrane protein structure predictions. *Cellular and Molecular Life Sciences*, *79*(1), 1–12. https://doi.org/10.1007/s00018-021-04112-1
13. Jang, Y. J., Qin, Q. Q., Huang, S. Y., Peter, A. T. J., Ding, X. M., &Kornmann, B. (2024). Accurate prediction of protein function using statistics-informed graph networks.*Nature Communications*, *15*(1), 1–12. https://doi.org/10.1038/s41467-024-50955-0
14. Jiang, Y., Wang, D., Yao, Y., Eubel, H., Künzler, P., Møller, I. M., &Xu, D. (2021).MULocDeep: A deep-learning framework for protein subcellular and suborganellar localization prediction with residue-level interpretation. *Computational and Structural Biotechnology Journal*, *19*, 4825–4839. https://doi.org/10.1016/j.csbj.2021.08.027
15. Jisna, V. A., &Jayaraj, P. B. (2021). Protein Structure Prediction: Conventional and Deep Learning Perspectives. *Protein Journal*, *40*(4), 522–544. https://doi.org/10.1007/s10930-021-10003-y

16. Khattak, A. U. R., Ullah, A., Rehman, A., Mahmood, T., Khattak, Q. W., Alotaibi, S., &Bahaj, S. A. O. (2023). Robust Deep Neural Network-Based Framework for Predicting and Classifying Capsid Protein Based on Biomedical Data. *IEEE Access*, *11*(September), 107412–107428. https://doi.org/10.1109/ACCESS.2023.3319485
17. Li, X., Han, P., Wang, G., Chen, W., Wang, S., & Song, T. (2022). SDNN-PPI: self-attention with deep neural network effect on protein-protein interaction prediction. *BMC Genomics*, *23*(1), 1–14. https://doi.org/10.1186/s12864-022-08687-2
18. Min, S., Park, S., Kim, S., Choi, H. S., Lee, B., & Yoon, S. (2021). Pre-Training of Deep Bidirectional Protein Sequence Representations with Structural Information.*IEEE Access*, *9*, 123912–123926. https://doi.org/10.1109/ACCESS.2021.3110269
19. Mohamed Mufassirin, M. M., Newton, M. A. H., Rahman, J., &Sattar, A. (2023). Multi-S3P: Protein Secondary Structure Prediction With Specialized Multi-Network and Self-Attention-Based Deep Learning Model. *IEEE Access*, *11*, 57083–57096. https://doi.org/10.1109/ACCESS.2023.3282702
20. Nallasamy, V., &Seshiah, M. (2023). Energy Profile Bayes and Thompson Optimized Convolutional Neural Network protein structure prediction. *Neural Computing and Applications*, *35*(2), 1983–2006. https://doi.org/10.1007/s00521-022-07868-0
21. Pearce, R., & Zhang, Y. (2021). Deep learning techniques have significantly impacted protein structure prediction and protein design. *Current Opinion in Structural Biology*, *68*, 194–207. https://doi.org/10.1016/j.sbi.2021.01.007
22. Qian, Y., Li, X., Zhang, Q., & Zhang, J. (2022). SPP-CPI: Predicting Compound-Protein Interactions Based on Neural Networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *19*(1), 40–47. https://doi.org/10.1109/TCBB.2021.3084397
23. Shin, I., Kang, K., Kim, J., Sel, S., Choi, J., Lee, J. W., Kang, H. Y., & Song, G. (2023).AptaTrans: a deep neural network for predicting aptamer-protein interaction using pretrained encoders. *BMC Bioinformatics*, *24*(1), 1–20. https://doi.org/10.1186/s12859-023-05577-6
24. Soleymani, F., Paquet, E., Viktor, H., Michalowski, W., &Spinello, D. (2022). Protein–protein interaction prediction with deep learning: A comprehensive review. *Computational and Structural Biotechnology Journal*, *20*, 5316–5341. https://doi.org/10.1016/j.csbj.2022.08.070
25. Sunny, S., Prakash, P. B., Gopakumar, G., &Jayaraj, P. B. (2023).DeepBindPPI: Protein–Protein Binding Site Prediction Using Attention Based Graph Convolutional Network. *Protein Journal*, *42*(4), 276–287. https://doi.org/10.1007/s10930-023-10121-9
26. Tasnim, F., Habiba, S. U., Mahmud, T., Nahar, L., Hossain, M. S., &Andersson, K. (2024). Protein Sequence Classification Through Deep Learning and Encoding Strategies. *Procedia Computer Science*, *238*(2019), 876–881. https://doi.org/10.1016/j.procs.2024.06.106
27. Wu, J., Liu, B., Zhang, J., Wang, Z., & Li, J. (2023). DL-PPI: a method on prediction of sequenced protein–protein interaction based on deep learning. *BMC Bioinformatics*, *24*(1), 1–21. https://doi.org/10.1186/s12859-023-05594-5
28. Zeng, X., Meng, F. F., Wen, M. L., Li, S. J., & Li, Y. (2024). GNNGL-PPI: multi-category prediction of protein-protein interactions using graph neural networks based on global graphs and local subgraphs. *BMC Genomics*, *25*(1), 1–13. https://doi.org/10.1186/s12864-024-10299-x
29. Zheng, L., Shi, S., Lu, M., Fang, P., Pan, Z., Zhang, H., Zhou, Z., Zhang, H., Mou, M., Huang, S., Tao, L., Xia, W., Li, H., Zeng, Z., Zhang, S., Chen, Y., Li, Z., & Zhu, F. (2024). AnnoPRO: a strategy for protein function annotation based on multi-scale protein representation and a hybrid deep learning of dual-path encoding. *Genome Biology*, *25*(1), 1–22. https://doi.org/10.1186/s13059-024-03166-1
30. Zhong, W., &Gu, F. (2022). Predicting Local Protein 3D Structures Using Clustering Deep Recurrent Neural Network. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *19*(1), 593–604. https://doi.org/10.1109/TCBB.2020.3005972