CIS 5220 – Final Project – Technical Report

Protein Function Prediction

April 2023

Team Members:

- Siyuan Ding; siyuand4; Email: `siyuand4@seas.upenn.edu`
- Yuzhuo Kang; yuzhuok; Email: `yuzhuok@seas.upenn.edu`
- Ian (Tom) Pan; tompan; Email: `tompan@seas.upenn.edu`

Abstract

Experimental methods for determining protein function are time consuming and costly. The need for high-throughput analysis and the vast amount of data generated by genome sequencing have fueled a growing interest in computational methods for protein function prediction.

This technical report presents a study exploring the use of deep learning techniques to predict protein function using the CAFA, UniProt, and STRING datasets. The study aimed to assess the effectiveness of various approaches, including traditional deep learning and geometric deep learning methods, based on protein sequence and structural information.

We compared these approaches by developing baseline models without structural information and advanced models employing novel techniques like k-mer representation learning, association rules, and interaction network analysis on input features. We also used ensemble models that combined BiLSTM with self-attention mechanisms and CNN, and experimented with embeddings to determine their potential for improving prediction accuracy. Model performance was evaluated using standard metrics such as accuracy, precision, recall, F1-score, and hamming loss. The results showed that BiLSTM with attention mechanisms, ensemble models stacking CNN and BiLSTM, and GNNs (GAT, GCN) outperformed baseline models and achieved higher accuracy than those reported in current papers.

In summary, this report emphasizes the potential of computational methods, especially those encoding the 3D structure of proteins in graphs, for predicting protein function. These methods can complement experimental approaches and have significant implications for drug discovery and personalized medicine.

1 Introduction

Proteins play crucial roles in the cell by adopting diverse 3-dimensional structures to perform specific functions. Although some functional regions of proteins are unstructured, the majority of protein domains have ordered and specific conformations. These structural features of proteins are essential in defining their diverse functions, including catalyzing biochemical reactions, transport, signal transduction, and providing mechanical stability. Several classification schemes have been used to categorize these functions, such as the Gene Ontology (GO) Consortium, Enzyme Commission (EC) numbers etc. Specifically for this project we formed our target problem towards predicting GO ontology terms, which provides a hierarchical organization of protein functions into three ontologies: Molecular Function (MF), Biological Process (BP), and Cellular Component (CC), which describe different aspects of protein functions.

Before the advent of deep learning, traditional machine learning classifiers were widely used for automatic prediction of protein function. These included support vector machines, random forests, and logistic regression, which demonstrated that integrative prediction schemes outperformed homology-based function transfer. Additionally, integration of multiple gene and protein network features typically yielded better results than sequence-based features (Guan et al. 2008). However, in the past decade, deep learning has led to significant advancements in performance across a wide range of problems, including protein function prediction and structure determination. Deep learning models can automatically extract high-level features from raw data and provide end-to-end predictions (Kulmanov et al. 2017). For example, deep learning has been used to learn protein sequence embeddings for feature engineering, and has shown great promise for predicting protein structure and function.

In this project, we aim to delve deeper into the subject matter by employing geometric deep learning methods. Traditional deep learning approaches have demonstrated the importance of structural features, but they fall short in fully exploiting the potential of 3D protein structure representations. To address this, we have incorporated various graph neural networks (GNNs) and self-attention mechanisms to encode the 3D representations of protein structures. Additionally, we have implemented several innovative strategies such as association rules learning, k-mer representation learning, and protein interaction networks to evaluate whether our model's performance can be further enhanced in comparison to existing models. Our results indicate a significant improvement in both prediction accuracy and F1-score relative to the baseline deep learning models. Moreover, GNNs demonstrate strong model performance, as evidenced by the high F1-score and low hamming loss.

2 Related Work

(Vu et al. 2021) predict protein functions using multi-layer CNN, MLP, and LSTM models. Protein sequences are represented using a one-hot encoding

scheme. This representation yields a total of 6,000 features. Reported F1-score and accuracy are around 0.6. We aim to improve these metrics by employing more advanced deep learning models, including deeply bidirectional fusion LSTM, AE-MLP hybrid model, and ensemble modeling. Furthermore, rather than using one-hot encoding, we implement k-mer representation learning based on motif patterns and leverage learned embeddings to capture local motifs relevant to protein functions.

In their research, Zhang, Song, Zeng, Wu, et al. 2020 and Zhang, Song, Zeng, Li, et al. 2019 employ protein interaction network features to infer GO annotations and predict protein functions. These models demonstrate substantial advancements compared to traditional approaches that depend exclusively on sequence information. Our study seeks to build upon this methodology by incorporating multiple functional partners rather than a single interaction network and capturing long-range dependencies between proteins.

(Rives et al. 2021) employs the Transformer model to achieve strong performance on protein-related tasks by capturing physicochemical relationships among amino acids, which provides valuable insights into the behavior of individual amino acids. We build upon this approach by applying association rules to physicochemical properties to expand the feature set. Our goal is to capture higher-order interactions and dependencies between amino acids, which is important in determining protein function, and interactions with other molecules.

(Jha et al. 2022) employs graph convolutional networks (GCNs) to predict protein interactions. The authors represent protein sequences as graphs and learn meaningful features within the graph representations. However, the application of graph deep learning to functional prediction remains relatively unexplored, highlighting the novelty and potential of this approach. We apply GCNs to predict protein function by leveraging the graph structure of protein sequences and facilitating multi-label classification. Another novel aspect of our study is the incorporation of protein interactions into the feature set. Deep learning models utilize large-scale interaction data to generate more accurate predictions.

3 Dataset and Features

3.1 Dataset

The Critical Assessment of Function Annotation (CAFA) dataset is widely used for deep learning-based protein function prediction. It comprises 386,735 protein sequences and 24,266 unique GO classes, which describe molecular function, biological process, and cellular component. Additionally, the dataset includes amino acid composition and sequence length.

To integrate protein interaction networks, we use the UniProt database, which contains information on over 200 million protein sequences from various organisms, including functional, structural, and interaction annotations. STRING is a database that offers protein interaction information, functional

associations, and networks. UniProt provides access to each protein's functional partners in STRING. We then extract these partners using the protein IDs in the CAFA dataset.

Due to computational limitations, we focus on homologous proteins as they are more likely to be consistent across databases and resources. We also select the 100 most frequent unique GO labels in our dataset. After addressing class imbalance, our dataset consists of 92135 protein sequences and 100 unique GO classes. Moreover, we subset our data to proteins with 300 or fewer amino acids, to help reduce training and inference times.

3.2 Features

Our dataset contains protein IDs and sequences, along with their corresponding GO labels and functional partners. To enhance our feature set for multi-label prediction, we will utilize this information to explore a broader range of attributes. These expanded features will potentially include interaction-based characteristics and sequence-derived properties. By incorporating these additional features, we aim to improve the accuracy and comprehensiveness of our protein function predictions.

3.2.1 Protein-Protein interactions

The STRING is a comprehensive database of known and predicted protein interactions. For each protein, the top five functional partners with the highest interaction scores were retained. A total of 75 unique functional partners were identified. These functional partners were transformed into features utilizing binary adjacency matrix (El-Lakkani et al. 2013). In a GCN, the adjacency matrix is used to define the graph structure and is multiplied with the node features to compute the graph convolutions. In an attention-based model, it can be used to compute the attention weights between nodes to better capture the global relationships between the input sequences.

3.2.2 Physicochemical properties

Physicochemical properties, including molecular weight, aromaticity, instability index, isoelectric point, and gravy, provide a quantitative characterization of protein sequences. These properties capture important structural and functional characteristics. Our study augments the feature space using an innovative approach that incorporate association rules with the physicochemical properties. Association rules identify relevant combinations of physicochemical properties that are strongly associated with the protein function. It can potentially improve performance by capturing higher-order interactions among the features. The Apriori algorithm is employed to uncover significant relationships among the diverse physicochemical properties of proteins, subsequently selecting pairs of properties that meet the specified support and confidence thresholds.

3.2.3 K-mer Representation Learning

An innovative technique, K-mer representation learning, is employed to analyze protein sequences by decomposing them into smaller, overlapping subsequences of length k. Subsequently, the frequency of each k-mer is computed, generating a frequency vector that serves as a feature representation. Both dipeptide motifs, encompassing all motif bases of length two, and 100 significant tri-peptide motifs are incorporated as features. In total, K-mer counting yields 500 distinct features. K-mer counting captures local sequence information, as it considers short contiguous subsequences, which provides insights into the presence or absence of certain sequence biologically relevantmotifs or patterns.

4 Methodology

4.1 CNN

CNNs employ convolutional layers with small filters that can identify and capture crucial sequence motifs responsible for protein function. CNNs also exhibit translation invariance, meaning that they can recognize patterns regardless of their position within the sequence. CNNs enables the model to capture various levels of information pertinent to protein function.

4.2 GRU Variant of RNN

We opted for the GRU variant of RNNs due to its ability to address the vanishing gradient problem, which often hinders the training of conventional RNNs on long sequences(Le et al. 2019). GRUs use gating mechanisms that allow for more effective propagation of information in sequences through time,

K-mer representation and learned embeddings of amino acid and learned embeddings were fed into the GRU model, which captured the sequential dependencies among amino acids(Xia et al. 2022). This vector representation was subsequently passed through a fully connected layer to obtain the probabilities of the protein belonging to each functional class.

4.3 Autoencoder-MLP Hybrid Model

We propose a hybrid model that integrates an autoencoder with a MLP in predicting protein functions. The autoencoder component serves captures the intrinsic structure within the sequences and reduces the input dimension. The encoder phase compresses the input data, while the decoder phase reconstructs compressed data representation (Ahmadlou et al. 2021). As shown in Figure, the extracted features are fed into the MLP, which leverages its hidden layers to learn non-linear relationships between the features and the target protein functions.

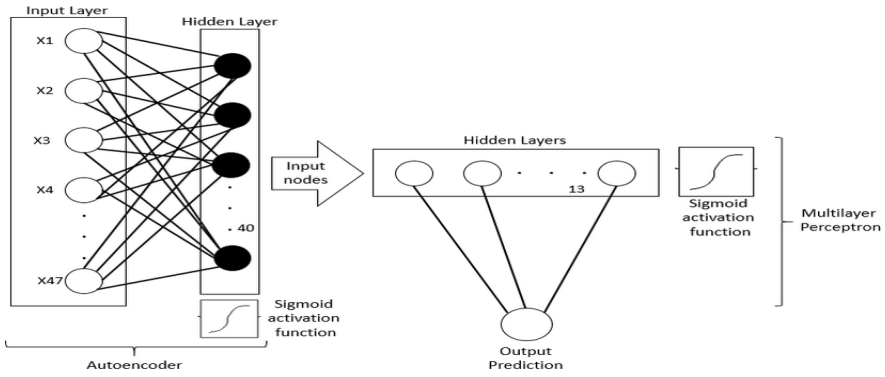


Figure 1: Autoencoder-MLP Hybrid Model Structure

4.4 Graph Convolutional Networks for Sequence-based Protein Function Prediction

GCNs enhance conventional neural networks by integrating graph-based data structures, which capture information inherent in graph representations. Each amino acid is considered as a node, and edges are created based on their sequential proximity. GCNs employ a series of graph convolutional layers to update node features by aggregating information from their neighbors (Zaki et al. 2021; Zamora-Resendiz et al. 2019). This message-passing mechanism learns meaningful node embeddings that capture the functional properties of the sequences. The final node embeddings are then aggregated using pooling operations to obtain a graph-level representation as shown in Figure, which is the input to a multi-label classification layer.

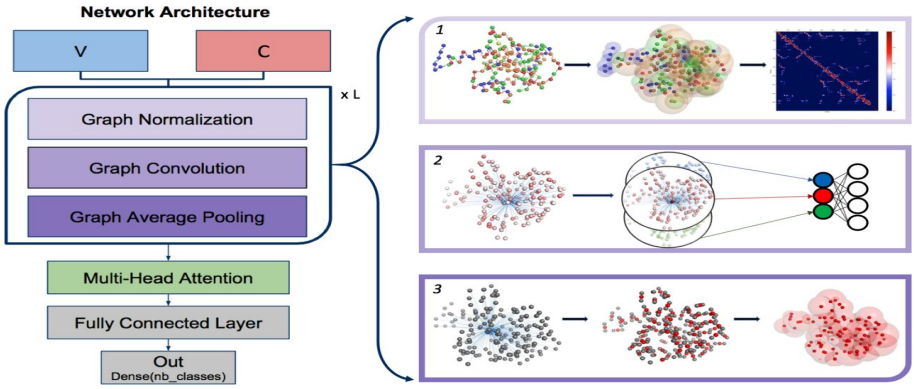


Figure 2: Graph Convolutional Network Structure

4.5 GAT

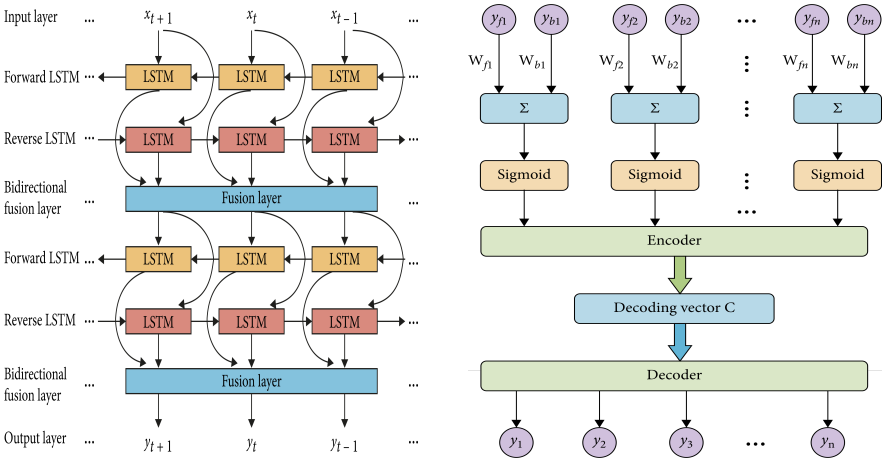
GAT, or Graph Attention Network, is a type of GNN that utilizes attention mechanisms to process graph-structured data more effectively. The attention mechanisms allow the model to weigh the importance of different neighboring nodes when aggregating information from a node's neighborhood, which helps the model learn better representations of the graph.

4.6 Bidirectional LSTM

In a bidirectional LSTM, two separate LSTM layers are used, one processing the sequence in the forward direction and the other in the backward direction. The hidden states of both LSTM layers are concatenated to produce the final output. Bidirectional LSTM can capture the long-range dependencies between the residues in the protein sequence, which can improve the prediction accuracy.

4.6.1 Structure of the deeply bidirectional fusion LSTM

The Bidirectional Deep Fusion LSTM employs a fusion layer to merge forward and reverse data within the model's hidden layer, resulting in a neural network structure with enhanced feature representation. As shown in Figure 3, the input of is derived from the upper bidirectional LSTM neuron's output, while the output serves as the input for the lower LSTM neuron. Upon entering the fusion layer, fusion weights are assigned to the forward and reverse neuron output data (Zheng et al. 2021). The fused vector yields vector data of a specified dimension through the Encoder-Decoder framework.



(a) Bidirectional LSTM structure (b) fusion layer structure

Figure 3: Bidirectional Deep Fusion LSTM structure

4.6.2 Bidirectional Fusion LSTM with Attention Mechanism

The attention mechanism is integrated into a BiLSTM by introducing an additional layer that dynamically computes the relevance of each input hidden state in relation to the current decoding time step as shown in Figure 4. This layer enhances the model's ability to focus on specific parts of the input sequence(Wang et al. 2018). The context vector captures the most relevant information from the input sequence, which is subsequently combined with the decoder's hidden state, yielding the attention mechanism output.

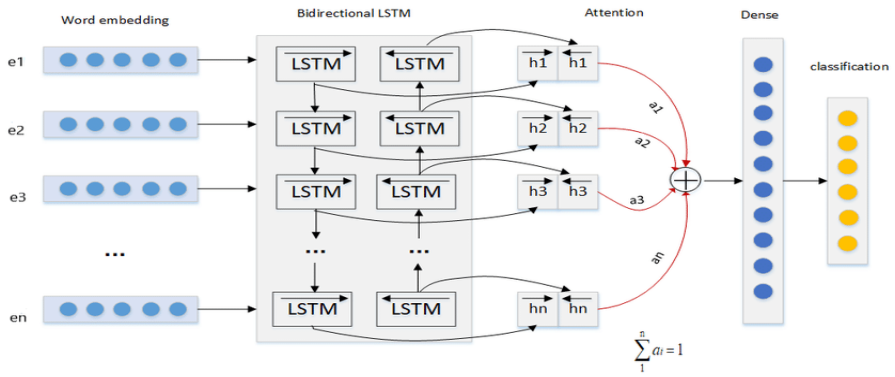


Figure 4: Attention-Based Bidirectional LSTM structure

$$\text{score}(h_j, s_{i-1}) = h_j^T W_a s_{i-1}$$

$$\alpha_{ij} = \frac{\exp(\text{score}(h_j, s_{i-1}))}{\sum_{k=1}^{T_x} \exp(\text{score}(h_k, s_{i-1}))}$$

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$

$$\hat{h}_i = \tanh(W_c[c_i; s_{i-1}])$$

The hidden state of the decoder LSTM at time step $i-1$ is s_{i-1}. The attention weight for the j-th input at the i-th time step in the decoder is α_{ij}. At time step i in the decoder, the output of the attention mechanism is given by $\hat{h}_i$. The tanh function imparts non-linearity to the attention layer, which captures the non-linear relationships between protein sequences and target functions.

4.7 Ensemble Learning

Stacking is an ensemble learning technique that enhances the prediction capability by integrating the outputs of multiple base models as input features for a aggregated higher-level model (Whalen et al. 2016). This hierarchical

structure allows the ensemble to capitalize on the strengths of diverse base models while minimizing their weaknesses. In multi-label classification, stacking capture diverse patterns in the data, making it more robust to overfitting.

We utilized two ensemble models in our approach, comprising a combination of a CNN and a Transformer, as well as a combination of CNNs and a BiLSTM. Ensemble can benefit from the strengths of both models. Since convolutional layers apply filters to local regions of the inputs, CNNs learn lower-level features like local motifs, which can be fed into the BiLSTMs to learn higher-level, context-dependent features. Transformers capture long-range relationships within sequences through self-attention mechanisms.Stacking these architectures enables the ensemble to learn from both patterns and long-range dependencies and predict protein functions more accurately.

4.8 Hyperparameter Tuning and Regularization Technique

In our deep learning models, we applied various regularization techniques such as L1 and L2 regularization, dropout, early stopping, and batch normalization. L1 regularization encouraged sparsity in weight matrices, minimizing the influence of irrelevant features, while L2 regularization fostered smaller weights for better generalization across multiple labels. Early stopping and batch normalization during training ensured generalization to unseen data. These techniques promoted sparse model coefficients, aiding in identifying crucial features and reducing noise from irrelevant ones, while also preventing overfitting to the training data.

We experimented with various hyperparameter values to enhance model performance. This involved testing different batch sizes, learning rates, number of epochs, and dropout rates to identify the best-performing model. We also compared performance using different optimizers (Adam, SGD, Adagrad) and loss functions (MSE, binary cross-entropy, categorical cross-entropy). Additionally, we explored the impact of adjusting batch normalization layer configuration, neuron and layer counts, and kernel size on prediction accuracy.

In the GCN model, we employed spectral regularization to constrain graph convolutional filters, ensuring a smooth spectral response and improved model generalization. Additionally, we implemented graph normalization techniques, such as symmetric and row normalization, to enable the model to learn effectively from interaction networks.

By manually testing different hyperparameter combinations, we determined an optimal set that yielded the best results. During the tuning process, we utilized a validation set to reduce overfitting. We evaluated model performance using various metrics, preferring F1-score over accuracy in multi-label classification tasks, as it accounts for both precision and recall in imbalanced datasets.

5 Results

In our study, the objective is to establish that the utilization of sophisticated deep learning models, in conjunction with innovative methodologies, can substantially augment the prediction accuracy of protein function. The proposed approach encompasses an array of tactics, including k-mer representation learning, association rules, fusion layers, attention mechanisms, and graph neural networks. This comprehensive combination is anticipated to yield efficacy in the context of multi-label classification tasks related to protein function prediction.

5.1 Traditional Deep Learning Models

We conduct a thorough evaluation of various traditional deep learning models, employing four essential evaluation metrics: accuracy, recall, precision, and F1-score. Accuracy reflects the model's ability to correctly identify a substantial proportion of protein functional categories. The F1-score holds greater importance than accuracy in multi-label prediction due to its capacity to balance precision and recall, offering a comprehensive performance assessment.

In the presented figures, the performance of advanced models generally surpasses that of the baseline models. In the absence of ensemble learning, the deep bidirectional fusion LSTM achieves the highest accuracy, precision, recall, and F1-score. Incorporating attention mechanisms can enhance accuracy by 6 percent and F1-score by 4 percent. The hybrid AE-MLP model exhibits the lowest accuracy among advanced models, with performance comparable to the baseline models. Among all baseline models, the CNN demonstrates the best performance.

Regarding the ensemble model, when the CNN and BiLSTM are combined, the resulting model outperforms all others, although its F1-score remains close to that of the deep bidirectional fusion LSTM. Interestingly, when the CNN and transformer are stacked together, there is no observable improvement in overall performance, with results resembling those of the standalone CNN.

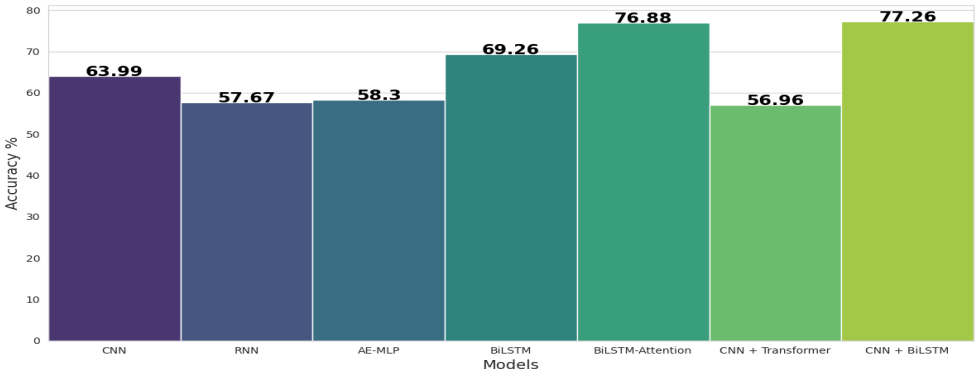


Figure 5: Accuracy Plot

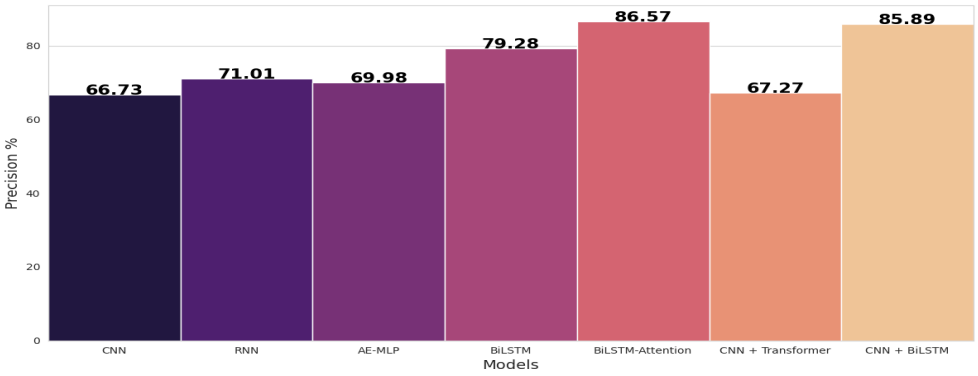


Figure 6: Precision Plot

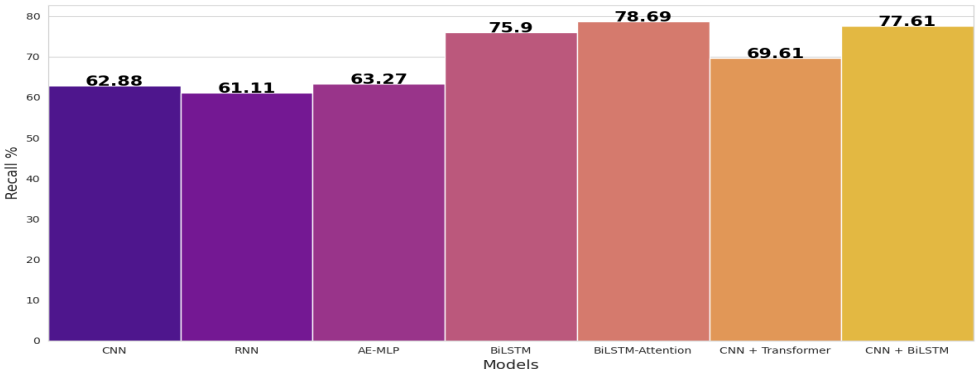


Figure 7: Recall plot

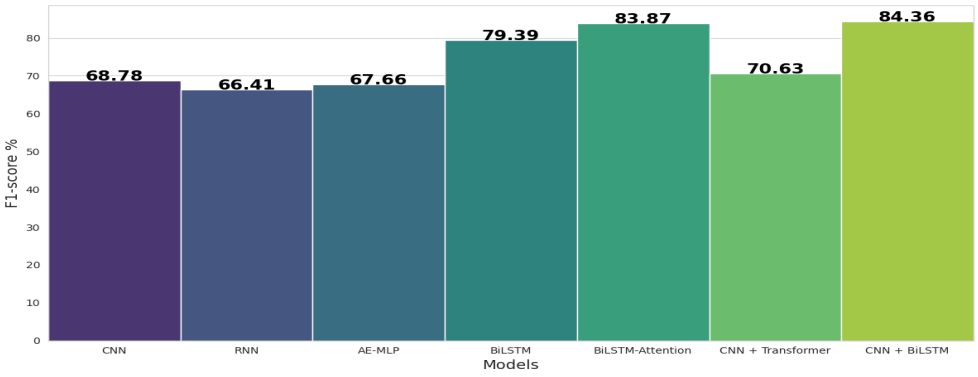
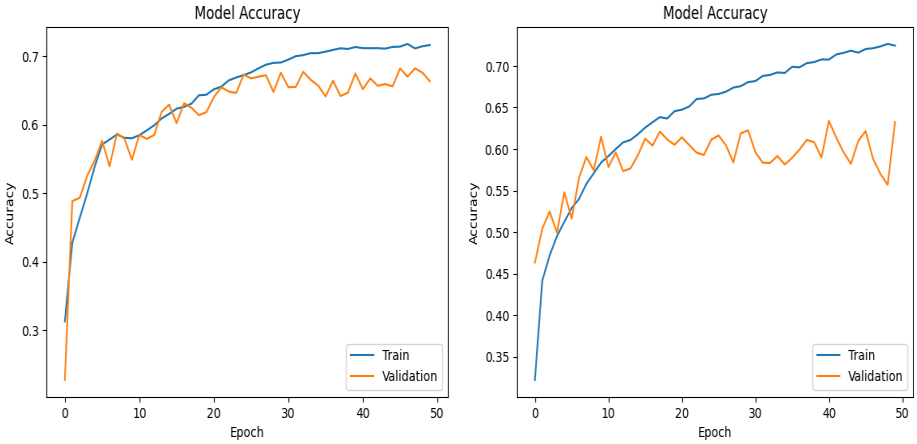


Figure 8: F1-score plot

5.1.1 BiLSTM result comparison



(a) Bidirectional LSTM learning curve (b) fusion layer structure

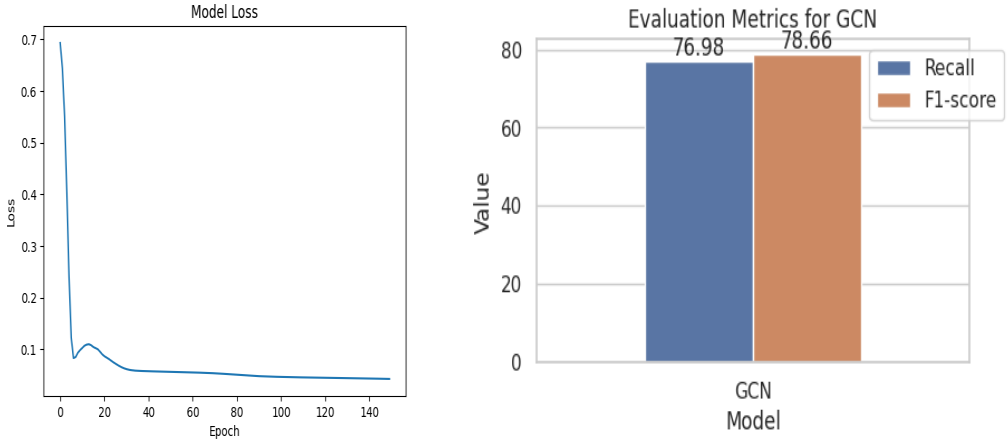
Figure 9: Bidirectional LSTM with attention mechanism learning curve

Bidirectional LSTM with attention mechanism up-and-down learning curve, while the model without attention is more flat. The model with attention is more complex and has more parameters than the one without attention, which makes it more susceptible to overfitting. The model without attention may have already reached its maximum potential and is unable to learn much from additional data, resulting in a flat learning curve.

5.2 Graph Deep Learning Models

5.2.1 GCN result

The GCNs demonstrate strong performance in this multi-label task. As the training process progresses, the loss curve stabilizes significantly, suggesting convergence and effective learning of underlying patterns within the data. Notably, the model's recall and F1-score reach 76.98% and 78.66% as shown in Figure, respectively, outperforming all baseline models. In multi-label classification, accuracy and precision may not be the most suitable evaluation metrics, as they do not consider the proportion of correct positive predictions relative to the total positive predictions.



(a) GCN learning curve

(b) GCN evaluation metric

Figure 10: Graph Convolutional Network Performance

5.2.2 GAT result

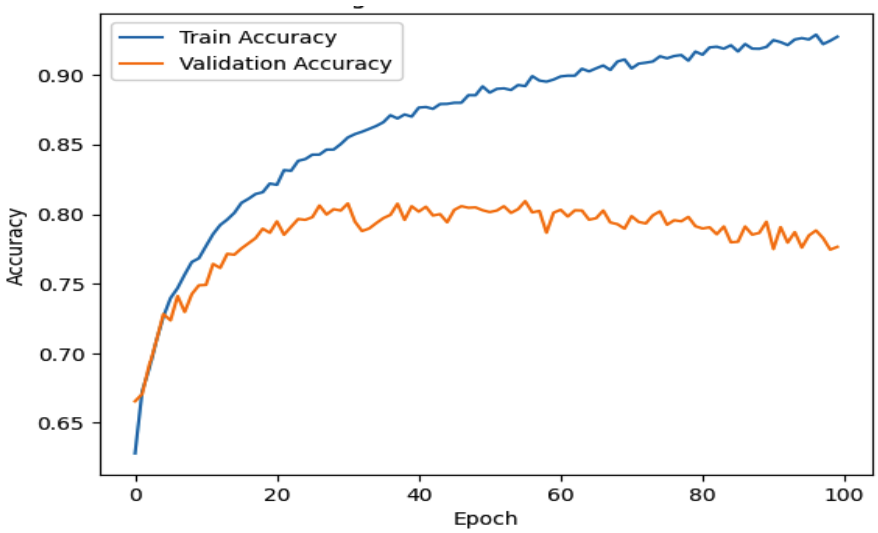


Figure 11: Accuracy Plot

GAT model has a validation accuracy of 0.8134. There are signs of overfitting, however, it should be noted we only trained the model on roughly a quarter of the training data to ease the computational burden. Training on more data would likely help improve generalizability.

5.3 Loss function choice and new evaluation metric

For multi-label classification task, we minimized the binary cross-entropy loss function, which calculates the loss for each label independently and combines the results. This approach is effective for handling imbalanced multi-label dataset

To evaluate the performance of our models, we also used the Hamming loss, which measures the average fraction of incorrect label predictions across all instances. The results of the Hamming loss align with the previously reported results, where the Bidirectional Fusion LSTM model demonstrated the best performance, with a loss value of 0.262. Additionally, the GCN model also performed well in our evaluation.

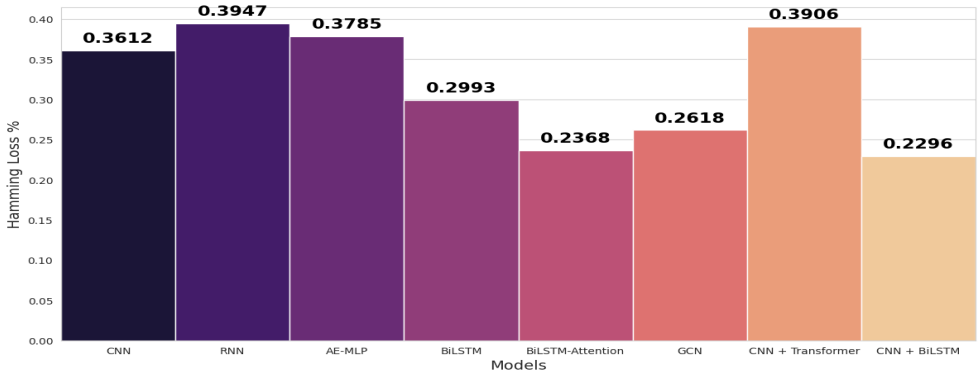


Figure 12: Hamming Loss plot

6 Discussion

We utilized conventional and graph-based deep learning models for protein function prediction. To achieve this, we employed a range of techniques, such as embedding, position-specific scoring matrices, k-mer representation learning, and association rules. We trained several deep learning models, including MLP, CNNs, RNNs, AE, BiLSTM with attention mechanism, GNNs, and ensemble modeling to accurately classify proteins into their functional categories. To evaluate the performance of these models, we used a variety of metrics, including accuracy, precision, recall, F1 score, and hamming loss, and compared our results to existing methods in the literature.

6.1 Findings

Advanced models, such as BiLSTMs with attention mechanisms and ensemble models that combine CNNs and BiLSTMs, outperform the baseline models in terms of accuracy, precision, and F1-score, achieving an average improvement of 10% to 15%. BiLSTMs possess a unique memory cell structure that enables them to store and retrieve information over extended periods, effectively mitigating the vanishing gradient problem often encountered in RNNs. Although CNNs can also handle variable-length inputs, they typically require more complex architectures or additional pre-processing steps compared to BiLSTMs. Ensemble methods adeptly capture both spatial and temporal features, resulting in superior overall performance. By averaging the predictions of individual models, ensemble methods reduce the influence of biases or inaccuracies inherent in each model, yielding more stable and dependable results.

Compared to another advanced model, the performance of AE-MLP and ensemble methods that combine CNNs and Transformers is relatively poor. One major limitation of MLPs is their lack of memory or recurrence, which restricts their ability to capture long-range dependencies between amino acids in protein

sequences. This can be a critical factor for accurately predicting protein functions, as distant amino acids often play an important role in determining the overall function. Additionally, the success of an ensemble model depends on the diversity of the individual models. If the CNN and Transformer models learn similar features or representations of the data, the ensemble may not benefit from their combination, leading to suboptimal performance.

GNNs also exhibit good performance whose hamming loss is 0.1% lower than the baseline models. The reason is that the graph of representation of proteins can include various types of nodes and edges, with associated features. It leverages the structure of the protein by taking in information from the contact map. The contact map representation of the structure gives information on which amino acids are close to each other. Though the GAT is restricted by the train data size, it shows the highest validation accuracy among all the models.

6.2 Limitations and Ethical Considerations

Our models exhibit several limitations. BiLSTMs with attention mechanisms can be computationally demanding in terms of memory and time complexity. This may restrict their scalability for extensive protein function prediction tasks or hinder their adoption with limited computational resources. Although attention mechanisms enable the model to concentrate on relevant portions of the protein sequence, deciphering the attention weights for biological insights remains challenging. Moreover, ensembles of deep learning models complicate the extraction of biological insights or the understanding of relationships between input features and protein function predictions. Similarly, GNNs face scalability challenges when applied to large protein networks or structures with numerous nodes and edges. These challenges result in increased computational complexity, greater memory requirements, and extended training durations.

Our models might face difficulties generalizing findings to unseen protein sequences from different organisms, particularly when these sequences deviate significantly from the training data. This may lead to diminished prediction accuracy for novel proteins that are underrepresented in the training data. Additionally, the model's accuracy is directly linked to the quality of the training data. Incomplete protein function annotations in the training data could constrain the model's predictive capabilities.

The model could potentially be misused for generating misleading or incorrect protein function predictions, which might negatively impact downstream applications such as drug development or the understanding of biological processes. Adversaries could exploit this vulnerability by manipulating input protein sequences, causing the model to yield deceptive predictions, and ultimately undermining the model's effectiveness in real-world applications.

6.3 Future Research Directions

With additional time and computational power, we will undertake several steps to optimize our model's performance. First, we will experiment with vari-

ous model architectures, such as Transformer-based models or capsule networks, to assess their suitability for protein function prediction tasks. We will also evaluate the model's performance across diverse organisms and protein families to identify and address potential biases, ensuring that the model performs well for different subgroups. Furthermore, we plan to implement data augmentation and advanced regularization techniques to prevent overfitting and enhance the model's generalization performance on new, unseen protein sequences.

Additionally, we aim to explore more scalable GNN architectures and training techniques to improve computational efficiency, enabling the application to large-scale protein datasets. We will also work on increasing the model's security against adversarial attacks to ensure its utility in real-world applications. The general principles and approaches used in our study could be valuable to researchers in the field of bioinformatics.

7 Conclusions

In conclusion, this study underscores the efficacy of deep learning techniques, especially those that integrate protein sequence and structural information, in predicting protein function. Through a comparative analysis of different approaches and the implementation of cutting-edge techniques, we have demonstrated that advanced models, such as BiLSTM with attention mechanisms, ensemble models combining CNN and BiLSTM, and GNNs, surpass baseline models and attain greater accuracy than findings in previous studies.

The accomplishments of these computational approaches, specifically those that represent 3D protein structures in graph form, emphasize their potential to act as crucial adjuncts to experimental methods within the realm of protein function prediction. Consequently, the adoption of these methodologies can pave the way for remarkable progress in areas such as drug discovery and personalized medicine, establishing them as indispensable resources for life science researchers and professionals alike.

References

Ahmadlou, M. et al. (2021). "Flood susceptibility mapping and assessment using a novel deep learning model combining multilayer perceptron and autoencoder neural networks". In: *Journal of Flood Risk Management* 14.1, e12683.

Guan, Yuanfang et al. (June 2008). *Predicting gene function in a hierarchical context with an ensemble of classifiers - genome biology.* URL: `https://doi.org/10.1186/gb-2008-9-s1-s3`.

Jha, Kanchan, Sriparna Saha, and Hiteshi Singh (2022). "Prediction of protein–protein interaction using graph neural networks". In: *Scientific Reports* 12.1, p. 8360.

Kulmanov, Maxat, Mohammed Asif Khan, and Robert Hoehndorf (Oct. 2017). "DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier". In: *Bioinformatics* 34.4, pp. 660–668. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btx624. eprint: https://academic.oup.com/bioinformatics/article-pdf/34/4/660/48914540/bioinformatics_34_4_660.pdf. URL: https://doi.org/10.1093/bioinformatics/btx624.

El-Lakkani, A. and S. El-Sherif (2013). "Similarity analysis of protein sequences based on 2D and 3D amino acid adjacency matrices". In: *Chemical Physics Letters* 590, pp. 192–195.

Le, Nguyen Quoc Khanh et al. (2019). "Computational identification of vesicular transport proteins from sequences using deep gated recurrent units architecture". In: *Computational and Structural Biotechnology Journal* 17, pp. 1245–1254.

Rives, Alexander et al. (2021). "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences". In: *Proceedings of the National Academy of Sciences* 118.15, e2016239118.

Vu, Thi Thuy Duong and Jaehee Jung (2021). "Protein function prediction with gene ontology: from traditional to deep learning models". In: *PeerJ* 9, e12019.

Wang, Y. et al. (2018). "BiLSTM-Attention and LSTM-Attention via Soft Voting in Emotion Classification". In: *IEEE Access* 6, pp. 42431–42439.

Whalen, S., O. P. Pandey, and G. Pandey (2016). "Predicting protein function and other biomedical characteristics with heterogeneous ensembles". In: *Methods* 93, pp. 92–102.

Xia, Weiqi et al. (2022). "PFmulDL: a novel strategy enabling multi-class and multi-label protein function annotation by integrating diverse deep learning methods". In: *Computers in Biology and Medicine* 145, p. 105465.

Zaki, N., H. Singh, and E. A. Mohamed (2021). "Identifying protein complexes in protein-protein interaction data using graph convolutional network". In: *IEEE Access* 9, pp. 123717–123726.

Zamora-Resendiz, R. and S. Crivelli (2019). "Structural learning of proteins using graph convolutional neural networks". In: *BioRxiv*, p. 610444.

Zhang, Fuhao, Hong Song, Min Zeng, Yaohang Li, et al. (2019). "DeepFunc: a deep learning framework for accurate prediction of protein functions from protein sequences and interactions". In: *Proteomics* 19.12, p. 1900019.

Zhang, Fuhao, Hong Song, Min Zeng, Fang-Xiang Wu, et al. (2020). "A deep learning framework for gene ontology annotations with sequence-and network-based information". In: *IEEE/ACM transactions on computational biology and bioinformatics* 18.6, pp. 2208–2217.

Zheng, T. et al. (2021). "The Bidirectional Information Fusion Using an Improved LSTM Model". In: *Mobile Information Systems* 2021, pp. 1–15.