DeepNeuroPred: Advancing Neuropeptide Prediction.

Sarthak Jindal, D Veera Harsha Vardhan Reddy, Avi Vyas, Godavarthi Sai Nikhil

Abstract—**Neuropeptides, a class of small biologically active molecules found in the nervous systems of all animals, including humans, played crucial roles in cell-to-cell communication and regulated various physiological processes and behaviors. The NeuroPred-PLM model emerged as a promising tool for neuropeptide prediction, combining convolutional neural networks (CNNs) with global multi-head attention mechanisms to enhance the semantic representation of protein sequences. However, there remained room for improvement in both prediction accuracy and model interpretability, which were crucial for advancing peptide research and accelerating drug discovery efforts. The optimization of the NeuroPred-PLM model involved leveraging a larger and updated dataset to improve the accuracy of neuropeptide predictions. The EsmModel architecture was fine-tuned using transfer learning techniques, multi-scale CNN layers were employed for local feature extraction, and global multi-head attention was utilized to capture sequence-wise dependencies. Additionally, several innovations over existing methods were introduced, including the development of a new model, DeepProtein, enhanced data preprocessing and visualization techniques, and refined regularization methods. These advancements were achieved by integrating advanced transformer-based protein language models (PLMs) and additional feature encodings, leading to a more accurate, robust, and interpretable neuropeptide prediction model. The outcomes of this endeavor were expected to significantly contribute to the field of bioinformatics by providing a state-of-the-art tool for neuropeptide identification, essential for understanding neural communication and for developing new treatments for neurological disorders.**

Index Terms—**Neuropeptides, Convolutional neural networks (CNNs), Transformer, Prediction.**

I. Introduction

Neuropeptides are small biologically active molecules found in the nervous systems of all animals, including humans. These molecules play a vital role in cell-to-cell communication and are responsible for regulating numerous physiological processes and behaviors. Accurately predicting and identifying neuropeptides is crucial for advancing the understanding of neural communication and developing therapies for neurological disorders. Computational models like NeuroPred-PLM have made significant strides in improving neuropeptide prediction by utilizing convolutional neural networks (CNNs) combined with global multi-head attention mechanisms, enhancing the semantic representation of protein sequences.

In this research, the objective was to further optimize the NeuroPred-PLM model by focusing on both prediction accuracy and interpretability. Modifications were made to the existing model architecture, resulting in a new version named DeepProtein. The EsmModel architecture was fine-tuned using transfer learning, and multi-scale CNN layers were employed to extract local sequence features. Additionally, global multi-head attention was utilized to capture long-range dependencies in protein sequences. Innovations in data preprocessing, visualization techniques, and regularization methods were also introduced to push the boundaries of current neuropeptide prediction approaches.

The motivation behind this work arose from the increasing need to enhance neuropeptide prediction tools, which are essential for bioinformatics research and drug discovery. Despite the promise of current models, limitations in accuracy and transparency still exist. By addressing these challenges, this research aimed to accelerate peptide research, contributing to the development of new treatments for neurological disorders and expanding the understanding of neural communication.

II. Description of the Problem

In the domain of **neuropeptide prediction**, significant challenges have persisted due to the complex nature of biological sequences. Neuropeptides are critical for cellular communication within the nervous system, regulating numerous physiological functions. However, identifying neuropeptides from protein sequences has remained a challenging task due to intricate structures and varying sequence patterns. Traditional methods for neuropeptide identification often prove time-consuming and lack the predictive accuracy required for large-scale studies.

Existing computational models, such as **NeuroPred-PLM**, while promising, exhibit significant limitations in accuracy and interpretability. Accurate and reliable prediction of neuropeptides is crucial for advancing biological research and developing treatments for neurological disorders. Current models face obstacles in capturing intricate dependencies between amino acids and generating meaningful predictions. The black-box nature of many machine learning models complicates the interpretation of how specific features contribute to predictions, posing barriers to practical applications in bioinformatics.

To address these challenges, modifications to the layers and hyperparameters of the existing PLM have been considered. Enhancements in the model's ability to predict neuropeptide sequences have been achieved. The revised model, named **DeepProtein**, aimed to improve prediction accuracy through these strategic adjustments. This research underscores the importance of accurately predicting neuropeptides, which play crucial roles in various biological processes. The final objective is the successful prediction of neuropeptide sequences, contributing to a deeper understanding of their functions in biological systems.

III. Literature Review

Neuropeptides play a central role in neural communication and have diverse physiological functions across species. Accurate prediction of neuropeptides from protein sequences is a crucial task in bioinformatics. The complexity of neuropeptide sequences presents significant challenges to traditional

TABLE I
SUMMARY OF COMPUTATIONAL METHODS FOR NEUROPEPTIDE PREDICTION

Model	Approach	Strengths	Limitations	References
Motif-based Search	Sequence alignment, motif recognition	Simple, interpretable	Low sensitivity, limited scalability	Meydan et al. (2013) [1]
CNN-based Methods	Convolutional layers for feature extraction	Improved local feature extraction	Limited global context capture	Li et al. (2008) [2]
SVM and ML-based Models	Machine learning with feature engineering	Good for small datasets	Requires manual feature selection	Salam et al. (2024) [3]
LSTM-based Models	Recurrent neural networks for sequence dependencies	Captures long-range dependencies	Sequential data processing limitations	Yi et al. (2019) [4]
Transformer-based Models	Attention mechanisms for global context	Captures long-range dependencies effectively	Model interpretability challenges	Rives et al. (2021) [5]
Transfer Learning-based Models	Pre-trained protein language models	Generalizes across diverse datasets	Requires large computational resources	Elnaggar et al. (2021) [6]

prediction methods, necessitating advanced machine learning approaches for improved accuracy and interpretability. In recent years, several models have been developed, each employing different techniques to enhance the accuracy and efficiency of neuropeptide prediction. This section reviews key advancements in the field, focusing on computational methods and deep learning architectures that have been applied to neuropeptide prediction.

A. Traditional Approaches to Neuropeptide Prediction

Early methods of neuropeptide prediction relied heavily on sequence alignment techniques and motif-based searches. These approaches, such as those discussed by [1], were limited by the complexity and variability of neuropeptide sequences across species. Motif-based approaches aimed to identify conserved patterns within protein sequences that are characteristic of neuropeptides, but these techniques often struggled with low sensitivity and specificity, especially for peptides that did not adhere to clear sequence motifs. Additionally, the rapid expansion of peptide datasets posed further challenges, as traditional methods were ill-equipped to handle large-scale data.

B. Machine Learning-Based Methods

To overcome the limitations of traditional approaches, machine learning models have been increasingly applied to neuropeptide prediction. [3] proposed the use of support vector machines (SVMs) and other machine learning techniques to predict bioactive peptides. However, these models required extensive feature engineering and struggled with capturing complex sequence dependencies.

Subsequently, the field moved towards more advanced methods, such as the NeuroPred model, which utilized convolutional neural networks (CNNs) to improve the prediction of neuropeptides. [2] introduced a CNN-based approach that focused on learning hierarchical features from peptide sequences. While this method improved prediction performance over traditional techniques, it still faced challenges in generalizing across diverse peptide families and integrating global sequence information.

C. Deep Learning and Transformer-Based Models

The introduction of deep learning architectures, such as recurrent neural networks (RNNs) and long short-term memory (LSTM) networks, enabled further improvements in peptide prediction. [4] employed LSTMs to capture long-range dependencies in peptide sequences, offering better performance than earlier models. However, these models were still limited by their reliance on sequential data processing, which made it difficult to capture global patterns in large datasets.

Recent advancements in transformer-based architectures, such as BERT and its variants, have significantly improved protein sequence modeling. Transformers, which rely on attention mechanisms, allow for better representation of long-range dependencies in sequences. Notably, the NeuroPred-PLM model combined convolutional layers with global multi-head attention, providing state-of-the-art results in neuropeptide prediction [5]. This architecture effectively integrates local and global information, allowing the model to capture both fine-grained features and long-distance dependencies. Additionally, the use of pre-trained protein language models (PLMs) has led to improvements in feature extraction from biological sequences, as demonstrated by [5].

D. Challenges and Opportunities in Neuropeptide Prediction

While deep learning models such as CNNs, LSTMs, and transformers have shown great promise, challenges remain. One key issue is the interpretability of these models. Many of the current architectures function as ”black boxes,” making it difficult for researchers to understand how the model arrives at its predictions. Additionally, there is still a need for models to handle the vast diversity in neuropeptide structures and functions. The integration of transfer learning, as explored by [6], presents an opportunity to address these challenges by enabling models to generalize across different datasets.

Further developments in neuropeptide prediction, such as the proposed iNP_ESM model, aim to enhance accuracy while improving model transparency and interpretability. The model’s use of advanced data preprocessing, visualization techniques, and refined regularization methods represents a promising step forward in addressing the current limitations of neuropeptide prediction technologies [7].

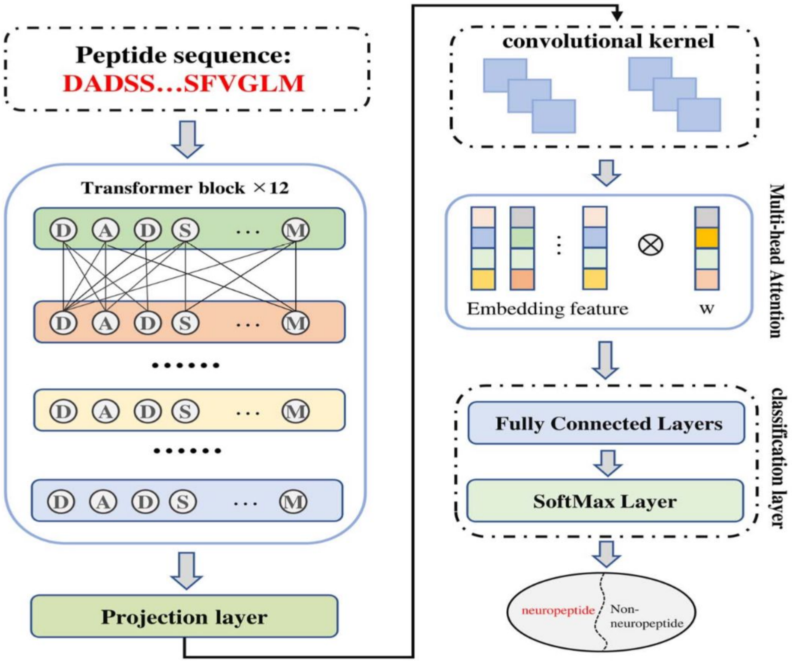

Fig. 1. The flowchart of the model architecture.

IV. Methodology

This research employed deep learning methodologies for **neuropeptide sequence prediction**. The approach included processing a curated dataset of positive (neuropeptide) and negative (non-neuropeptide) sequences, followed by implementing a **Transformer-based model** for classification. The methodology encompassed several stages: **data collection**, **preprocessing**, **model architecture**, **training**, and **evaluation**.

A. Dataset

Data was sourced directly from the **NeuroPred-FRL** website, providing an updated version that included **2,929 new sequences**, resulting in a total of **5,214 neuropeptide** and **6,641 non-neuropeptide** sequences. The raw dataset was organized into four files: two training files (one for positive and one for negative sequences) and two testing files (also divided into positive and negative). These files were combined, and random shuffling was applied to ensure a more generalized dataset for subsequent training and evaluation.

B. Data Preprocessing

The sequence data was prepared using the **AutoTokenizer** from the pre-trained **facebook/esm2_t33_650M_UR50D** model. This tokenizer applied padding and truncation to standardize each input to a uniform length of **128 tokens**, converting training and testing sequences into a suitable format for the model. The training and testing labels were extracted and converted into tensors for efficient processing during model training and evaluation. This stage was essential for structuring the raw sequence data for deep learning applications.

C. Model Architecture

The model architecture presented in this research is designed to predict neuropeptide sequences effectively [8]. It is based on a deep learning framework that incorporates advanced techniques such as **embedding layers**, **transformer encoder layers**, **convolutional layers**, **multi-head attention mechanisms**, and **fully connected layers**. These components work collaboratively to extract meaningful features from the sequence data and perform prediction tasks.

Feature Extraction from the DeepProtein Model

Feature extraction transformed raw input sequences into meaningful representations using several processing layers. The embedding layer first converted token IDs into dense vector representations, facilitating a richer understanding of the input data. The subsequent transformer encoder layers further refined these representations by applying self-attention mechanisms, allowing the model to focus on relevant parts of the sequence.

Model Layers

Transformer Block

The model contained **six transformer encoder layers**, each responsible for processing sequential data. Transformers were chosen for their capacity to capture long-range dependencies, which are crucial for biological data. Within each transformer layer, the self-attention mechanism played a crucial role by computing attention scores that determined the focus on different parts of the input sequence. This mechanism allowed the model to dynamically weigh the importance of different tokens, enhancing its ability to capture contextual dependencies. Mathematically, this can be represented as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

Where Q, K, and V represent the query, key, and value matrices, and d_k denotes the dimensionality of the key vectors. Linear transformations were applied within each transformer block to project the input data into a space suitable for attention calculations, allowing the model to learn complex representations through weight adjustments during training. Layer normalization and dropout were also employed to stabilize and accelerate training, as well as to prevent overfitting. The transformer layer's structure can be described as:

$$X^{(i)} = \text{TransformerEncoderLayer}^{(i)}(X^{(i-1)}) \tag{2}$$

where $X^{(0)}$ is initialized with the embedding output, and $X^{(N)}$ is the final output of the transformer layers.

Projection Layer

The projection layer reduced the dimensionality of the transformer layer outputs, enabling more efficient handling of data in subsequent stages. The transformation in the projection layer was given by:

$$P = X^{(N)}W_p + b_p \tag{3}$$

where W_p and b_p represent the projection weight matrix and bias term, respectively.

Convolutional Layers

Two 1D convolutional layers followed the projection layer to further refine the features. The first convolution used a kernel size of **3** to capture local patterns, and the second convolution, with a kernel size of **1**, reduced the dimensionality while preserving crucial features. ReLU activations introduced non-linearity. This can be expressed as:

$$C_1 = \text{ReLU}(\text{Conv1D}(P)) \tag{4}$$

$$C_2 = \text{ReLU}(\text{Conv1D}(C_1)) \tag{5}$$

Where C_1 and C_2 represent the intermediate representations capturing hierarchical features within the sequence.

Multi-Head Attention

It was applied to enable the focus on various parts of the input sequence simultaneously. Each attention head computed attention scores based on the learned weight matrix W, with the attention calculation represented as:

$$A = \text{softmax}(C_2 \cdot W^T) \tag{6}$$

This allowed the model to weigh the importance of different features, resulting in the overall learned representations.

Classification Layers

The final representation from the attention mechanism was fed into fully connected layers for classification. The representation was first processed through a ReLU activation function, followed by linear transformations to produce the logits for classification. The softmax function was then used to interpret these logits as class probabilities, as represented by:

$$\hat{y} = \text{softmax}(F_2) \tag{7}$$

This output indicated the probability distribution over the classes, with the highest probability indicating the predicted class.

Total Trainable Parameters

The model contained approximately **1,686,912 trainable parameters**. The extensive use of linear transformations enabled efficient learning of complex mappings between input features and output classes, maintaining flexibility in capturing intricate relationships.

D. Loss Function and Optimizer

During training, the model utilized **cross-entropy loss** to evaluate the discrepancy between the predicted logits and true labels. This loss function was defined as:

$$L = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(\hat{y}_i) \tag{8}$$

Where y_i and $\hat{y}_i$ represent the true label and the predicted probability for each sample.

The **Adam optimizer** was chosen for updating the model's parameters, offering efficient training through its adaptive learning rates and momentum, ensuring faster convergence.

E. Performance Metrics

Various metrics assessed the model's classification performance, including **accuracy (ACC)**, **F1 score (F1)**, **precision (Pre)**, and **recall (Rec)**. These metrics provided a comprehensive evaluation of the model's ability to accurately classify sequences, reflecting its overall effectiveness in neuropeptide classification. The corresponding formulas are:

$$\text{Pre} = \frac{TP}{TP + FP} \tag{9}$$

$$\text{Rec} = \frac{TP}{TP + FN} \tag{10}$$

$$F1 = \frac{2 \cdot (\text{Pre} \cdot \text{Rec})}{\text{Pre} + \text{Rec}} \tag{11}$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

where TP, TN, FP, and FN denote the numbers of true positive, true negative, false positive, and false negative samples, respectively.

V. Proposed Improvements and Ideas

Here are some ideas which can be developed differently and will improve the overall performance:

A. Utilization of an Updated Dataset

The dataset had been updated with an additional **2,929 sequences**, increasing the diversity and volume of samples. This augmentation was intended to improve generalization and model robustness by exposing the model to a broader spectrum of neuropeptide and non-neuropeptide sequences, enhancing prediction reliability in biological contexts.

B. Implementation of Tokenization

The **AutoTokenizer** from the pre-trained **facebook/esm2_t33_650M_UR50D** model had been applied for tokenization. This standardized all input sequences to a uniform length of **128 tokens**, ensuring compatibility with the transformer architecture. Such preprocessing had been aimed at improving feature extraction and model performance by facilitating efficient data representation.

C. Proposed Modifications to Model Architecture

Modifications to the model architecture are being proposed with the aim of increasing prediction accuracy. The revised architecture, to be named DeepProtein, will involve fine-tuning the existing layers and integrating additional components to enhance feature extraction and representation learning. These adjustments are expected to enable the model to capture complex relationships within the sequence data more effectively.

D. Incorporation of Advanced Regularization Techniques

The incorporation of advanced regularization techniques, including **dropout** and **layer normalization**, is being proposed to mitigate the risk of overfitting. These methods are intended to balance model complexity with performance, particularly on unseen data, thereby improving generalization without compromising prediction accuracy.

E. Proposed Optimization of Model Architecture

Optimizations to the model architecture are being considered to reduce computational load. These proposals include reducing the number of transformer layers and simplifying certain components, resulting in more resource-efficient processing. This strategy is expected to preserve predictive accuracy while lowering the computational burden, making the model more accessible in environments with limited resources.

References

[1] C. Meydan, H. H. Otu, and O. U. Sezerman, "Prediction of peptides binding to MHC class I and II alleles by temporal motif mining," *BMC Bioinformatics*, vol. 14, Suppl 2, S13, 2013. https://doi.org/10.1186/1471-2105-14-S2-S13.

[2] Z. C. Li, X. B. Zhou, Z. Dai, and X. Y. Zou, "Prediction of protein structural classes by Chou's pseudo amino acid composition: Approached using continuous wavelet transform and principal component analysis," *Amino Acids*, vol. 37, pp. 415-425, 2008. https://doi.org/10.1007/s00726-008-0170-2.

[3] A. Salam, F. Ullah, F. Amin, I. A. Khan, E. Garcia Villena, A. K. Castilla, and I. de la Torre, "Efficient prediction of anticancer peptides through deep learning," submitted April 29, 2024, accepted June 11, 2024, and published July 19, 2024.

[4] H. C. Yi, Z. H. You, X. Zhou, L. Cheng, X. Li, T. H. Jiang, and Z. H. Chen, "ACP-DL: A Deep Learning Long Short-Term Memory Model to Predict Anticancer Peptides Using High-Efficiency Feature Representation," *Mol Ther Nucleic Acids*, vol. 17, pp. 1-9, Sept. 2019. doi: 10.1016/j.omtn.2019.04.025. PMID: 31173946; PMCID: PMC6554234.

[5] A. Rives, J. Meier, T. Sercu, and R. Fergus, "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences," *Proceedings of the National Academy of Sciences*, vol. 118, no. 15, e2016239118, 2021. https://doi.org/10.1073/pnas.2016239118.

[6] A. Elnaggar, M. Heinzinger, C. Dallago, G. Rihawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, D. Bhowmik, and B. Rost, "ProtTrans: Towards Cracking the Language of Life's Code Through Self-Supervised Deep Learning and High Performance Computing," arXiv, July 13, 2020, last revised May 4, 2021. arXiv:2007.06225v3 [cs.LG].

[7] H. Li, L. Jiang, K. Yang, S. Shang, M. Li, and Z. Lv, "iNP_ESM: Neuropeptide Identification Based on Evolutionary Scale Modeling and Unified Representation Embedding Features," *International Journal of Molecular Sciences*, vol. 25, no. 13, article 7049, 2024. https://doi.org/10.3390/ijms25137049.

[8] L. Wang, C. Huang, M. Wang, Z. Xue, Y. Wang, "NeuroPred-PLM: an interpretable and robust model for neuropeptide prediction by protein language model," *Briefings in Bioinformatics*, vol. 24, no. 2, bbad077, Mar. 2023. https://doi.org/10.1093/bib/bbad077