NeuroPred: Leveraging Machine Learning For Neuropeptide Sequence Prediction

Aryan Gandotra[1], Lakshit Wasan[1], Priyansh Jain[1], Syed Mohd Abid[1]

[1]Department of Computer Science and Engineering,
School of Engineering and Technology,
BML Munjal University, Gurugram, Haryana 122413, INDIA

***Abstract*—Because neuropeptides are important signaling molecules for such a broad range of biological processes it is essential to identify them so that they may be used in drug discovery and therapeutic purposes. However, there are many ML and DL techniques for neuropeptide prediction, and none produce models that are interpretable and improve in performance. In this study we describe a novel neuropeptide prediction model that automatically generates semantic representations of neuropeptides and uses a protein language model. First, it uses the basic machine learning algorithms: The models here are: LR, KNN, SVM, NB, and later advances to models such as Random Forest, Decision Tree, XGBoost, catboost and the mania of deep learning models: ANN. Various evaluatiaon metrics were used to evaluate the performance under this study. This study uses the primary dataset from the "NeuroPep 2.0" database resulting from the "NeuroPred-PLM" study, where sequences were trimmed to sizes between 5 and 100 characters.**

***Index Terms*—Neuropeptide prediction, Protein language model, Machine learning, Deep learning, Feature extraction, Neuropep 2.0, NeuroPred-PLM, Logistic Regression, Random Forest, ANN, CNN, Semantic representations, Drug discovery, Matthew's correlation coefficient.**

I. Introduction

Neuropeptides (NPs) are chains of amino acids that mediate many processes such as metabolism, behaviour, pain perception and cell communication. In vertebrates and invertebrates, NPs are signaling molecules and key regulators of physiological functions. Accurate identification and prediction of neuropeptides is critical to further elucidating peptide based communication and therapeutic applications, given their role in neurobiology.

High costs and time consuming processes have limited conventional experimental methods for NPs identification, typically by mass spectrometry and liquid chromatography. Furthermore, these methods do not give a complete understanding of the sequence derived features of neuropeptides. To overcome these limitations, we have developed machine learning (ML) models for more efficient and accurate prediction of neuropeptides from sequence information. In this review, we present a detailed exploration of the machine learning models used for NP prediction in recent studies, summarizing key methodologies, encoding techniques, and evaluation metrics used.

II. Literature Review

Neuropeptides (NPs) are a class of biomolecules that are known to be critically involved in a wide range of biological processes, including neural signaling, metabolism and behavior regulation. Determination of neuropeptide identity and prediction are crucial for understanding peptide based signaling pathways and the development of peptide based therapeutics. Researchers have expended considerable effort over the years to develop computational models for neuropeptide prediction, taking advantage of recent advances in machine learning and feature engineering.

An example of this is the NeuroPred-FRL model, which combines several encoding techniques, classifiers, and a complex two step feature selection process to improve neuropeptide prediction accuracy [1]. Robust predictive performance is shown by the model's comprehensive feature extraction, using 11 encoding schemes, and the combination of several machine learning classifiers, including SVM, ERT, RF, AdaBoost, KNN, and Naive Bayes. In the two step feature selection method, XGBoost is used to rank feature importance and a consecutive forward search with Random Forest classifier is used to select the most relevant features for neuropeptide identification.

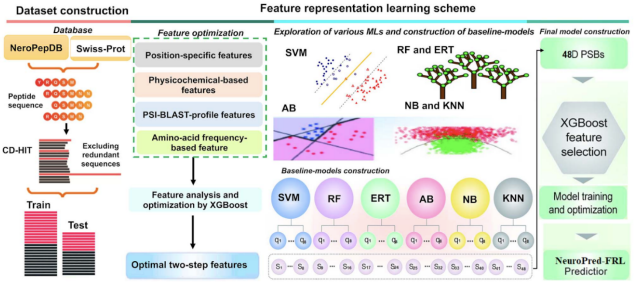

Fig. 1. A description of NeuroPred-FRL's overall workflow of development. The specific steps are: dataset construction, feature extraction, exploring various ML classifiers, building baseline models and final meta model construction.

The iNP_ESM model builds upon the success of NeuroPred FRL and introduces a novel approach to neuropeptide identification. iNP_ESM is instead based on protein language models, particularly Evolutionary Scale Modeling (ESM) and Unified

Representation (UniRep) to extract meaningful sequence based features [3]. By using deep learning based embeddings that capture rich semantic and structural information from protein sequences, we use them as inputs to a Support Vector Machine (SVM) classifier. We evaluated the model's performance over a number of widely used machine learning algorithms, and showed that the language model based feature representations are effective.

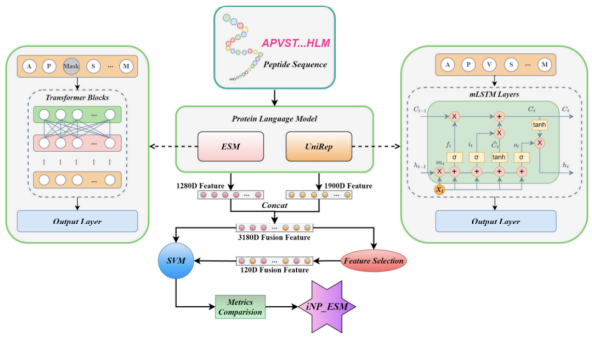

Fig. 2. Overview of the iNP_ESM Model.

The NeuroPred-Fuse model is another innovative approach to neuropeptide prediction that uses a two layer stacking architecture [2]. Six sequence-derived feature encoding schemes are integrated with multiple feature selection techniques in the first layer. The complementary strengths of the different encoding schemes and classifiers are coaxed: the outputs of the first layer are merged and fed into a logistic regression classifier in the second layer.

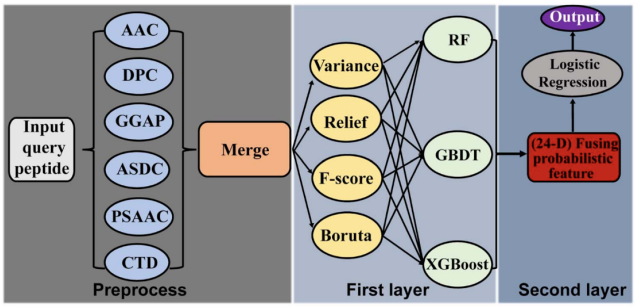

Fig. 3. Workflow of NeuroPpred-Fuse.

The NeuroPID model is concerned with the identification of neuropeptide precursors (NPPs) rather than the mature neuropeptides. Using a supervised machine learning approach, we train on a dataset of manually annotated NPPs from the UniProtKB database [15]. Each sequence is transformed into a vector of approximately 560 primary sequence derived features including amino acid composition and biophysical properties via feature extraction. We evaluated the performance of the model using various machine learning algorithms and showed its potential to identify neuropeptide precursors.

Advanced techniques are integrated into the NeuroPred-PLM model to improve both the prediction accuracy and interpretability of neuropeptide identification [17]. The model takes advantage of a Protein Language Model (ESM) to represent peptide sequences in rich semantic space, from which the model learns locally relevant features using a Projection Layer and Multi-Scale Convolutional Neural Networks (CNN). To enhance the interpretability of the model, a Global Multi-Head Attention Network is introduced.

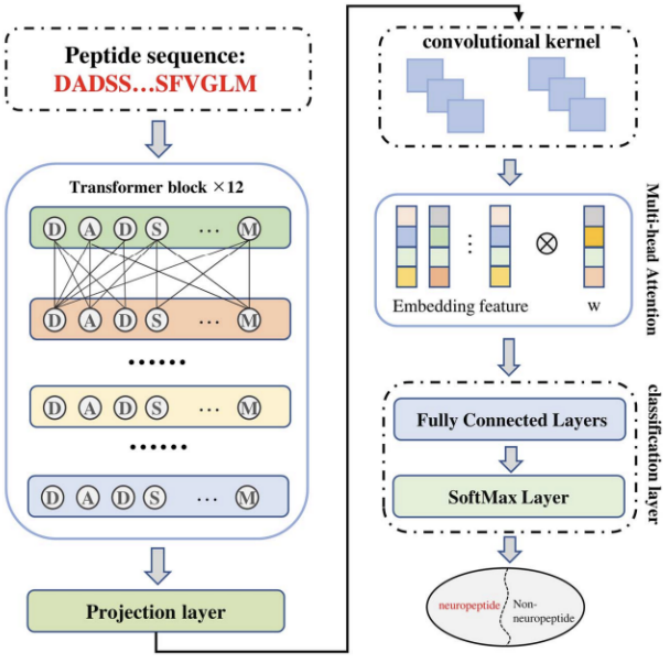

Fig. 4. The flowchart of NeuroPred-PLM.

PredNeuroP is another ensemble based approach that combines all sorts of machine learning algorithms and a large set of feature descriptors through a two layer stacking framework. Combining nine feature groups with five different machine learning algorithms, the first layer generates base models. Second layer: The outputs from the top performing base models is given to a logistic regression classifier.

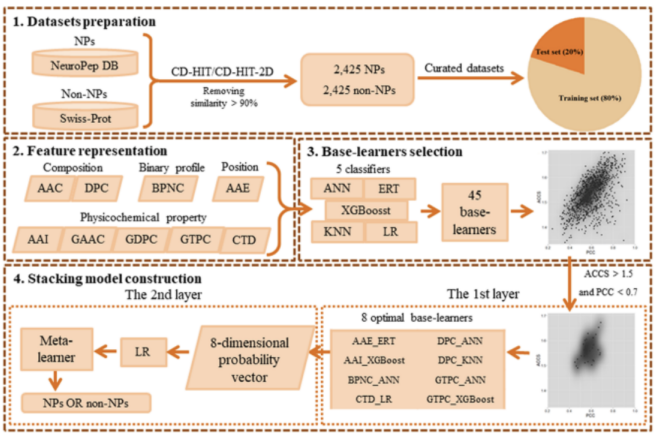

Fig. 5. Framework of PredNeuroP based on the stacking method.

The NeuroCNN_GNB model is an ensemble of convolutional neural networks (CNN) trained on particular feature encoding schemes [4]. A Gaussian Naive Bayes (GNB) classifier is used to integrate these baseline CNN models. For interpretability, the Shapley Additive exPlanation (SHAP) algorithm is further used to enhance the model performance.

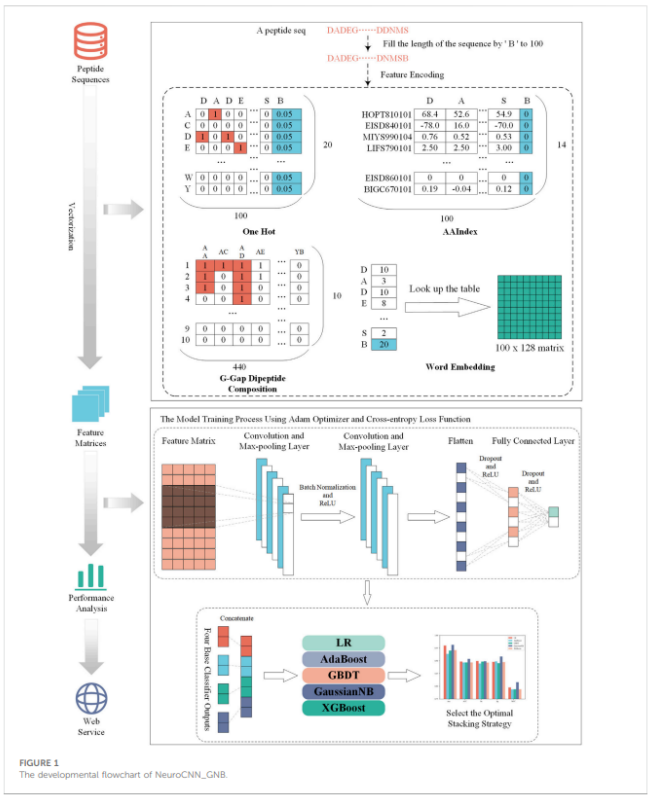

Fig. 6. The developmental flowchart of NeuroCNN_GNB.

Finally, the reviewed models show diverse ways of neuropeptide prediction, including feature engineering, machine learning integration, protein language model based representations and ensemble. Stacking ensemble approaches and deep learning based features are demonstrated by the NeuroPred-FRL, iNP_ESM, NeuroPredFuse and PredNeuroP models. NeuroPred-PLM, NeuroCNN_GNB and NeuroPID all emphasize precursor identification, while NeuroPred-PLM and NeuroCNN_GNB also try to improve interpretability. Future work will investigate more advanced deep learning architectures, incorporate structural information, and utilize multimodal approaches, to improve neuropeptide prediction models.

III. Materials and Methods

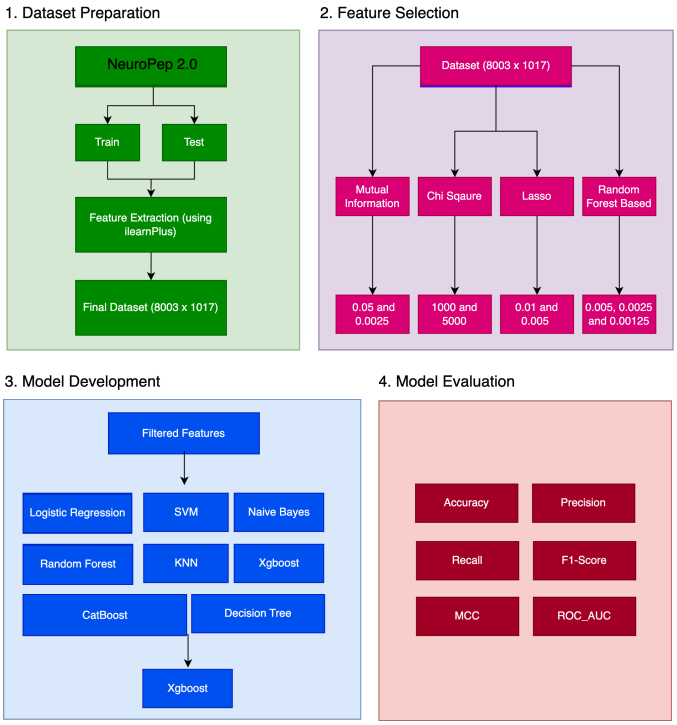

Fig. 7. Schematic framework of the NeuroPred model.

A. Datasets

For this study, the dataset used was pulled from the NeuroPep 2.0 database, which contained 11,282 experimentally validated neuropeptide sequences, with 5,333 newly added in version 2.0. After filtering, 8,003 sequences were retained, which contained neuropeptides of lengths from 5 to 100 amino acids.

B. Feature Extraction

A preprocessed protein sequence dataset was used as the input, which was then subjected to feature extraction using the iLearn Plus software, a specialized tool for automatic feature generation in bioinformatics and machine learning tasks. iLearn Plus converts raw protein sequence data into an integrated format that is amenable to downstream machine

learning analysis. Four key feature descriptors were extracted: AAC, ASDC, CTDD, and DPC. Such features are required to understand the functional and structural characteristics of proteins.

*1) **AAC**:* This descriptor calculates the relative frequency of each of the 20 standard amino acids in a sequence, defined as:

$$\text{AAC}(i) = \frac{f_i}{N}$$

*2) **ASDC**:* This descriptor computes the distribution of amino acids across the sequence and is represented as:

$$\text{ASDC}_{ik} = \frac{P_k}{L} \times 100,$$

*3) **CTDD**:* This descriptor captures sequence-order information by calculating composition (C), transition (T), and distribution (D). Transition (T) is defined as:

$$T(r,s) = \frac{N(r,s) + N(s,r)}{N-1},$$

*4) **DPC**:* This descriptor computes the frequency of dipeptides as:

$$\text{DPC}(j) = \frac{M_j}{N-1},$$

The features were extracted and stored as individual CSV files and then merged together in to a single dataset for further machine learning model development. The final dataset, containing protein sequences in one-letter amino acid codes along with their biological classifications, is publicly available at: https://github.com/AryanGandotra/NEURO-PRED/blob/main/Features/Data%202/Train/combined_data.csv

C. Workflow

The *NeuroPred* framework follows a structured workflow: Initially, peptide sequences were encoded using four distinct feature types. A feature selection procedure was subsequently applied to these encodings, combining the features into a unified set. To refine the feature set further, four separate feature selection techniques were utilized to pinpoint the most relevant and non-redundant features. Following this, eight base classifiers, each derived from unique individual classifiers, were integrated. The performance of the fused classifiers was assessed using metrics such as Acc., Prec., Recall, F1 Score, MCC and the ROC-AUC score."

D. Feature Selection

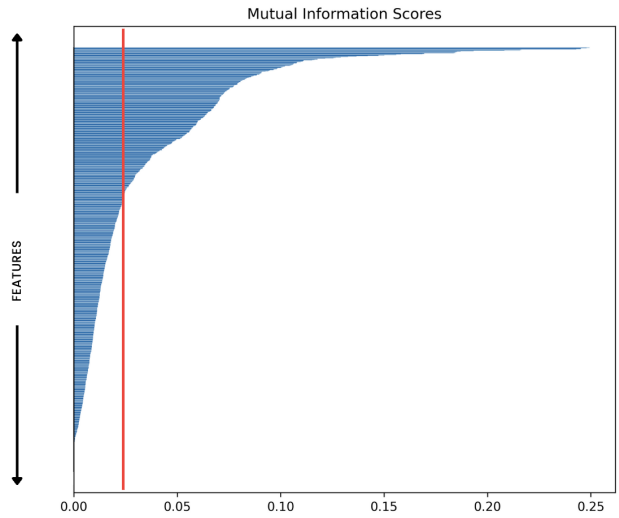

Fig. 8. Feature Selection using MIF with threshold greater than 0.025

The study invariably involves feature selection to select the most relevant and non-redundant features to improve the model's performance while reducing computational complexity. In this study, four feature selection methods were employed: "**Mutual Information, Chi-Square, Lasso Regression,**" and "**Random Forest**". Below are the details of the techniques and corresponding formulas.

1. Mutual Information Feature Selection: The amount of information shared between a feature and the target variable is quantified by Mutual Information (MI), and hence feature relevance can be evaluated. Then, we calculated MI scores for each feature and rank them, with a threshold of 0.05 initially, and we got 219 features. They were evaluated using the before mentioned classifiers and metrics. To further explore feature relevance, we decreased the threshold to 0.025 and found 342 features, but performance improvement was negligible. The formula for MIF between two variables X and Y is:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} P(x,y) \cdot \log \frac{P(x,y)}{P(x)P(y)}$$

2. Chi-Square Feature Selection: The Chi-Square test tests the association between a categorical feature and a target feature compared with the null hypothesis under observed and expected frequency. First, we used thresholds of Chi-Square scores greater than 1000 and 5000 to find 100 and 31 features, respectively, in this study. Further hyperparameter tuning discovered that a Chi-Square score greater than 1000 yielded large accuracy improvements on many models. The formula for the Chi-Square test is:

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

3. Lasso Feature Selection: Lasso regression has an additional regularization term that shrinks absolute size of regression coefficients, so some coefficients are shrunk to zero. It enables the discovery of important features for prediction. Initially, a threshold of 0.01 was used to select 46 features. At a lower threshold of 0.005, we found 80 features but could not significantly improve model accuracy. The formula for Lasso regression is:

$$\beta = \arg\min \left(\frac{1}{2n} \sum_{i=1}^{n} (y_i - \mathbf{x}_i \cdot \beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right)$$

4. Random Forest Feature Selection: The feature importance evaluated by Random Forest is to train multiple decision trees and aggregating their predictions. Relevant features were identified using feature importance scores, with thresholds of 0.005, 0.0025 and 0.00125 tested with 30, 68 and 154 features respectively. Using the selected metrics the selected features were further evaluated. For classification, the Random Forest prediction is:

$$y = \text{Mode}(T_1(x), T_2(x), \ldots, T_m(x))$$

For regression, it is:

$$y = \frac{1}{m} \sum_{i=1}^{m} T_i(x)$$

Through these comprehensive feature selection techniques, optimal features were identified, significantly improving model performance. Each method provided unique insights, ensuring a robust and interpretable feature selection process.

E. Model Development

In developing a robust neuropeptide prediction system, the study investigated a wide range of ML and DL methods. This iterative approach aimed to identify models that were not only accurate but also interpretable and generalizable. Both basic and advanced predictive models were constructed using traditional ML algorithms such as LR, SVM, KNN, NB, DT, RF, XGBoost, and CatBoost.

Deep learning techniques were further deployed using a base artificial neural network (ANN) and more complex architectures that continue down to adding additional layers, dropout and batch normalization. The resulting predictive framework was more accurate and more efficient for model training.

1. Logistic Regression estimates the likelihood of an instance being assigned to a specific class by applying the sigmoid function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

2. SVM seeks to identify the hyperplane that maximizes the separation between different classes. The optimization problem is formulated as:

$$\min_{w,b} \frac{1}{2} ||w||^2$$

Subject to:

$$y_i(w \cdot x_i + b) \geq 1, \forall i$$

3. KNN assigns a class to an instance by taking a majority vote from its k-nearest neighbors, using a chosen distance metric:

For classification:

$$y = \text{Mode}(y_{i1}, y_{i2}, \ldots, y_{ik})$$

4. Naive Bayes It uses Bayes' theorem under the assumption of feature independence:

$$P(C_k|x) = \frac{P(C_k)P(x|C_k)}{P(x)}$$

5. Decision Tree
Decision Trees divide data at nodes to reduce impurity, employing different measures.

Gini impurity:

$$\text{Gini} = 1 - \sum_{i=1}^{C} p_i^2$$

Entropy:

$$\text{Entropy} = -\sum_{i=1}^{C} p_i \log_2(p_i)$$

Information Gain:

$$\text{Information Gain} = \text{Entropy}_{\text{parent}} - \sum_{j=1}^{k} \frac{N_j}{N} \text{Entropy}_{\text{child}_j}$$

6. Random Forest (RF) It is a collection of decision trees that combines their predictions to improve predictions and minimize overfitting. For classification:

$$y = \text{Mode}(T_1(x), T_2(x), \ldots, T_m(x))$$

For regression:

$$y = \frac{1}{m} \sum_{i=1}^{m} T_i(x)$$

7. XGBoost XGBoost is a gradient-boosting method that enhances predictions by minimizing an objective function:

$$L(\Theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

8. CatBoost CatBoost is a gradient-boosting method designed to efficiently manage categorical features, using ordered boosting to mitigate overfitting and bias.

Feed-Forward Neural Network (FFNN): A simple FFNN with multiple hidden layers is used as the baseline model. The ReLU activation is used in each layer and dropout added to ensure that we not over fit. For the binary classification what we have used the final output layer which is taking using a sigmoid activation. This model can be used as a baseline against which to compare more advanced configurations.

Deep Neural Network with Dropout (DNN-D): The FFNN is extended with additional hidden layers and dropout regularization with a higher rate (0.3) in the DNN-D. To cleaner the learning process, we also use batch normalisation at every layer in the model. The better performance and generalization provided from this more complex model makes it an ideal model to model more complex data patterns.

Regularized Deep Neural Network (R-DNN): The R-DNN model outperforms the DNN-D by introducing L2 regularization to reduce model complexity and improve generalization. LeakyReLU activation is also introduced to overcome the issue with vanishing gradients, batch normalization and dropout regularization are also introduced as well. We build this model aiming to achieve the maximum performance and robustness against overfitting in this case of complex or large datasets.

By combining these traditional ML methods with more advanced DL architectures, we were able to thoroughly explore predictive capabilities, with a highly accurate and interpretable neuropeptide prediction system.

F. Evaluation Metrics

Several key metrics were used to evaluate the proposed models on how well they predicted. The metrics are correct, sensitive, and robust, and thus give an accurate representation of model performance. The evaluation criteria used include:

- **Precision**

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall**

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1-Score**

$$\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Accuracy**

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Matthew's Correlation Coefficient**

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

G. Hyperparameter Tuning

Hyperparameter tuning was performed for the following traditional models using grid search and random search techniques. The details of the hyperparameters for each model are listed below:

Model	Hyperparameter	Values
LR	`C`	[0.1, 1, 10, 100]
	`max_iter`	[100, 1000, 10000]
SVM	`C`	[0.1, 1, 10, 100]
	`gamma`	['scale', 'auto']
KNN	`n_neighbors`	[3, 5, 7, 9]
	`weights`	['uniform', 'distance']
	`algorithm`	['ball_tree', 'kd_tree', 'brute']
NB	`var_smoothing`	[1e-9, 1e-10, 1e-11, 1e-12]
DT	`criterion`	['gini', 'entropy']
	`splitter`	['best', 'random']
	`max_depth`	[10, 20, ... 90, 100]]
RF	`n_estimators`	[100, 200, 300, 400, 500]
	`criterion`	['gini', 'entropy']
	`max_depth`	[10, 20, ... 90, 100]
XGBoost	`n_estimators`	[100, 200, 300, 400, 500]
	`max_depth`	[3, 4, 5, 6, 7, 8, 9, 10]
CatBoost	`iterations`	[100, 200, 300, 400, 500]
	`depth`	[3, 4, 5, 6, 7, 8, 9, 10]

TABLE II
HYPERPARAMETERS FOR DIFFERENT MACHINE LEARNING MODELS

H. Model Evaluation

Metrics described above were rigorously evaluated using five fold cross validation to ensure robust and reliable results. The best performing model was chosen by its ability to consistently earn high scores on all evaluation metrics.

The results, summarized in Table III, we show the performance of each model on different metrics. Finally, the **XGBoost** model had the highest accuracy, precision, recall, F1 score and ROC-AUC score. Notably, **XGBoost** also achieved the highest Matthew's Correlation Coefficient (MCC) making it the most well rounded model in terms of classification accuracy and robustness.

TABLE III
VALUES WITH FEATURE SELECTION OF MUTUAL INFORMATION WITH THRESHOLD GREATER THAN 0.025. THE XGBOOST VALUES ARE MARKED AS THE BEST BASED ON ACCURACY.

Model	Log Reg.	SVM	KNN	NB	DT	RF	XGBoost	CatBoost
Accuracy	0.836	0.836	0.824	0.774	0.839	0.896	**0.919**	0.918
Precision	0.830	0.819	0.806	0.834	0.836	0.866	**0.904**	0.903
Recall	0.841	0.858	0.850	0.678	0.839	0.934	**0.936**	0.934
F1 Score	0.835	0.838	0.828	0.748	0.838	0.899	**0.919**	0.918
MCC	0.672	0.672	0.650	0.557	0.678	0.794	**0.838**	0.836
ROC-AUC	0.836	0.836	0.825	0.773	0.839	0.896	**0.919**	0.918

As can be seen in Table 1, all key metrics show that **XGBoost** model performed better than others. This model performs better than any other model for the given task, especially in feature selection using Mutual Information (MI) with a threshold of 0.025.

IV. RESULTS AND DISCUSSION

The results from our evaluation underscore the outstanding performance and consistency of **XGBoost** in predicting neuropeptides. Specifically, **XGBoost**, using mutual information for feature selection and a threshold of **0.025**, demonstrated reliable performance across various evaluation metrics. Notably, these results were obtained without any hyperparameter tuning, emphasizing the stability and effectiveness of **XGBoost** for this task.

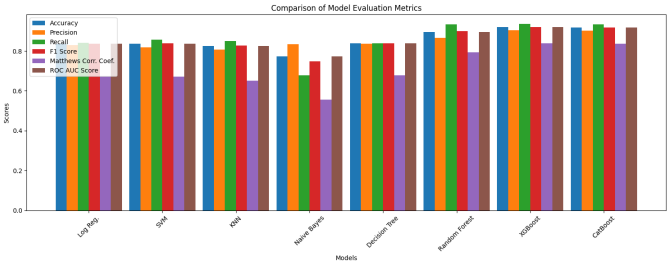

Fig. 9. Model Comparison over different metrics

To ensure a comprehensive evaluation, we trained and evaluated a total of **1120** distinct machine learning models, including "Logistic Regression, SVM, KNN, Naive Bayes, Decision Tree, Random Forest, XGBoost, and CatBoost" across multiple performance metrics: Accuracy, Precision, Recall, F1 Score, Matthews Correlation Coefficient, and ROC AUC Score. Among the models assessed, Random Forest, XGBoost, and CatBoost consistently demonstrated superior performance, with **XGBoost** emerging as the most reliable model for neuropeptide prediction tasks.

In conclusion, the evaluation results clearly indicate that **XGBoost**, with Mutual Information feature selection (threshold greater than 0.025), provides the best overall performance and is therefore the preferred model for neuropeptide prediction. However, depending on specific task requirements, such as prioritizing recall or model simplicity, alternative models such as "CatBoost" or even "Random Forest" may be suitable. In scenarios where reduced training time is essential, **CatBoost** offers a compelling option, given its competitive performance and efficiency.

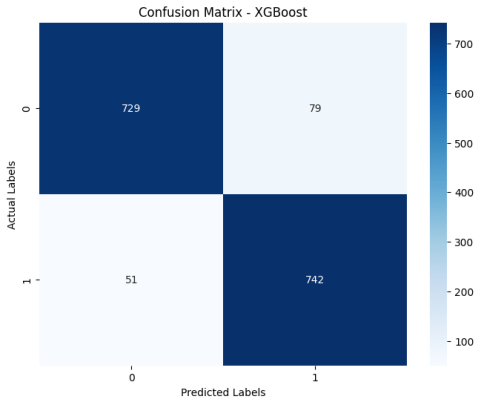

Fig. 10. Confusion matrix for XGBoost.

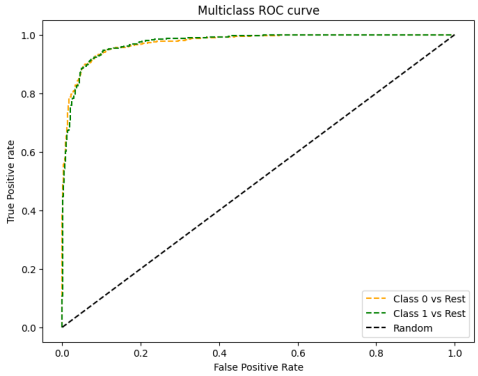

Fig. 11. Multiclass ROC Curve for XGBoost.

V. Conclusion

Finally, in this study, the study presents NeuroPred as a new prediction model that could enhance identification of neuropeptides using sequence data. In order to obtain a versatile and accurate model, four feature selection techniques and four different feature encoding methods were applied together with machine learning algorithms to integrate the most comprehensive data regarding amino acid sequences. This led to the formation of 1,017 features and these were combined to make multi-view probabilistic features to develop the model. Furthermore, feature and model explanatory analysis were conducted to determine the most important features for neuropeptide classification. There were DPC, CTD AAC, and ASDC encoding among the various techniques used.

Another independent test proved that NeuroPred outperformed the existing approaches and basic deep learning (DL) models. In particular, it has been noticed that with a large number of features, manual approaches to feature engineering, which use all the available features, demonstrated higher accuracy compared to deep learning models, where feature elimination resulted in the loss of significant features. Furthermore, the efficiency of feature selection in ANNs was lower as compared to that of the ML models which have resulted in the enhanced performance of XGBoost over DL models. The study believes that the NeuroPred method, which is proposed, will be able to meet these requirements and will become a valuable high-throughput, cost-effective approach to large-scale analysis, and timely identification of therapeutic neuropeptides. This approach is multiple feature selection and screens important features besides developing a more reliable model by fusing eight base classifiers with four feature selection methods compared to the current state of the art models.

REFERENCES

[1] Hasan, M. M., Alam, M. A., Shoombuatong, W., Deng, H. W., Manavalan, B., ... Kurata, H. (2021). NeuroPred-FRL: An interpretable prediction model for neuropeptide identification using feature representation learning. *Briefings in Bioinformatics*, 22(6), bbab167.
Bin, Y., Zhang, W., Tang, W., Dai, R., Li, M., Zhu, Q., & Xia, J. (2020). Prediction of neuropeptides from sequence information using ensemble classifier and hybrid features. *Journal of Proteome Research*, 19(9), 3732–3740.

[2] Jiang, M., Zhao, B., Luo, S., Wang, Q., Chu, Y., Chen, T., Mao, X., Liu, Y., Wang, Y., Jiang, X., Wei, D.Q., & Xiong, Y. (2021). NeuroPred-Fuse: We propose an interpretable stacking model for prediction of neuropeptides by combining sequence information and feature selection methods. *Briefings in Bioinformatics*, 22(6), bbab310. https://doi.org/10.1093/bib/bbab310.

[3] Li, H., Jiang, L., Yang, K., Shang, S., Li, M., & Lv, Z. (2024). iNP_ESM: Evolutionary scale modeling and unified representation embedding features for neuropeptide identification. *International Journal of Molecular Sciences*, 25(13), 7049. https://doi.org/10.3390/ijms25137049.
Karsenty, S., Rappoport, N., Ofer, D., Zair, A. and Linial, M. (2014). NeuroPID: Nucleic Acids Research, 42(Web Server issue), W182–W186. https://doi.org/10.1093/nar/gku363 A classifier of neuropeptide precursors.
Wang, L., Huang, C., Wang, M., Xue, Z., and Wang, Y. (2023). NeuroPred-PLM: A neuropeptide prediction by protein language model, interpretable and robust. *Briefings in Bioinformatics*, 24(2), bbad077. https://doi.org/10.1093/bib/bbad077.

[4] Liu, D., Lin, Z., and Jia, C. (2023). NeuroCNN_GNB: An ensemble model to predict neuropeptides by a convolutional neural network and Gaussian naive Bayes, Frontiers in Genetics, 14, 1226905.
Wang, L., Huang, C., Wang, M., Xue, Z., & Wang, Y. (2023). NeuroPred-PLM: An interpretable and robust neuropeptide prediction by protein language model. *Briefings in Bioinformatics*, 24(2):1–9.
Liu, D., Lin, Z., & Jia, C. (2023). NeuroCNN_GNB: A convolution neural network and Gaussian naive Bayes ensemble model to predict neuropeptides. *Frontiers in Genetics*, 14:1226905. doi:10.3389/fgene.2023.1226905.
Jiang, M., Zhao, B., Luo, S., Wang, Q., Chu, Y., Chen, T., Mao, X., Liu, Y., Wang, Y., Jiang, X., Wei, D. Q., & Xiong, Y. (2023). NeuroPred-Fuse: We develop an interpretable stacking model for predicting neuropeptides by combining sequence information and feature selection techniques.
Yin, S., Mi, X., Shukla, D. (2023). Leveraging machine learning models for peptide–protein interaction prediction. doi:10.1039/d3cb00208j.
Wang, L., Zeng, Z., Xue, Z., & Wang, Y. (2023). A Deep NeuropePred: We propose a robust and universal tool to predict cleavage sites from neuropeptide precursors by protein language model.

[5] Van Bael, S., Watteyne, J., Boonen, K. et al. (2018). Mass spectrometric evidence for neuropeptide-amidating enzymes in *Caenorhabditis elegans*. *Journal of Biological Chemistry*, 293:doi:10.1074/jbc.RA117.000731.
Svensson, M., Skold, K., Svenningsson, P., et al.

[6] (2003). Peptidomics based discovery of novel neuropeptides. *Journal of Proteome Research*, 2:213–9. doi:10.1021/pr020010u.
Kormos, V. & Gaszner, B. (2013). Role of neuropeptides in anxiety, stress, and depression: All in the family: neuropeptides to humans. *Neuropeptides*, 47:401–19.

[7] Cai, W.
tyburski2017head Tyburski, A. L., Cheng, L., Assari, S., et al. (2017). Frequent mild head injury promotes trigeminal sensitivity concomitant with microglial proliferation, astrocytosis, and increased neuropeptide levels in the trigeminal pain system. *Journal of Headache and Pain*, 18:16.

[8] Carniglia, L., Ramirez, D., Durand, D.,... (2017). Neuropeptides and microglial activation in inflammation, pain, and neurodegenerative diseases. *Mediators of Inflammation*, 2017:5048616.
Vapnik, V. 2013. *The Nature of Statistical Learning Theory*. New York: Springer Science & Business Media.
Couvineau, A., Dayot, S., Nicole, P., et al.,

[9] . The anti-tumoral properties of orexin/hypocretin hypothalamic neuropeptides: An unexpected therapeutic role. *Frontiers in endocrinology (Lausanne)*, 9: 573.
H. Zeng, Y. Qin, E. Du, et al. in

[10] . Discovery of conserved and novel neuropeptides in the American cockroach by genomics and peptidomics.
Author:

[11] Che F. Y, Biswas R, Fricker L. D (2005). Relative quantitation of peptides in wild-type and Cpe(fat/fat) mouse pituitary using stable isotopic tags and mass spectrometry. *Journal of Mass Spectrometry*, 40:227–37.
Barson, J. R. (2020). The role of neuropeptides in drug and ethanol abuse: Drug and alcohol use disorders. *Brain Research*, 1740:146876.

[12] Boonen et al. (2008). Peptidomics: The integrated approach of MS, hyphenated techniques and bioinformatics to neuropeptide analysis: *Journal of Separation Science*, 31:427–45.
Chapman, L. F., et al.

[13] A polypeptide formed in man during neuronal activity, known as neurokinin. Observations on the axon reflex and antidromic dorsal root stimulation. *Transactions of the American Neurological Association*, 85:42–5.
Bin, Y., Zhang, W., Tang, W. et al. (2020). Prediction of neuropeptides from sequence information using ensemble classifier and hybrid features. *Journal of Proteome Research*, 19:3732–40.

[14] J. Kang, Y. Fang, P. Yao, et al., (2019). NeuroPP: The prediction of neuropeptide precursors based on optimal sequence composition: a tool. *Interdisciplinary Sciences*, 11:108–14.
Karsenty, S., Rappoport, N., Ofer, D., et al.

[15] . NeuroPID: Nucleic Acids Research 42: W182–6 (2014, a classifier of neuropeptide precursors).
Kim, Y., Bark, S., Hook, V., et al.

[16] NeuroPedia: A neuropeptide database and spectral library. *Bioinformatics* 27:2772–3.
Wang, Y., Wang, M., Yin, S., et al.

[17] NeuroPep: *Database (Oxford)*, 2015:bav038, a comprehensive resource of the neuropeptides.
Hasan, M. M., Schaduangrat, N., Basith, S. et al. (2020). HLPpred-Fuse: Fused multiple feature representation leading to improved and better prediction of hemolytic peptide and its activity. *Bioinformatics*, 36(13):3350–6.

[18] Manavalan, B., Basith, S., Shin, T. H., et al. (2019). mAHTPred: A meta predictor based on sequence for better prediction of antihypertensive peptides using effective feature encoding. *Bioinformatics*, 35:2757 – 65.

[19] Boopathi, V., Subramaniyam, S., Malik, A., et al. (2019). mACPpred: Identification of anticancer peptides using a support vector machine based meta-predictor. *International Journal of Molecular Sciences*, 20.

[20] Xu, Z. C., Feng, P. M., Yang, H., et al. (2019). iRNAD: A computational tool to determine which RNA sequence sites in the genome can be D modified. *Bioinformatics*.

[21] Yang, H., Lv, H., Ding, H., et al. (2018). iRNA-2OM: A 2O*meth*-yation sites predictor based on sequence, *Journal of Computational Biology*, 25:1266–77.