Journal of Pharmaceutical Analysis 15 (2025) 101410

Contents lists available at ScienceDirect

Journal of Pharmaceutical Analysis

journal homepage: www.elsevier.com/locate/jpa

Original article

HyPepTox-Fuse: An interpretable hybrid framework for accurate peptide toxicity prediction fusing protein language model-based embeddings with conventional descriptors

Duong Thanh Tran [a,1], Nhat Truong Pham [a,1], Nguyen Doan Hieu Nguyen [a], Leyi Wei [b], Balachandran Manavalan [a,*]

[a] Department of Integrative Biotechnology, College of Biotechnology and Bioengineering, Sungkyunkwan University, Suwon, 16419, Republic of Korea
[b] Centre for Artificial Intelligence Driven Drug Discovery, Faculty of Applied Science, Macao Polytechnic University, Macao SAR, 999078, China

ARTICLE INFO

Article history:
Received 31 December 2024
Received in revised form
28 June 2025
Accepted 19 July 2025
Available online 24 July 2025

Keywords:
Peptide toxicity
Hybrid framework
Multi-head attention
Transformer
Deep learning
Machine learning
Protein language model

ABSTRACT

Peptide-based therapeutics hold great promise for the treatment of various diseases; however, their clinical application is often hindered by toxicity challenges. The accurate prediction of peptide toxicity is crucial for designing safe peptide-based therapeutics. While traditional experimental approaches are time-consuming and expensive, computational methods have emerged as viable alternatives, including similarity-based and machine learning (ML)-/deep learning (DL)-based methods. However, existing methods often struggle with robustness and generalizability. To address these challenges, we propose HyPepTox-Fuse, a novel framework that fuses protein language model (PLM)-based embeddings with conventional descriptors. HyPepTox-Fuse integrates ensemble PLM-based embeddings to achieve richer peptide representations by leveraging a cross-modal multi-head attention mechanism and Transformer architecture. A robust feature ranking and selection pipeline further refines conventional descriptors, thus enhancing prediction performance. Our framework outperforms state-of-the-art methods in cross-validation and independent evaluations, offering a scalable and reliable tool for peptide toxicity prediction. Moreover, we conducted a case study to validate the robustness and generalizability of HyPepTox-Fuse, highlighting its effectiveness in enhancing model performance. Furthermore, the HyPepTox-Fuse server is freely accessible at https://balalab-skku.org/HyPepTox-Fuse/ and the source code is publicly available at https://github.com/cbbl-skku-org/HyPepTox-Fuse/. The study thus presents an intuitive platform for predicting peptide toxicity and supports reproducibility through openly available datasets.

1. Introduction

Peptides, short chains of amino acids, are essential biomolecules with diverse biological functions [1]. They show great promise for treating diseases such as cancer, diabetes, and cardiovascular disorders [2,3]. When assessing peptides as therapeutic candidates, it is essential to consider not only their immunogenicity and stability but also their toxicity. Toxicity poses a significant challenge in drug development, because certain peptides exhibit toxic properties that limit their clinical application [4,5]. Consequently, the accurate prediction of peptide toxicity is critical for designing safe and effective peptide-based therapeutics.

Conventional approaches often depend on wet lab experiments, which are time-intensive, costly, and prone to variability due to experimental conditions and reproducibility issues [6]. These methods are becoming impractical with rapid increases in the number of potential peptide drugs due to advancements in sequencing technologies. To address this issue, advancements in computational biology have facilitated the development of methods to accelerate peptide toxicity screening. These methods can broadly be divided into two categories: similarity-based and machine learning (ML)/deep learning (DL)-based methods. Similarity-based methods, such as the basic local alignment search tool (BLAST) [7], compare a new sequence to known toxic peptides,

Peer review under responsibility of Xi'an Jiaotong University.
* Corresponding author.
E-mail address: bala2022@skku.edu (B. Manavalan).
[1] Both authors contributed equally to this work.

https://doi.org/10.1016/j.jpha.2025.101410

assuming that a high degree of sequence similarity indicates a likelihood of toxicity. However, these methods have limitations: they depend on the availability of homologous toxic peptides and encounter difficulties with large datasets. Additionally, their dependence on e-value cutoffs and predefined sequence similarity thresholds can compromise prediction accuracy (Acc).

By contrast, ML- and DL-based methods have gained prominence for predicting peptide toxicity. These approaches use complex algorithms to learn patterns from large datasets of peptides and their toxicity. Several tools have been developed for predicting peptide toxicity. Notably, DeepTox [8] predicts toxicity across 12 different toxic effects; ProTox-II [9] focuses on multistep toxicity prediction based on five categories: acute toxicity, organ toxicity, toxicological endpoints, toxicological pathways, and toxicity targets; and ToxinPred (1.0, 2.0, and 3.0) [10–12] predicts whether peptides are toxic or non-toxic. Specifically, BTXpred [13] and NTXpred [14] have focused on classifying bacterial toxins and neurotoxins. Additionally, tools such as ClanTox [15], SpiderP [16], TOXIFY [17], and ToxDL [18] focus on filtering out toxins from peptides derived from animals, making potential therapies safer. Advanced DL-based frameworks for peptide toxicity prediction include NNTox [19] based on gene ontology and neural networks, ToxMVA [20] based on deep autoencoder, ACPred-BMF [21] based on convolutional neural networks (CNNs) and gated recurrent units (GRUs), ATSE [22] based on graph neural networks (GNNs), and ToxGIN [23] based on graph isomorphism networks (GINs). Despite these advancements, the accurate prediction of peptide toxicity remains a challenge. Current methods have been reported to struggle with robustness and generalizability. Moreover, many existing approaches rely on a single type of feature representation, which limits their ability to comprehensively understand the complex nature of these biomolecules.

To address these challenges, we introduce HyPepTox-Fuse, a novel framework that enhances peptide toxicity prediction by combining protein language model (PLM)-based embeddings with conventional descriptors (Fig. 1). Our approach leverages embeddings derived from large pre-trained models to capture the contextual and sequential information embedded within peptide sequences. These embeddings are integrated using a cross-modal multi-head attention mechanism [24] to facilitate the interaction between each pair of PLM-based embeddings, and a Transformer architecture [24] is employed to effectively fuse all interacted features. In addition to PLM-based embeddings, our framework incorporates conventional descriptors, which are meticulously selected through a two-step process: feature ranking based on validated performance and feature selection based on a light gradient-boosting machine (LightGBM) [25]. The HyPepTox-Fuse framework performs superior cross-validation and testing on independent datasets, consistently outperforming existing methods that rely on single-feature modalities. The key contributions of this study are as follows:

- We introduce HyPepTox-Fuse, a novel framework that combines PLM-based embeddings from large pre-trained models with conventional descriptors, enabling richer and more comprehensive representations of peptide sequences.
- We employ a cross-modal multi-head attention mechanism and a Transformer architecture to effectively integrate and fuse features from multiple modalities, addressing the limitations of single-feature approaches.
- We design a robust feature-ranking and selection pipeline, ensuring that only the most relevant conventional descriptors are incorporated, thereby enhancing overall model performance.
- We demonstrate that HyPepTox-Fuse significantly outperforms existing methods in both cross-validation and independent dataset evaluations, achieving state-of-the-art performance in peptide toxicity prediction.
- We conduct a case study to validate the effectiveness and robustness of the proposed method, further demonstrating the generalizability of HyPepTox-Fuse to a new dataset.

Collectively, these contributions advance current knowledge on computational peptide toxicity analysis and pave the way for the development of reliable and scalable therapeutic peptides.

2. Materials and methods

2.1. Dataset

We utilized the benchmark dataset constructed by Rathore et al. [11], which was carefully curated from multiple reliable databases and datasets from previous studies, making it one of the most recent and comprehensive datasets. Toxic peptides were collected from different databases, including Conoserver [26], DRAMP 3.0 [27], CAMPR3 [28], dbAMP2.0 [29], YADAMP [30], DBAASP-v3 [31], UniProt [32], and SwissProt [33]. In contrast, non-toxic peptides were retrieved from UniProt using the search query "NOT toxin NOT toxic AND reviewed: yes". It must be noted that all curated peptides included in the dataset were limited to a maximum of 35 amino acids and excluded non-natural amino acids. Additionally, the Venny software [34] was utilized to remove identical peptides, resulting in 5,518 unique toxic peptides and 4,321 non-toxic peptides. However, because the dataset remained imbalanced, additional non-toxic peptides were generated from SwissProt to ensure equal negative (non-toxic) and positive (toxic) samples. The final dataset comprised 5,518 non-toxic and 5,518 toxic peptides. Subsequently, it was divided into two subsets: 80% for the training dataset (4,414 non-toxic and 4,414 toxic peptides) and 20% for the independent dataset (1,104 non-toxic and 1,104 toxic peptides). An overview of the dataset preparation process is illustrated in Fig. 1.

The original dataset notably lacked explicit information regarding peptide dosage and exposure duration, critical parameters for accurately evaluating peptide toxicity. To overcome this limitation, we manually curated experimental data for numerous peptides, wherever available, and some of them were evaluated across multiple biological targets. Fig. S1A illustrates the distribution of experimentally tested concentrations across various biological indices, including minimum inhibitory concentration (MIC), minimum bactericidal concentration (MBC), and inhibitory concentration (IC). Observed interquartile intervals and typical exposure times were: MIC: 9.38–100 μg/mL, 16–24 h; MBC: 12.06–128 μg/mL, 16–24 h; IC: 9.06–73.29 μg/mL, 24–72 h. The broad variability in dosage levels and exposure times highlights the context-dependent nature of peptide toxicity. These observations strongly suggest that toxicity is not solely an inherent property of peptides, but it is significantly influenced by experimental conditions, notably dose and exposure duration. Fig. S1B shows the number of reported experiments for each specific biological index, highlighting MIC as the most commonly reported metric with 9,292 instances. This amount underscores its prominence and utility in antimicrobial screening. Indices such as MBC and IC, although less frequently reported, remain essential for discerning the threshold of high efficacy or potential toxicity, thereby providing a more comprehensive understanding of peptide bioactivity.

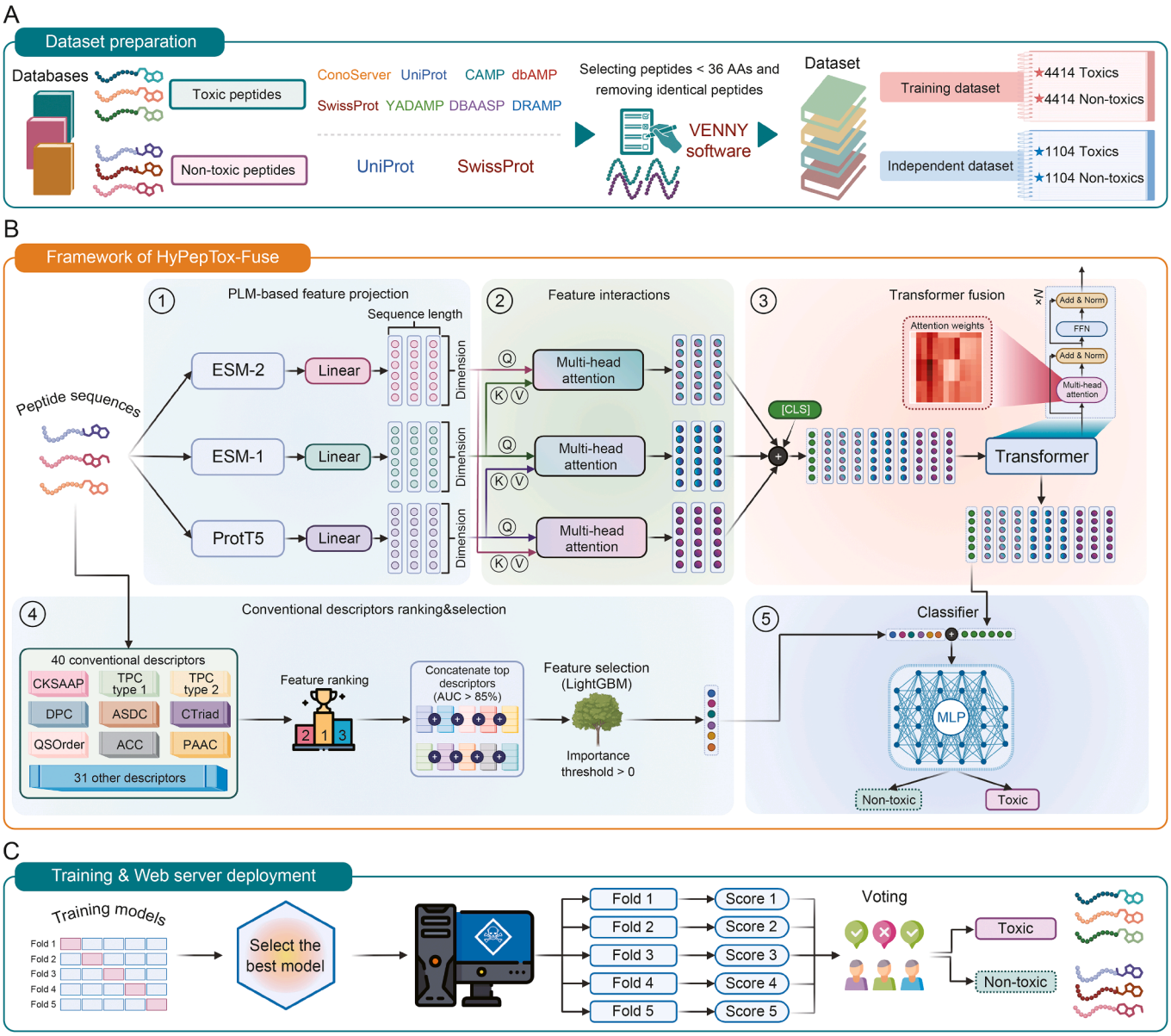

Fig. 1. Overview of the HyPepTox-Fuse framework. (A) The dataset, collected from multiple reliable databases, was preprocessed to include 4,414/4,414 non-toxic/toxic peptides for training and 1,104/1,104 non-toxic/toxic peptides for testing. (B) The overall architecture integrates a novel hybrid approach that combines protein language model (PLM)-based embeddings with conventional descriptors. (C) The model is trained using 5-fold cross-validation, and a voting method aggregates predictions from the five probability scores to classify peptides as non-toxic or toxic. CAMP: collection of antimicrobial peptides; dbAMP: database of anti-microbial peptides; YADAMP: yet another database of antimicrobial peptides; DBAASP: database of antimicrobial activity and structure of peptides; DRAMP: data repository of antimicrobial peptides; AAs: amino acids; ESM: evolutionary scale modeling; Q: query; K: key; V: value; CLS: classification; Add: addition; Norm: normalization; FFN: feed-forward network; CKSAAP: composition of k-spaced amino acid pairs; TPC: tripeptide composition; DPC: dipeptide composition; ASDC: adaptive skip dipeptide composition; CTriad: conjoint triad; QSOrder: quasi-sequence-order descriptors; ACC: auto cross-covariance; PAAC: pseudo-amino acid composition; AUC: area under the receiver operating characteristic curve; LightGBM: light gradient-boosting machine; MLP: multilayer perceptron. Created with BioRender.com and icons from Flaticon.com.

2.2. Feature extraction

To capture the relationships within peptide sequences, we employed five pre-trained PLMs, evolutionary scale modeling (ESM)-1 [35], ESM-2 [36], and three variants of ProtTrans [37] (ProtT5, ProtBERT, and ProtALBERT), which have shown superior performance in recent protein and peptide-related research [38–44]. A detailed description of these PLM-based embeddings can be found in previous studies [42–44], and details of the versions of the pre-trained models used in this study are provided in Table S1 [35–37].

Additionally, we utilized 40 conventional descriptors extracted from iFeatureOmega [45], including amino acid composition, grouped amino acid composition, quasi-sequence-order, and pseudo-amino acid composition. Detailed descriptions of these conventional descriptors can be found in previous studies [42,43,46], and detailed information on each descriptor is provided in Table S2 [45].

2.3. Framework of HyPepTox-fuse

2.3.1. Preliminary

To effectively fuse information from different PLM-based embeddings, our HyPepTox-Fuse framework utilizes the multi-head attention (MHA) mechanism [24] and Transformer architecture [24], as illustrated in "Framework of HyPepTox-Fuse" in Fig. 1. The

MHA mechanism facilitates the fusion of each pair of PLM-based embeddings by learning the cross-relationships between them. Transformer architecture excels at capturing long-range dependencies and complex relationships within sequential data, making it well suited for analyzing peptide sequences and integrating information from these fused embeddings. This section provides an overview of these core components.

2.3.1.1. MHA mechanism. In our framework, the MHA mechanism performs the role of cross-attention to facilitate the fusion of information between each pair of PLM-based embeddings. This enables the model to selectively attend to relevant information from one embedding while processing the other, enabling a more nuanced integration of their respective representations. This mechanism can be summarized as follows:

$$\mathrm{MHA}(Q,K,V)=\mathrm{concat}([\mathrm{head}_1,\mathrm{head}_2,\ldots,\mathrm{head}_h])W_O, \tag{1}$$

$$\begin{aligned}\mathrm{head}_i&=\mathrm{Attention}(Q_iW_{Q_i},K_iW_{K_i},V_iW_{V_i})\\&=\mathrm{softmax}\left(\frac{(Q_iW_{Q_i})(K_iW_{K_i})^T}{\sqrt{d_{\mathrm{model}}/h}}\right)(V_iW_{V_i}).\end{aligned} \tag{2}$$

Here, Q, K and V are the query, key, and value matrices, respectively, derived from the input features of two different PLM-based embeddings. Specifically, when fusing embedding A and embedding B, Q is derived from embedding A, whereas K and V are derived from embedding B. This allows the attention mechanism to learn the extent to which each element in embedding B should be attended to when processing embedding A. W_O, W_{Q_i}, W_{K_i}, and W_{V_i} refer to the learnable weight matrices that project the inputs into the appropriate spaces. The dimensionality of the model is denoted as d_{model}, and h represents the number of attention heads.

2.3.1.2. Transformer architecture. The Transformer architecture is a revolutionary model that utilizes the MHA mechanism and parallel processing to efficiently capture contextual relationships in sequences, thereby replacing traditional recurrent and convolutional neural networks. It features an encoder-decoder structure, where the encoder encodes input features into representations, and the decoder generates outputs from these representations. In this framework, we focus solely on the encoder component of the Transformer to represent features. The Transformer encoder can be summarized in three steps, as follows:

First, the input sequence X is first transformed into an initial embedding:

$$H_0=\mathrm{Embed}(X) \tag{3}$$

Second, the encoder consists of N_{TE} layers, where each layer performs two key operations: 1) MHA with a residual connection, and 2) feed-forward network (FFN) with another residual connection:

$$H_l'=\mathrm{LN}(H_{l-1}+\mathrm{MHA}(H_{l-1})),\forall l\in\{1,\ldots,N_{\mathrm{TE}}\}, \tag{4}$$

$$H_l=\mathrm{LN}(H_l'+\mathrm{FFN}(H_l')). \tag{5}$$

Finally, after passing through all N_{TE} layers, the output of the last layer represents the final encoded representation:

$$\mathrm{TE}(X)=H_{N_{\mathrm{TE}}} \tag{6}$$

Each encoder layer applies residual connections to stabilize training and improve gradient flow. It adds the output of MHA to the input from the previous layer and applies layer normalization (LN). Next, it passes this result through an FFN, adds another residual connection, and applies LN again. These operations ensure efficient learning while maintaining stability.

2.3.2. Workflow of HyPepTox-Fuse

As illustrated in Fig. 1, the HyPepTox-Fuse framework consists of four main modules: 1) PLM-based embedding projection, 2) feature interactions (FIs), 3) Transformer fusion (TF), 4) conventional descriptors ranking and selection, and 5) classifier. The details of these modules are presented below:

2.3.2.1. PLM-based embedding projection. Assume that the extracted features from the three best pre-trained models are denoted as $A_1\in\mathbb{R}^{L\times d_{A_1}}$, $A_2\in\mathbb{R}^{L\times d_{A_2}}$, and $A_3\in\mathbb{R}^{L\times d_{A_3}}$, respectively, where L is the length of the sequence embeddings, and d_{A_1}, d_{A_2}, and d_{A_3} are the embedding dimensions of the corresponding features. As A_1, A_2, and A_3 differ in their dimensions, we projected all features onto the same dimension d_{PLM}, as follows:

$$A_i^{\mathrm{prj}}=\mathrm{Linear}(A_i),\forall i\in\{1,2,3\},A_i^{\mathrm{prj}}\in\mathbb{R}^{L\times d_{\mathrm{PLM}}} \tag{7}$$

2.3.2.2. FIs. The FIs module aims to obtain the interacted signals across different PLM-based embeddings. Every set of two features would constitute a pair of inputs. Here, we define the pairs as a set, $P=\{(1,2),(2,3),(3,1)\}$. We use the MHA mechanism to obtain each interacted feature of the three pairs, denoted as $\mathrm{FI}_{(i,j)}$, which can be summarized as follows:

$$\mathrm{FI}_{(i,j)}=\mathrm{MHA}\left(A_i^{\mathrm{prj}},A_j^{\mathrm{prj}},A_j^{\mathrm{prj}}\right),\forall(i,j)\in P,\mathrm{FI}_{(i,j)}\in\mathbb{R}^{L\times d_{\mathrm{PLM}}} \tag{8}$$

2.3.2.3. TF. After obtaining three pairs of interacted feature outputs $\mathrm{FI}_{(i,j)}$, we concatenate them into a single sequence with the length of $3L$ and append a learnable special embedding ([CLS]) at the beginning of the sequence. The concatenated features can be represented as follows:

$$\mathrm{FI}_{\mathrm{cc}}=\mathrm{concat}\left(\left[[\mathrm{CLS}],\mathrm{FI}_{(i,j)}\middle|(i,j)\in P\right]\right),\mathrm{FI}_{\mathrm{cc}}\in\mathbb{R}^{(3L+1)\times d_{\mathrm{PLM}}} \tag{9}$$

This special embedding is inspired by the BERT model [47], which is added at the beginning of the input sequence during pre-training to represent the entire sequence in downstream tasks, particularly for classification. We also analyzed this approach by comparing it with the averaging features approach described in Section 3.5.4.

Finally, we fed $\mathrm{FI}_{\mathrm{cc}}$ through a Transformer encoder to learn the interaction of all sequences. This resulted in a new representation, $\tilde{\mathrm{FI}}_{\mathrm{cc}}$, as follows:

$$\tilde{\mathrm{FI}}_{\mathrm{cc}}=\mathrm{TE}(\mathrm{FI}_{\mathrm{cc}}),\mathrm{FI}_{\mathrm{cc}}\in\mathbb{R}^{(3L+1)\times d_{\mathrm{PLM}}} \tag{10}$$

2.3.2.4. Conventional descriptors ranking and selection. We built a simple neural network to rank the 40 conventional descriptors for training and evaluating each single-feature model. We then concatenated features with area under the receiver operating characteristic (ROC) curve (AUC) > 0.85 on cross-validation and denoted these concatenated conventional descriptors (CCDs) as Z. While the dimensionality of Z is high and complex, this can lead to overfitting and increased computational costs. To address this issue, we used LightGBM [25] to select the most relevant features, with a threshold for the importance score greater than zero. After

this process, we obtained optimized CCDs, denoted as $\tilde{Z} \in \mathbb{R}^{d_{conv}}$. To evaluate the contribution of these features to final model performance, we conducted experiments by gradually adding top-ranked CCDs, which are presented in detail in Section 3.3.

2.3.2.5. Classifier. In this module, we concatenate the optimized CCDs $\tilde{Z}$ and the special embedding ([CLS]) to produce the concatenated feature C:

$$C = \text{concat}\left(\tilde{Z}, \widetilde{FI}_{cc,0}\right), C \in \mathbb{R}^{(d_{conv}+d_{PLM})} \tag{11}$$

This final feature is subsequently passed through a multilayer perceptron (MLP) to produce logits for classifying toxic or non-toxic peptides:

$$\tilde{C} = \text{MLP}(C) \tag{12}$$

2.3.3. Optimization process of HyPepTox-Fuse

We carried out a comprehensive optimization process to select the best model for constructing HyPepTox-Fuse. During training, we employed stratified 5-fold cross-validation to tune the hyperparameters within a predefined search range, incorporating stratified sampling instead of random selection. The model weights were optimized using the AdamW optimizer [48] and a learning rate scheduler was employed to reduce the learning rate by a specific ratio after a predefined number of epochs. Further details regarding the hyperparameter search range can be found in Table S3.

In this study, we utilized two loss functions: binary cross-entropy loss (L_{BCE}) for supervised learning and normalized temperature-scaled cross-entropy loss (L_{NTXent}) [49,50] for contrastive learning. The L_{BCE} served as the primary objective function for optimizing the model parameters, whereas the L_{NTXent} acted as an auxiliary function to enhance the model's ability to distinguish between positive and negative samples. Specifically, L_{NTXent} helped the model bring the representations of semantically similar samples (positive pairs) closer together in a shared embedding space, and ensured that unrelated samples (negative pairs) were pushed apart. The loss functions are defined as follows:

$$L_{BCE} = \frac{1}{N}\sum_{i=1}^{N}[y_i \log \hat{y}_i + (1-y_i)\log(1-\hat{y}_i)], \tag{13}$$

$$L_{NTXent} = -\frac{1}{2N}\sum_{i=1}^{N}\left[\log \frac{\exp\left(\text{sim}\left(z_i, z_j\right)/\tau\right)}{\sum_{k=1}^{2N} 1_{[k\neq i]} \exp(\text{sim}(z_i, z_k)/\tau)}\right]. \tag{14}$$

Here, N is the number of samples, y_i is the ground-truth label, $\hat{y}_i$ is the predicted probability for the positive class, $\text{sim}(*,*)$ denotes the cosine similarity between two representations, τ is the temperature scaling parameter, and $1_{[k\neq i]} \in \{0,1\}$ is an indicator function evaluating to 1 if $k \neq i$.

To control the contributions of L_{BCE} and L_{NTXent}, we introduced a weighted combination of the two losses, controlled by their respective weights. The total loss function is defined as:

$$L_{total} = \alpha \times L_{BCE} + (1-\alpha) \times L_{NTXent} \tag{15}$$

This approach ensures that the primary focus remains on minimizing the supervised binary cross-entropy loss, whereas the contrastive loss provides auxiliary support to improve feature discrimination.

2.4. Performance evaluation metrics

In this study, several commonly used performance evaluation metrics were utilized to evaluate the model's performance during training, independent testing, and comparison. These metrics included sensitivity (SEN), specificity (SPE), Acc, F1-score (F1), and Matthews correlation coefficient (MCC). The mathematical equations for these metrics are as follows:

$$\text{SEN} = \frac{\text{TP}}{\text{TP}+\text{FN}}, \tag{16}$$

$$\text{SPE} = \frac{\text{TN}}{\text{TN}+\text{FP}}, \tag{17}$$

$$\text{Acc} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}, \tag{18}$$

$$\text{F1} = \frac{2\times\text{TP}}{2\times\text{TP}+\text{FP}+\text{FN}}, \tag{19}$$

$$\text{MCC} = \frac{\text{TP}\times\text{TN}-\text{FP}\times\text{FN}}{\sqrt{(\text{TP}+\text{FP})\times(\text{TP}+\text{FN})\times(\text{TN}+\text{FP})\times(\text{TN}+\text{FN})}}. \tag{20}$$

Here, true positives (TP) represents the number of predictions correctly identified as positive samples, true negatives (TN) represents the number of predictions correctly identified as negative samples, false positives (FP) stands for the number of predictions incorrectly identified as positive samples, and false negatives (FN) denotes the number of predictions incorrectly identified as negative samples. SEN measures the ability of a model to correctly identify positive samples, whereas SPE assesses its ability to correctly identify negative samples. Acc indicates the overall proportion of correct predictions across all samples. F1 balances precision and SEN, offering a single metric that accounts for both false positives and false negatives. Lastly, MCC provides a comprehensive measure of classification quality, considering all four categories of the confusion matrix.

Additionally, we utilized AUC to summarize the model's ability to distinguish between classes across all decision thresholds. The formula for AUC is as follows:

$$\text{AUC} = \sum_{i=1}^{n-1}(\text{FPR}_{i+1}-\text{FPR}_i)\cdot\frac{\text{TPR}_i+\text{TPR}_{i+1}}{2}, \tag{21}$$

$$\text{FPR}_i = \frac{\text{FP}_i}{\text{FP}_i+\text{TN}_i}, \tag{22}$$

$$\text{TPR}_i = \frac{\text{TP}_i}{\text{TP}_i+\text{FN}_i}. \tag{23}$$

Here, FP rate (FPR_i) and TP rate (TPR_i) quantify the proportion of negative samples incorrectly classified and positive samples correctly classified as positive at a specific threshold i, and n is the total number of thresholds evaluated.

3. Results and discussion

3.1. Performance assessment of conventional descriptors

Initially, we extracted 40 conventional descriptors using the iFeatureOmega toolkit [45] and constructed baseline models using a deep neural network architecture. Fig. 2 highlights the performance of the top 13 descriptors with an AUC > 0.85 based on 5-

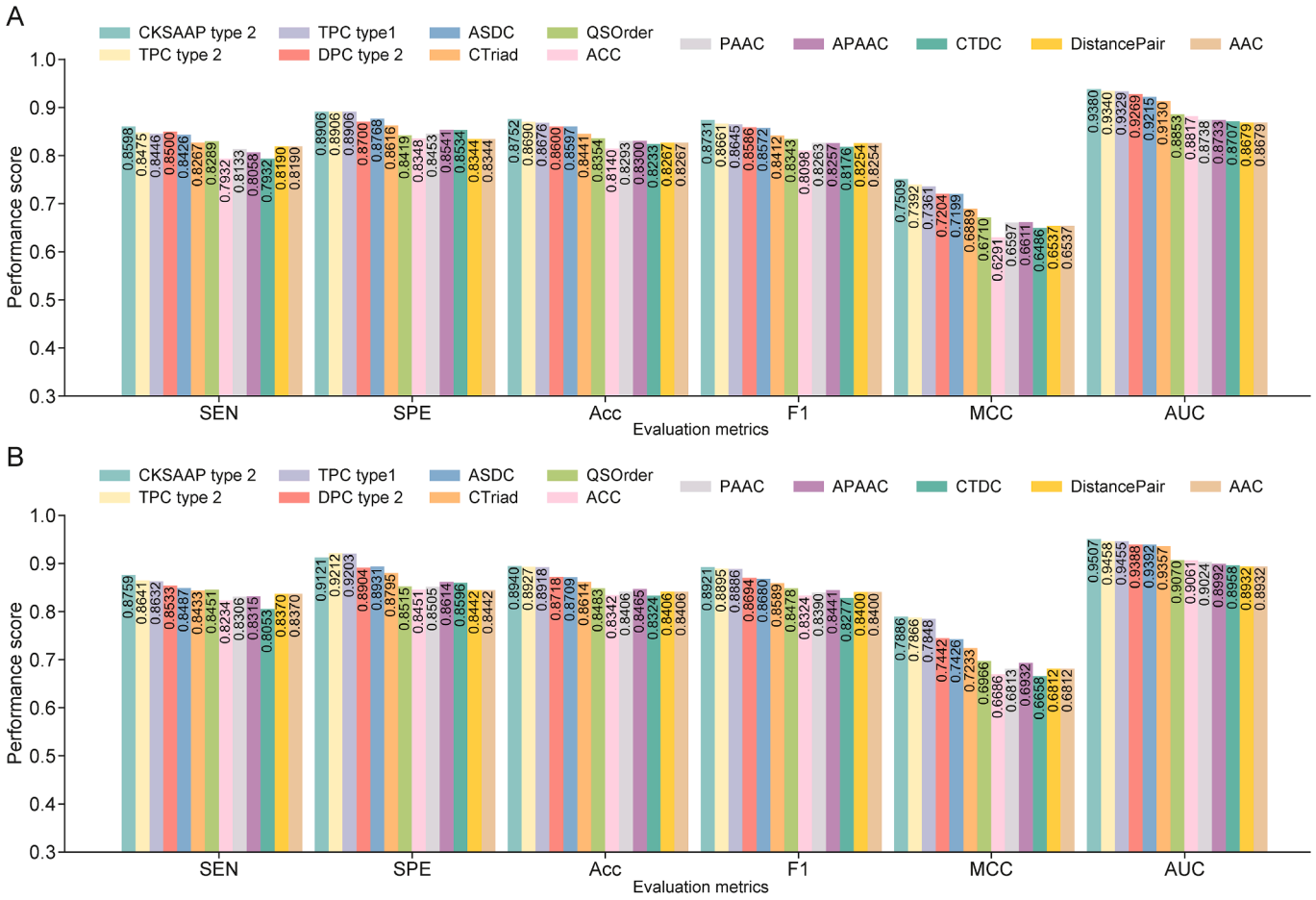

Fig. 2. Performance comparison between conventional descriptors with area under the receiver operating characteristic curve (AUC) > 0.85. (A) Performance evaluation on cross-validation datasets. (B) Performance evaluation on the independent dataset. CKSAAP: composition of k-spaced amino acid pairs; TPC: tripeptide composition; ASDC: adaptive skip dipeptide composition; QSOrder: quasi-sequence-order descriptors; PAAC: pseudo-amino acid composition; APAAC: amphiphilic pseudo-amino acid composition; CTDC: composition; DistancePair: pseudo-amino acid composition of distance-pairs and reduced alphabet; AAC: amino acid composition; DPC: dipeptide composition; CTriad: conjoint triad; ACC: auto cross-covariance; SEN: sensitivity; SPE: specificity; Acc: accuracy; F1: F1-score; MCC: Matthews correlation coefficient.

fold cross-validation (see Tables S4 and S5 for comprehensive results). Notably, the composition of k-spaced amino acid pairs (CKSAAP) type 2 achieved the highest performance across all key metrics, with an MCC of 0.7509, Acc of 0.8752, F1 of 0.8731, and AUC of 0.9380 in cross-validation, and an MCC of 0.7886, Acc of 0.8940, F1 of 0.8921, and AUC of 0.9507 in independent evaluation. Interestingly, tripeptide composition (TPC) types 1 and 2, dipeptide composition (DPC) type 2, and adaptive skip DPC (ASDC) also achieved competitive performance, with MCC values above 0.7200 and 0.7400 in cross-validation and independent evaluations, respectively. The remaining features among the top 13 showed moderate performance in both cross-validation and independent evaluations. However, they consistently exhibited AUC values greater than 0.8500, demonstrating their potential to capture peptide toxicity. Consequently, we selected the top 13 conventional descriptors for further analysis. These results suggest that CKSAAP type 2 is the most robust feature descriptor, but integrating multiple descriptors could further enhance classification performance by leveraging their complementary strengths. It is important to note that integrating all 40 conventional descriptors would result in an excessively high-dimensional input space, introducing significant redundancy and noise while significantly increasing computational demands. To overcome this, the HypepTox-Fuse framework strategically retains only the top 13 most informative features identified through baseline performance evaluation, thereby optimizing input representation, reducing computational costs, and enhancing predictive performance.

3.2. Performance evaluation of PLM-based embeddings

We evaluated the performance of different pre-trained PLM-based embeddings, including ESM-1 [35], ESM-2 [36], and three variants of ProtTrans [37]. As shown in Fig. S2, ProtT5 achieved the highest performance in cross-validation evaluation, followed by ESM-1 and ESM-2, which showed competitive performance. ProtT5 obtained Acc, F1, MCC, and AUC values of 0.8811, 0.8787, 0.7630, and 0.9335, respectively. This indicates its superior ability to capture meaningful peptide sequence features for classification. Additionally, ESM-1 and ESM-2 demonstrated balanced performances in terms of SEN and SPE, making them reliable choices for feature extraction. In contrast, ProtBERT and ProtALBERT exhibited lower performances, with MCC, Acc, F1, and AUC values of 0.7000, 0.8498, 0.8488, and 0.9014, and 0.7413, 0.8700, 0.8668, and 0.8925, respectively. These results indicate that ProtBERT and ProtALBERT are less effective in capturing the nuanced features required for accurate classification. To validate the robustness and effectiveness of these embeddings, we assessed their transferability to an independent dataset. Interestingly, ProtT5, ESM-1, and ESM-2 consistently outperformed ProtBERT and ProtALBERT, achieving

performances comparable to those observed during cross-validation. Based on this evaluation, ProtT5, ESM-1, and ESM-2 were selected as the top three PLM-based embeddings for integration into the HyPepTox-Fuse framework. Their complementary strengths are expected to enhance the overall performance and reliability of the framework, as discussed in Section 3.3.

3.3. Performance evaluation of hybrid PLM-based embeddings and CCDs

HyPepTox-Fuse integrates the best-performing PLM-based embeddings (ProtT5, ESM-1, and ESM-2) and conventional descriptors to classify peptides as toxic or non-toxic. Specifically, we first concatenated the top-ranked descriptors with an AUC > 0.85 to identify the most informative conventional descriptors, including ASDC, auto cross-covariance (ACC), CKSAAP type 2, TPC types 1 and 2, conjoint triad (CTriad), quasi-sequence-order descriptors (QSOrder), composition (CTDC), amphiphilic pseudo-amino acid composition (APAAC), pseudo-amino acid composition (PAAC), and pseudo-amino acid composition of distance-pairs and reduced alphabet (DistancePair). We then used LightGBM [25] to rank these features based on importance scores greater than zero, resulting in 887-dimensional (887-D) CCDs. Fig. S3 details the importance scores of each contributed feature in the CCDs and the total scores of the top-ranked CCDs. Next, we employed a sequential forward search to generate nine subsets of feature dimensions, starting from 100-D to the total dimensional features, with an interval of 100-D. Table S6 provides the details of the feature dimensions of each descriptor that contributed to each subset. These feature subsets were then integrated with the top PLM-based embeddings to identify the optimal model for determining peptide toxicity. Figs. 3 and 4 illustrate the performance metrics and ROC curves from both cross-validation and independent evaluations.

The results showed that integrating the top CCDs with the best PLM-based embeddings improved model performance, especially as the feature dimensions of the top CCDs significantly increased during cross-validation evaluation (Fig. 3). The model using the best PLM-based embeddings with the top 887-D CCDs achieved the highest performance, with MCC, Acc, F1, and AUC values of 0.8036, 0.9016, 0.9003, and 0.9443, respectively. Compared with the other models, it obtained notable improvements of 1.31%–3.91% in MCC, 0.66%–1.95% in Acc, 0.70 %–2.01% in F1, and 0.43%–2.79% in AUC. Furthermore, the zoomed-in ROC curves (Fig. 4) clearly show the performance gap between the model using only the best PLM-based embeddings and that using the best PLM-based embeddings with all 887-D CCDs, indicating that using only PLM-based embeddings limits precision in this critical range. Therefore, we selected the model using the top 887-D CCDs and best PLM-based embeddings as the final model for predicting peptide toxicity, named HyPepTox-Fuse.

HyPepTox-Fuse performance decreased slightly during independent evaluation. Notably, the model using only the best PLM-based embeddings achieved the highest performance in terms of

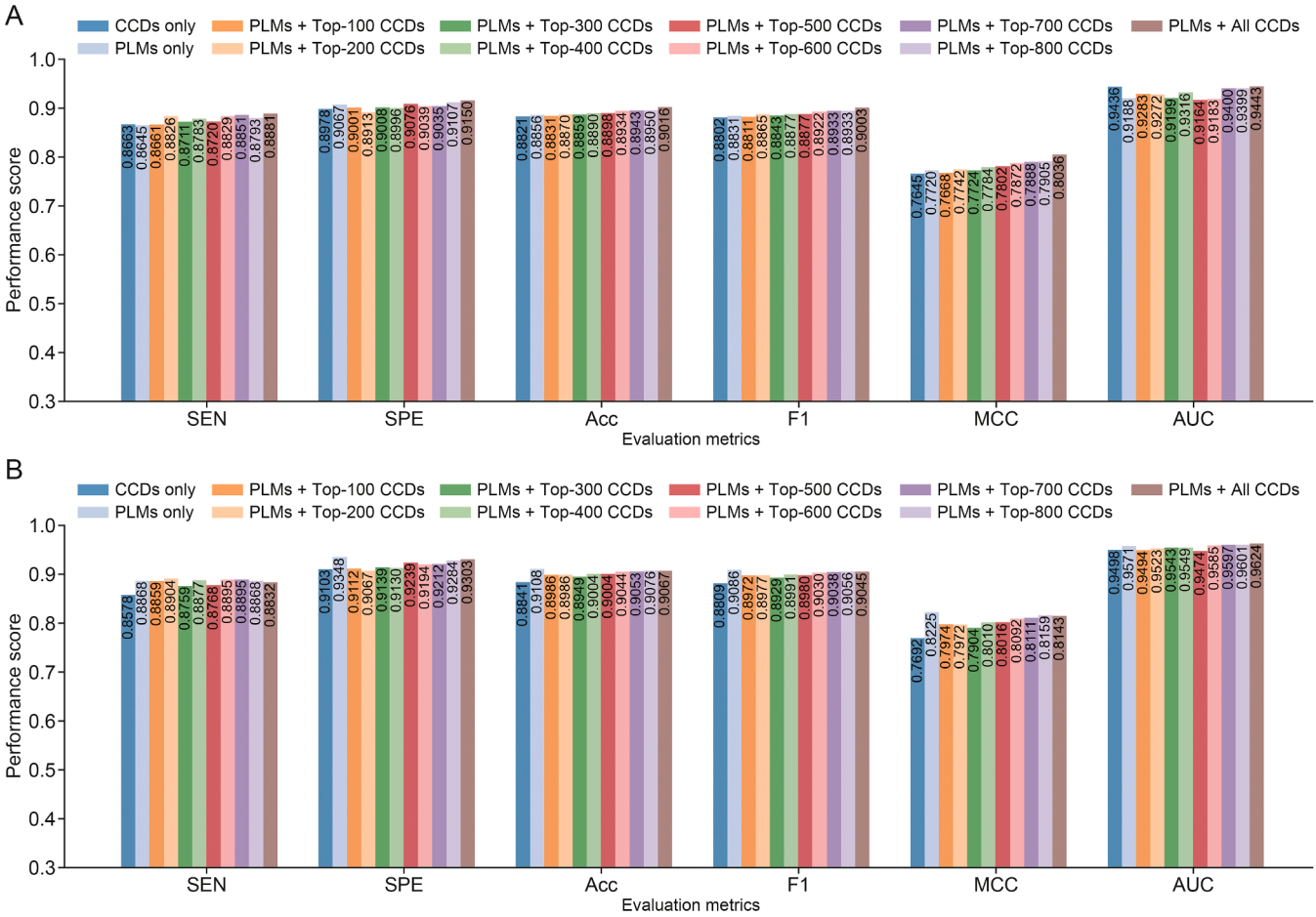

Fig. 3. Performance comparison between experiments when combining protein language model (PLM)-based embeddings with multiple top concatenated conventional descriptors (CCDs). (A) Performance evaluation on 5-fold cross-validation dataset. (B) Performance evaluation on the independent dataset. SEN: sensitivity; SPE: specificity; Acc: accuracy; F1: F1-score; MCC: Matthews correlation coefficient; AUC: area under the receiver operating characteristic curve.

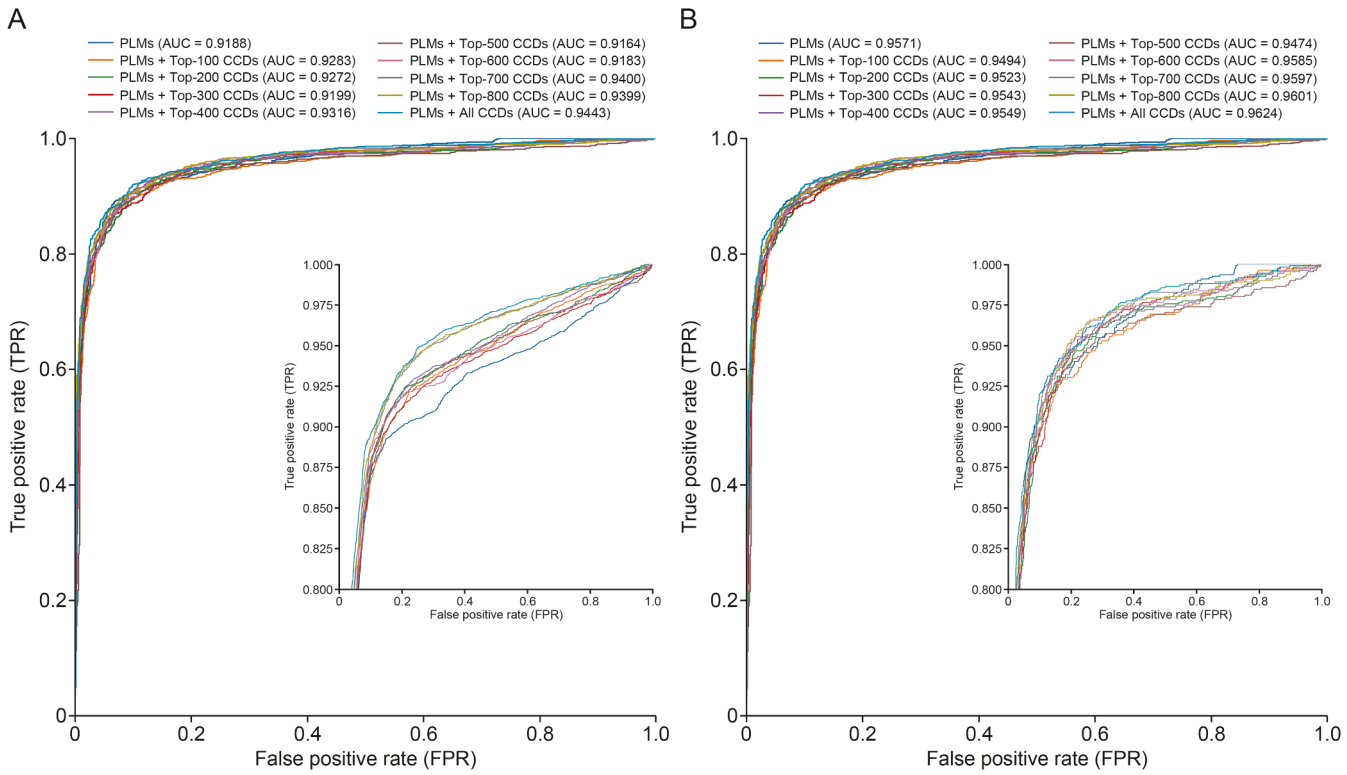

Fig. 4. Receiver operating characteristic (ROC) curves visualizations of multiple experiments. (A) ROC curves on cross-validation evaluation. (B) ROC curves on independent evaluation. The insets indicate a zoom-in range where the true positive rate is greater than 0.8. TPR: true positive rate; FPR: false positive rate; PLMs: protein language models; CCDs: concatenated conventional descriptors; AUC: area under the receiver operating characteristic curve.

MCC. HyPepTox-Fuse ranked third overall, with MCC, Acc, F1, and AUC values of 0.8143, 0.9067, 0.9045, and 0.9624, respectively. However, when considering the AUC values, HyPepTox-Fuse ranked first, demonstrating the model's ability to effectively generalize to unseen data. These findings emphasize the importance of selecting the most informative features to achieve a balance between model performance and computational efficiency. This approach not only enhances peptide toxicity prediction, but also provides valuable insights for advancing future models in this field.

3.4. Performance comparison with existing methods on the same training and independent datasets

To assess the performance of HyPepTox-Fuse relative to existing methods, we compared it with ToxinPred 3.0, as both frameworks were trained on the same dataset. Fig. S4 illustrates the performance comparison between the two variants of HyPepTox-Fuse (with and without CCDs) and two variants of ToxinPred 3.0 (ML and DL) [11]. It must be noted that we selected the final models for each of the ML and DL approaches from a prior study [11]. ToxinPred 3.0 (ML) exhibited the lowest performance on the training dataset, with MCC, Acc, and F1 values of 0.7440, 0.8710, and 0.8540, respectively. Interestingly, ToxinPred 3.0 (DL) demonstrated superior performance compared with ToxinPred 3.0 (ML) and HyPepTox-Fuse (PLMs only) in terms of MCC, suggesting its capacity to capture both local and global contextual information to identify peptide toxicity. However, HyPepTox-Fuse (PLMs + All CCDs) achieved the highest performance on the training dataset and significantly outperformed ToxinPred 3.0 models. Compared with ToxinPred 3.0 models, HyPepTox-Fuse exhibited notable improvements of 2.36%–5.96% in MCC, 3.06%–5.16% in Acc, and 4.63%–12.03% in F1. These findings highlight the effectiveness of HyPepTox-Fuse in capturing both local and global contextual information from peptide sequences, facilitating discrimination between toxic and non-toxic peptides by leveraging PLM-based embeddings and CCDs.

We compared HyPepTox-Fuse with widely used peptide toxicity prediction tools, including CSM-Toxin [51], ToxIBLT [52], and three versions of ToxinPred (1.0, 2.0, and 3.0) [10–12]. Table 1 shows that HyPepTox-Fuse outperformed all the existing peptide toxicity prediction tools on the same independent dataset. It should be noted that the performance metrics of ToxIBLT and ToxinPred 1.0 were obtained from ToxinPred 3.0 [11], whereas those of the other tools were obtained via their web servers. CSM-Toxin performed poorly on the independent dataset, with a performance similar to that of the random classifier (Acc = 0.5041). ToxinPred 2.0 performed relatively better than CSM-Toxin, showing a strong bias toward predicting toxic peptides (high SEN and low SPE). Interestingly, ToxIBLT, ToxinPred 1.0, and ToxinPred 3.0 (ML + Motif-EmeRging and with Classes-Identification) showed improved performance, with MCC and Acc values exceeding 0.6000 and 0.8000, respectively. ToxinPred 3.0 (ML) achieved the highest performance among the existing methods, with MCC, Acc, F1, and AUC values of 0.7788, 0.8890, 0.8865, and 0.9502, respectively. However, HyPepTox-Fuse outperformed all the existing tools on the same independent dataset, with MCC, Acc, F1, and AUC values of 0.8143, 0.9067, 0.9045, and 0.9624, respectively. Compared with existing tools, HyPepTox-Fuse achieved notable improvements of 3.55%–79.97% in MCC, 1.77%–40.26% in Acc, 1.80%–75.27% in F1, and 1.22%–47.19% in AUC. Moreover, McNemar's test results showed that HyPepTox-Fuse outperformed other methods with statistical significance. These findings highlight its ability to effectively balance the detection of toxic and

Table 1
Performance comparison between HyPepTox-Fuse and existing methods on the ToxinPred 3.0 independent dataset.

Model	SEN	SPE	Acc	F1	MCC	AUC	*P*-value
CSM-Toxin	0.0888	0.9194	0.5041	0.1518	0.0146	0.4905	<0.00001
ToxIBLT	0.7000	0.8900	0.8700	0.7513	0.6100	0.8000	NA
ToxinPred 1.0	0.6500	**0.9800**	0.8200	0.7098	0.6700	0.8500	NA
ToxinPred 2.0	**0.9275**	0.5172	0.7224	0.7696	0.4877	0.8589	<0.00001
ToxinPred 3.0 (ML)	0.8668	0.9112	0.8890	0.8865	0.7788	0.9502	0.00309
ToxinPred 3.0 (DL)	0.8659	0.8170	0.8415	0.8453	0.6838	0.9181	<0.00001
ToxinPred 3.0 (ML + MERCI)	0.7228	0.9330	0.8279	0.8077	0.6708	0.8998	<0.00001
ToxinPred 3.0 (DL + MERCI)	0.7074	0.8868	0.7971	0.7771	0.6040	0.8767	<0.00001
HyPepTox-Fuse	0.8832	0.9303	**0.9067**	**0.9045**	**0.8143**	**0.9624**	–

NA: not available; –: no data. The machine learning (ML) model is based on the extra tree (ET) classifier. The deep learning (DL) model is based on an artificial neural network combined with a long short-term memory network (ANN-LSTM). The *P*-value between HyPepTox-Fuse and other methods was calculated using the McNemar's test. The bold data indicate the best performance among the compared methods. SEN: sensitivity; SPE: specificity; Acc: accuracy; F1: F1-score; MCC: Matthews correlation coefficient; AUC: area under the receiver operating characteristic curve; MERCI: Motif-EmeRging and with Classes-Identification.

non-toxic peptides while maintaining high precision. Integrating advanced PLM-based embeddings and CCDs into the HyPepTox-Fuse framework contributed to its superior performance, allowing it to capture a broader range of relevant peptide characteristics. This makes HyPepTox-Fuse a promising tool for peptide toxicity prediction, offering substantial improvements over existing methods.

3.5. Model interpretation and analyses of HyPepTox-Fuse

3.5.1. Visualization of learned features

To understand how the different models learn and represent features, we visualized the learned feature representations derived from individual models (CCDs and PLM-based embeddings) and their combination (HyPepTox-Fuse). We employed uniform manifold approximation and projection (UMAP) [53], a novel dimensionality reduction technique, to transform the original features extracted by the models into 2-D vector representations. Fig. 5 illustrates the feature representations of the three models learned on the training and their predictions on independent datasets. Notably, the feature representations generated by the models using only PLM-based embeddings exhibited a greater degree of overlap than those utilizing CCDs. In contrast, HyPepTox-Fuse yielded feature representations with a clear separation between the toxic and non-toxic peptides. These findings suggest that HyPepTox-Fuse effectively captures discriminative information by leveraging both PLM-based embeddings and CCDs. Furthermore, the similarity between the feature representations extracted from the independent dataset and those from the training dataset indicates the robustness and transferability of the models. This points to their ability to effectively generalize to independent and external data, enhancing their applicability for peptide toxicity prediction.

3.5.2. Attention-based model interpretation

To understand how HyPepTox-Fuse makes its prediction, we analyzed the attention weights extracted from Transformer Fusion and compared them with sequence motifs identified using the multiple expectation maximizations for motif elicitation (MEME) tool [54]. The parameters used to reconstruct motifs via MEME are listed in Table S7 [54]. Fig. 6 shows attention heatmaps highlighting the regions of peptide sequences assigned high importance scores by the model during prediction. These regions strongly correspond to the biologically significant motifs identified by MEME, as evidenced by the matching of conserved residues across both methods. As shown in Fig. 6A, the attention-based approach emphasizes the “CCSNP” region, which closely aligns with the motif identified by MEME. Similar patterns are observed in Figs. 6B and C, where the attention mechanism highlights conserved regions such as “CCSG” and “LKDV,” which also align well with MEME-derived motifs and exhibit substantial biological relevance. The consistency between the attention scores and MEME motif analysis underscores both the interpretability and biological relevance of our attention-based model. HyPepTox-Fuse not only excels in classification but also effectively identifies critical sequence features that drive its predictions. The attention mechanism provides an additional layer of explainability by pinpointing the amino acid residues that contribute most significantly to the predictions, offering valuable insights to guide further biological studies, particularly in therapeutic peptide design.

3.5.3. Ablation study of loss function

As mentioned in Section 2.3.3, we utilized two loss functions: binary cross-entropy loss (L_{BCE}) and normalized temperature-scaled cross-entropy loss (L_{NTXent}). Notably, we used L_{NTXent} loss as an auxiliary loss function to improve feature discrimination. To assess its contribution, we conducted an ablation study by removing L_{NTXent} from the total loss function. It should be noted that we conducted this analysis based on the configuration of the final HyPepTox-Fuse model described in Section 3.3. Table S8 presents results of the performance comparison between the models trained with and without L_{NTXent}. The model using the L_{NTXent} auxiliary loss function consistently outperformed the model without it on both the training and independent datasets, with modest improvements of 0.73% in MCC, 0.38% in Acc, and 0.39% in F1, and 1.47%, 1.00%, and 0.72%, respectively. Although these improvements are relatively small, this finding suggests that utilizing L_{NTXent} loss function can enhance the overall performance of the final model. Additionally, utilizing contrastive learning along with supervised learning can improve the discriminative feature representations between toxicity and non-toxic peptides, leading to improved performance.

Moreover, to evaluate the impact of the temperature parameter (τ) of L_{NTXent} auxiliary loss function on model performance, we analyzed SEN, SPE, and Acc metrics during cross-validation over 100 epochs (Fig. S5). The zoomed-in visualization emphasizes the differences between the three distinct τ values: 0.1, 0.5, and 1.0. As shown in Fig. S5A, $\tau = 0.1$ and $\tau = 0.5$ yield superior SEN values compared to $\tau = 1.0$. After the 40th epoch, $\tau = 0.5$ slightly surpasses $\tau = 0.1$, but this advantage proves inconsistent in the later epochs. Fig. S5B presents the SPE metric, revealing a significant disparity between $\tau = 0.1$ and $\tau = 0.5$, suggesting that the model using $\tau = 0.5$ may exhibit a bias toward negative samples (non-toxic peptides), which is undesirable for achieving a balanced performance. This conclusion is further supported by the analysis of the Acc metric depicted in Fig. S5C. After the fourth epoch,

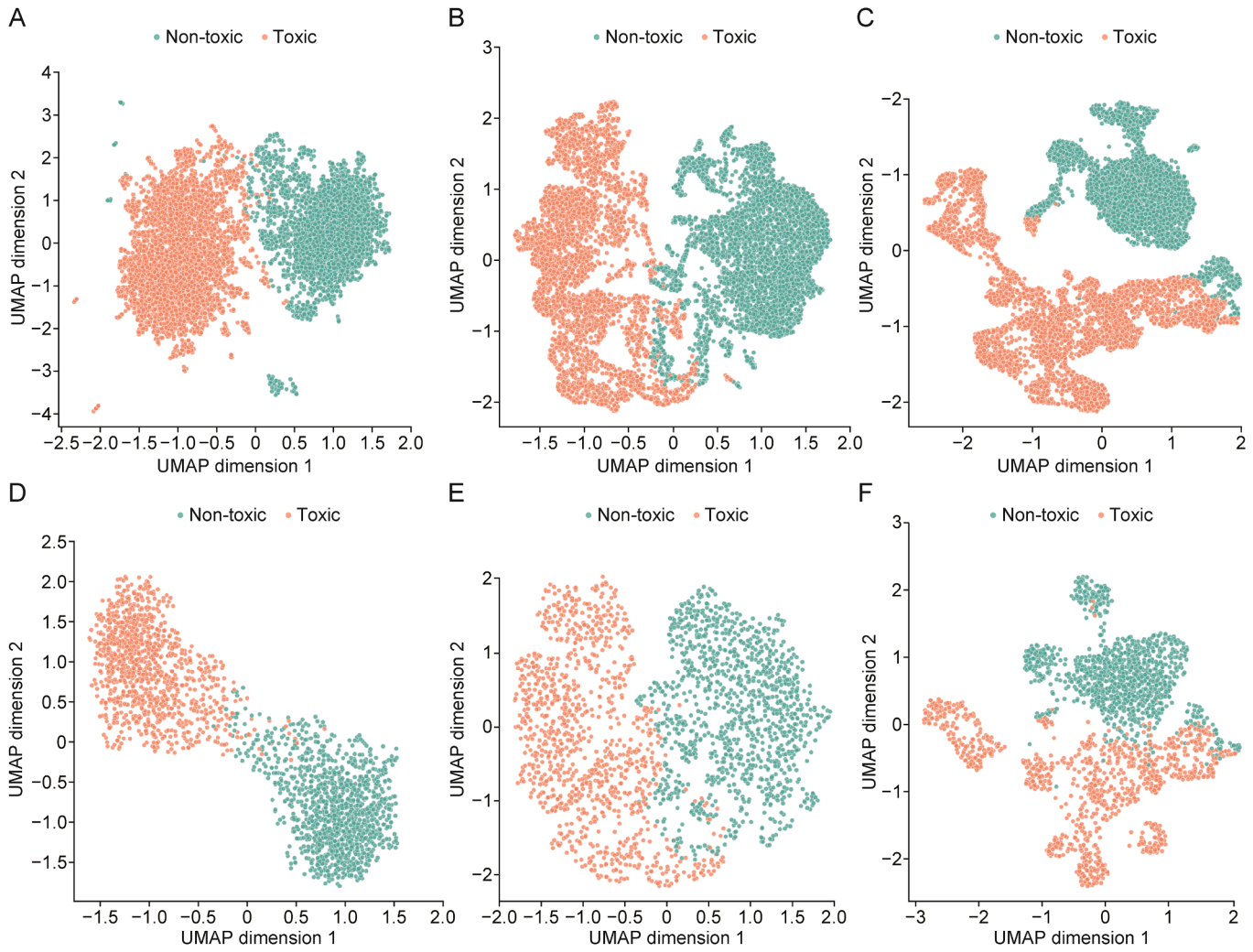

Fig. 5. Uniform manifold approximation and projection (UMAP) visualization of learned features. (A) Only concatenated conventional descriptors (CCDs) features on the training dataset. (B) Only protein language model (PLM)-based embeddings on the training dataset. (C) HyPepTox-Fuse features on the training dataset. (D) Only CCDs features on the independent dataset. (E) Only PLM-based embeddings on the independent dataset. (F) HyPepTox-Fuse features on the independent dataset.

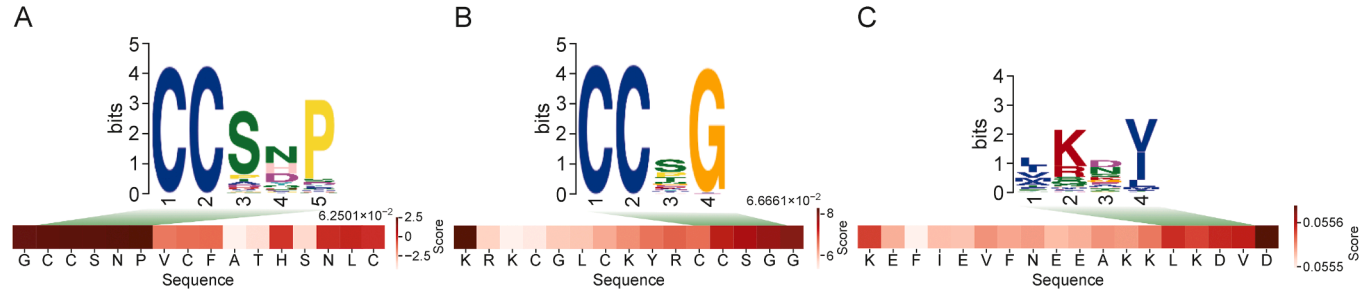

Fig. 6. Visualization of attention heatmaps compared with top 3 motifs discovered by the multiple expectation maximizations for motif elicitation (MEME) tool, where the attention heatmaps closely match the motifs. Attention heatmaps for the (A) "CCSNP" motif, (B) "CCSG" motif, and (C) "LKDV" motif.

$\tau = 0.1$ consistently outperforms the other values, indicating a clear and stable performance. Overall, the model utilizing $\tau = 0.1$ consistently achieves better performance and maintains a more balanced relationship between SPE and SEN metrics, underscoring the importance of selecting an appropriate temperature value for L_{NTXent}, as improperly chosen temperature parameters can lead to suboptimal outcomes.

3.5.4. Ablation study of architecture modules

To investigate the contribution of each module in HyPepTox-Fuse, we conducted an ablation study to assess the impact of the FIs and TF modules on model performance. Fig. 7 shows the effects of these modules during the cross-validation and independent evaluations. The model without FIs (referred to as w/o FIs) employs a simple concatenation of all features, whereas the model

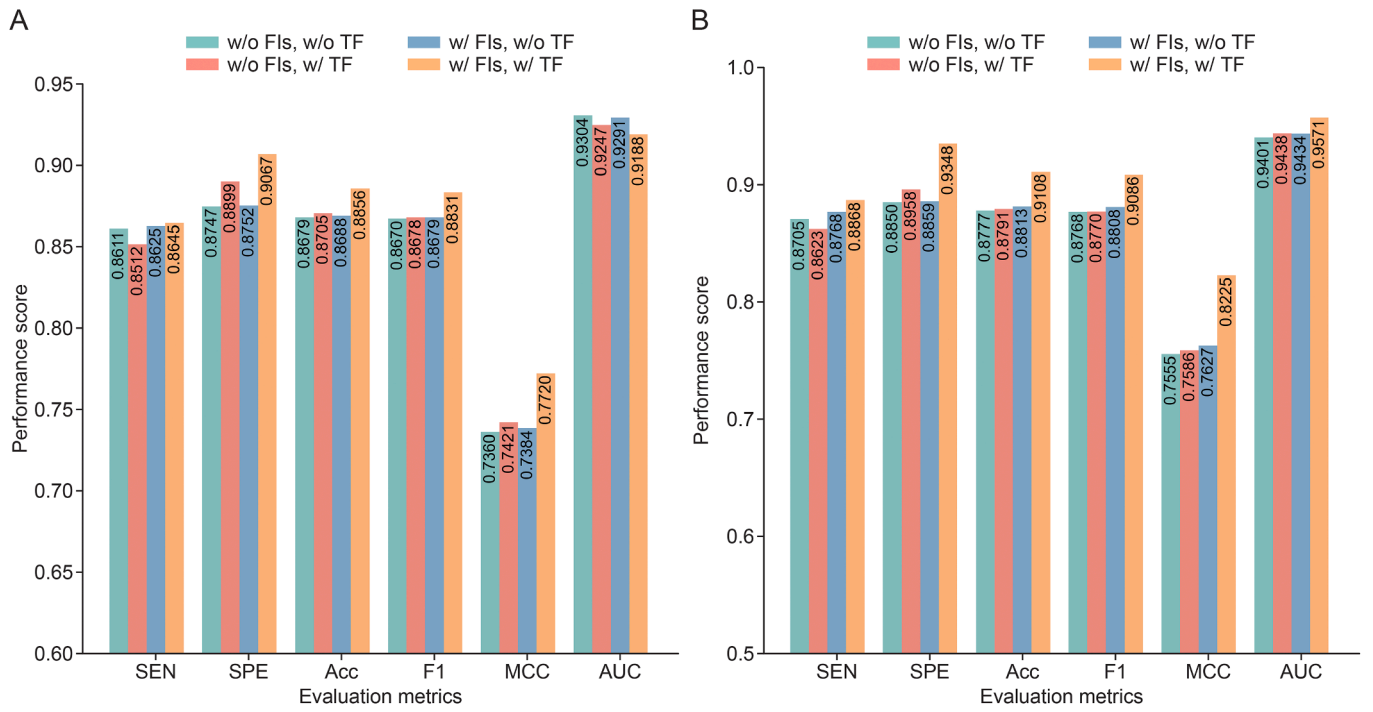

Fig. 7. Performance comparison of each ablation study of architecture's modules on the cross-validation and independent datasets. (A) Performance evaluation on cross-validation datasets. (B) Performance evaluation on independent dataset. w/o: without; FIs: feature interactions; TF: Transformer fusion; w/: with; SEN: sensitivity; SPE: specificity; Acc: accuracy; F1: F1-score; MCC: Matthews correlation coefficient; AUC: area under the receiver operating characteristic curve.

lacking TF (w/o TF) utilizes averaged sequence embeddings to generate a single embedding for classification. To isolate the influence of these modules from conventional features, we conducted the ablation study using only PLM-based embeddings. In cross-validation, the model excluding both FIs and TF exhibited the lowest performance, with MCC, Acc, and F1 values of 0.7360, 0.8679, and 0.8670, respectively. Integrating either FIs or TF yielded slight improvements across nearly all global metrics except AUC. However, the model incorporating both FIs and TF significantly outperformed the other models, demonstrating notable enhancements of 2.99%–3.60% in MCC, 1.51%–1.77% in Acc, and 1.52%–1.61% in F1. Similar trends were observed for the independent dataset. Integrating either FIs or TF slightly improved performance, whereas the model with both FIs and TF consistently outperformed the others across all evaluation metrics. HyPepTox-Fuse showed significant improvements of 5.98%–6.70% in MCC, 1.00%–2.45% in SEN, 3.90%–4.98% in SPE, 2.95%–3.31% in Acc, 2.78%–3.18% in F1, and 1.33%–1.70% in AUC. Overall, this analysis demonstrates the effectiveness and robustness of integrating both FIs and TF within the HyPepTox-Fuse framework, resulting in enhanced performance.

3.6. Case study

To further validate the robustness and generalizability of HyPepTox-Fuse, we employed an additional benchmark dataset from ToxTeller [55]. This dataset comprises peptide sequences of up to 50 amino acids, including 2151 non-toxic and 1978 toxic peptides for training and 100 non-toxic and 100 toxic peptides designated for independent evaluation.

It is important to note that we could not directly apply the trained HyPepTox-Fuse model, which was built on the ToxinPred 3.0 training dataset, to evaluate the ToxTeller independent dataset, due to overlapping samples. Therefore, we adopted the same methodology outlined in this study to develop a new model by utilizing the ToxTeller training dataset (Figs. S6–S10, Tables S9 and S10 and Supplementary data). We subsequently assessed our proposed method on the ToxTeller independent dataset, comparing its performance with that of existing methods, including the CSM-Toxin, ToxIBLT, and ToxTeller models. Because the ToxIBLT server was unavailable, we reproduced its results by training the ToxIBLT model on the ToxTeller training dataset using its open-source code to ensure a fair comparison. For the ToxTeller models, results were derived from the original publication [55]. Importantly, ToxinPred 3.0 was excluded from this evaluation because of overlapping training samples with the ToxTeller independent dataset.

Table 2 shows the results of a performance comparison of all methods applied to the ToxTeller independent dataset. CSM-Toxin exhibited consistently poor performance, comparable with that of a random classifier (Acc = 0.4950), showing a strong bias toward non-toxic peptides, with an SPE of 0.9300 and an SEN of 0.0600.

Table 2
Performance comparison between HyPepTox-Fuse and existing methods on the ToxTeller independent dataset.

Model	SEN	SPE	Acc	F1	MCC	AUC
CSM-Toxin	0.0600	0.9300	0.4950	0.1062	−0.0203	0.3654
ToxIBLT	0.6800	0.9000	0.7900	0.7640	0.5946	0.8576
ToxTeller-LR	0.7000	0.9100	0.8050	0.7820	0.6240	0.8700
ToxTeller-SVM	0.7500	0.9300	0.8400	0.8240	0.6910	0.9250
ToxTeller-RF	0.7100	**0.9600**	0.8350	0.8110	0.6920	0.9220
ToxTeller-XGB	0.7700	0.9400	0.8550	0.8420	0.7210	0.9300
HyPepTox-Fuse	**0.8400**	0.9300	**0.8850**	**0.8796**	**0.7731**	**0.9540**

The bold data indicate the best performance among the compared methods. SEN: sensitivity; SPE: specificity; Acc: accuracy; F1: F1-score; MCC: Matthews correlation coefficient; AUC: area under the receiver operating characteristic curve; LR: linear regression; SVM: support vector machine; RF: random forest; XGB: extreme gradient boosting.

Conversely, ToxIBLT demonstrated reasonable predictive ability for non-toxic peptides, but misclassified a substantial number of toxic peptides, yielding an Acc of 0.7900, an F1 of 0.7640, an MCC of 0.5946, and an AUC of 0.8576. Among the four models reported by ToxTeller, including logistic regression (ToxTeller-LR), support vector machine (ToxTeller-SVM), random forest (ToxTeller-RF), and extra gradient boosting (ToxTeller-XGB), ToxTeller-XGB achieved the highest performance, with an Acc of 0.8550, an F1 of 0.8420, an MCC of 0.7210, and an AUC of 0.9300. Notably, HyPepTox-Fuse significantly surpassed all existing methods across global metrics, achieving an Acc of 0.8850, an F1 of 0.8796, an MCC of 0.7731, and an AUC of 0.9540. Specifically, it demonstrated improvements of 3.00%–39.00% in Acc, 3.76%–77.34% in F1, 5.21%–79.34% in MCC, and 2.40%–58.86% in AUC compared with existing methods. These results confirm the superior predictive capability of HyPepTox-Fuse, with fewer errors and misclassifications of toxic peptides, underscoring the effectiveness of our proposed method in enhancing model performance.

3.7. Web server development

The HyPepTox-Fuse framework for peptide toxicity prediction has been deployed and is accessible at https://balalab-skku.org/HyPepTox-Fuse/. This deployment utilizes an ensemble approach, wherein predictions are generated based on the collective outputs of five models derived from 5-fold cross-validation. The final prediction is determined through a voting mechanism that considers the probabilities predicted by each of the five models. The web server offers access to all the training and independent datasets utilized in this study, thereby ensuring transparency and facilitating the reproducibility of the results. Users can interact with the HyPepTox-Fuse server by uploading a file containing multiple sequences in FASTA format or by directly entering one or more query sequences in FASTA format. Upon successful completion of the analysis, the results are presented through a user-friendly interface, allowing for easy viewing and analysis of the predictions. Additionally, the results can be downloaded in the CSV format for future reference, providing flexibility and convenience. Moreover, to support local deployment, the implementation of HyPepTox-Fuse is available via the GitHub repository at https://github.com/cbbl-skku-org/HyPepTox-Fuse/.

3.8. Future work

The novel approach of HyPepTox-Fuse can be applied to other biological sequence analysis tasks, such as RNA post-transcriptional modification [56,57], protein post-translational modification [43,46,58], anticancer peptide identification [42,59], and other peptide therapeutic function predictions [60]. Although HyPepTox-Fuse proves to be effective in identifying peptide toxicity, there is still room for further improvements. Future research will focus on expanding the datasets to enable the identification of multiple types of toxicity peptides and extending the study to include peptides with non-canonical amino acids. Additionally, we plan to investigate the mechanisms of action in greater detail and utilize multimodal features, such as structural and sequence information, thereby enhancing the model's biological interpretability and predictive power. By addressing these challenges, HyPepTox-Fuse could be pivotal in advancing therapeutic peptide design, enabling safer and more effective drug development. HyPepTox-Fuse employed an auxiliary loss function for contrastive learning combined with supervised learning. In a later study, we could integrate a Siamese network-based contrastive learning framework as an unsupervised approach [61] for feature representation learning to enhance HyPepTox-Fuse architecture. Finally, with the advantage of large language models (LLMs), future studies can focus on fine-tuning models such as generative pre-trained Transformer (GPT)-based models and prompting-based methods [62] to enhance the performance of peptide toxicity prediction.

4. Conclusion

Peptides are essential biomolecules with considerable therapeutic potential for various diseases. However, their clinical application is often limited by toxicity challenges. The accurate prediction of peptide toxicity is crucial for the development of safe peptide-based therapeutics. Traditional experimental methods are time-consuming and expensive, prompting the emergence of computational approaches, including similarity-based techniques and computational methods, as effective alternatives. In this study, we introduce HyPepTox-Fuse, a novel interpretable hybrid framework that combines PLM-based embeddings and conventional descriptors to predict peptide toxicity. Rigorous cross-validation and independent evaluations demonstrated that HyPepTox-Fuse achieves superior performance across all metrics compared to existing tools. By leveraging a cross-modal multi-head attention mechanism and Transformer architecture, HyPepTox-Fuse effectively integrates diverse features, achieving state-of-the-art performance. The integration of the top CCDs with PLM-based embeddings significantly enhanced performance, highlighting the complementary nature of these modalities. Our model interpretation further revealed biologically relevant sequence patterns through attention weights, which aligned with known motifs, showcasing the model's ability to identify critical sequence features for its predictions.

CRediT authorship contribution statement

Duong Thanh Tran: Writing – original draft, Visualization, Validation, Software, Methodology, Data curation, Conceptualization. **Nhat Truong Pham:** Writing – original draft, Software, Methodology, Conceptualization. **Nguyen Doan Hieu Nguyen:** Visualization, Software, Methodology, Conceptualization. **Leyi Wei:** Writing – original draft, Methodology. **Balachandran Manavalan:** Writing – review & editing, Writing – original draft, Supervision, Resources, Funding acquisition, Conceptualization.

Data availability

The web server for HyPepTox-Fuse, along with the training and independent datasets, is freely accessible at https://balalab-skku.org/HyPepTox-Fuse/. The implementation of HyPepTox-Fuse is publicly available at https://github.com/cbbl-skku-org/HyPepTox-Fuse/ for local deployment.

Declaration of competing interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF), funded by the Ministry of Science and ICT, Republic of Korea (Grant No.: RS-2024-00344752). This research was also supported by the Department of Integrative Biotechnology, Sungkyunkwan University (SKKU) and the BK21 FOUR Project, Republic of Korea. The authors would like to thank the Korea Bio Data Station (K-BDS) for providing computing resources, including technical support. The authors also thank the Korea Association for

ICT Promotion (KAIT) for providing computing resources and technical support. Fig. 1 and the Graphical Abstract were drawn by using Flaticon.com and BioRender.com.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jpha.2025.101410.

References

[1] D.J. Craik, D.P. Fairlie, S. Liras, et al., The future of peptide-based drugs, Chem. Biol. Drug Des. 81 (2013) 136–147.
[2] K. Fosgerau, T. Hoffmann, Peptide therapeutics: current status and future directions, Drug Discov. Today 20 (2015) 122–128.
[3] J. Thundimadathil, Cancer treatment using peptides: current therapies and future prospects, J. Amino Acids 2012 (2012), 967347.
[4] E.A.G. Blomme, Y. Will, Toxicology strategies for drug discovery: present and future, Chem. Res. Toxicol. 29 (2016) 473–504.
[5] F. Khan, K. Niaz, M. Abdollahi, Toxicity of biologically active peptides and future safety aspects: an update, Curr. Drug Discov. Technol. 15 (2018) 236–242.
[6] M. Duracova, J. Klimentova, A. Fucikova, et al., Proteomic methods of detection and quantification of protein toxins, Toxins 10 (2018), 99.
[7] S.F. Altschul, T.L. Madden, A.A. Schäffer, et al., Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, Nucleic Acids Res. 25 (1997) 3389–3402.
[8] G. Klambauer, T. Unterthiner, A. Mayr, et al., DeepTox: toxicity prediction using deep learning, Toxicol. Lett. 280 (2017), S69.
[9] P. Banerjee, A.O. Eckert, A.K. Schrey, et al., ProTox-II: a webserver for the prediction of toxicity of chemicals, Nucleic Acids Res. 46 (2018) W257–W263.
[10] N. Sharma, L.D. Naorem, S. Jain, et al., ToxinPred2: an improved method for predicting toxicity of proteins, Briefings Bioinf. 23 (2022), bbac174.
[11] A.S. Rathore, S. Choudhury, A. Arora, et al., ToxinPred 3.0: an improved method for predicting the toxicity of peptides, Comput. Biol. Med. 179 (2024), 108926.
[12] S. Gupta, P. Kapoor, K. Chaudhary, et al., In silico approach for predicting toxicity of peptides and proteins, PLoS One 8 (2013), e73957.
[13] S. Saha, G.P. Raghava, BTXpred: prediction of bacterial toxins, Silico Biol. 7 (2007) 405–412.
[14] S. Saha, G.P. Raghava, Prediction of neurotoxins based on their function and source, Silico Biol. 7 (2007) 369–387.
[15] G. Naamati, M. Askenazi, M. Linial, ClanTox: a classifier of short animal toxins, Nucleic Acids Res. 37 (2009) W363–W368.
[16] E.S.W. Wong, M.C. Hardy, D. Wood, et al., SVM-based prediction of propeptide cleavage sites in spider toxins identifies toxin innovation in an Australian tarantula, PLoS One 8 (2013), e66279.
[17] T.J. Cole, M.S. Brewer, TOXIFY: a deep learning approach to classify animal venom proteins, PeerJ 7 (2019), e7200.
[18] X.Y. Pan, J. Zuallaert, X. Wang, et al., ToxDL: deep learning using primary structure and domain embeddings for assessing protein toxicity, Bioinformatics 36 (2021) 5159–5168.
[19] A. Jain, D. Kihara, NNTox: gene ontology-based protein toxicity prediction using neural network, Sci. Rep. 9 (2019), 17923.
[20] H. Shi, Y. Li, Y. Chen, et al., ToxMVA: an end-to-end multi-view deep autoencoder method for protein toxicity prediction, Comput. Biol. Med. 151 (2022), 106322.
[21] B.Q. Han, N. Zhao, C.S. Zeng, et al., ACPred-BMF: bidirectional LSTM with multiple feature representations for explainable anticancer peptide prediction, Sci. Rep. 12 (2022), 21915.
[22] L.S. Wei, X.C. Ye, Y.Y. Xue, et al., ATSE: a peptide toxicity predictor by exploiting structural and evolutionary information based on graph neural network and attention mechanism, Briefings Bioinf. 22 (2021), bbab041.
[23] Q.L. Yu, Z.X. Zhang, G.X. Liu, et al., ToxGIN: an in silico prediction model for peptide toxicity via graph isomorphism networks integrating peptide sequence and structure information, Brief, Bioinform 25 (2024), bbae583.
[24] A. Vaswani, N. Shazeer, N. Parmar, et al., Attention is all you need, in: Book Attention Is All You Need, Series Attention Is All You Need, 2017, pp. 5998–6008.
[25] G. Ke, Q. Meng, T. Finley, et al., LightGBM: a highly efficient gradient boosting decision tree, in: Book LightGBM: A Highly Efficient Gradient Boosting Decision Tree, Series LightGBM: A Highly Efficient Gradient Boosting Decision Tree, 2017, pp. 3146–3154.
[26] Q. Kaas, J.C. Westermann, R. Halai, et al., ConoServer, a database for conopeptide sequences and structures, Bioinformatics 24 (2008) 445–446.
[27] G.B. Shi, X.Y. Kang, F.Y. Dong, et al., Dramp 3.0: an enhanced comprehensive data repository of antimicrobial peptides, Nucleic Acids Res. 50 (2022) D488–D496.
[28] F.H. Waghu, R.S. Barai, P. Gurung, et al., CAMPR3: a database on sequences, structures and signatures of antimicrobial peptides, Nucleic Acids Res. 44 (2016) D1094–D1097.
[29] J.H. Jhong, L.T. Yao, Y.X. Pang, et al., dbAMP 2.0: updated resource for antimicrobial peptides with an enhanced scanning method for genomic and proteomic data, Nucleic Acids Res. 50 (2022) D460–D470.
[30] S.P. Piotto, L. Sessa, S. Concilio, et al., YADAMP: yet another database of antimicrobial peptides, Int. J. Antimicrob. Agents 39 (2012) 346–351.
[31] M. Pirtskhalava, A.A. Amstrong, M. Grigolava, et al., DBAASP v3: database of antimicrobial/cytotoxic activity and structure of peptides as a resource for development of new therapeutics, Nucleic Acids Res. 49 (2021) D288–D297.
[32] A. Bateman, M.J. Martin, S. Orchard, et al., UniProt: the universal protein knowledgebase in 2021, Nucleic Acids Res. 49 (2021) D480–D489.
[33] A. Bairoch, R. Apweiler, The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000, Nucleic Acids Res. 28 (2000) 45–48.
[34] J.C. Oliveros, VENNY. An Interactive Tool for Comparing Lists with Venn Diagrams, 2007 ([LinkOut]).
[35] A. Rives, J. Meier, T. Sercu, et al., Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences, P. Natl. Acad. Sci. USA 118 (2021), e2016239118.
[36] Z.M. Lin, H. Akin, R.S. Rao, et al., Evolutionary-scale prediction of atomic-level protein structure with a language model, Science 379 (2023) 1123–1130.
[37] A. Elnaggar, M. Heinzinger, C. Dallago, et al., ProtTrans: toward understanding the language of life through self-supervised learning, IEEE Trans. Pattern Anal. Mach. Intell. 44 (2022) 7112–7127.
[38] F. Zhang, J.F. Li, Z.G. Wen, et al., FusPB-ESM2: fusion model of ProtBERT and ESM-2 for cell-penetrating peptide prediction, Comput. Biol. Chem. 111 (2024), 108098.
[39] F. Indriani, K.R. Mahmudah, B. Purnama, et al., ProtTrans-glutar: incorporating features from pre-trained transformer-based models for predicting glutarylation sites, Front. Genet. 13 (2022), 885929.
[40] Z.H. Kilimci, M. Yalcin, ACP-ESM: a novel framework for classification of anticancer peptides using protein-oriented transformer approach, Artif. Intell. Med. 156 (2024), 102951.
[41] V.T. Le, Z.J. Zhan, T.T.P. Vu, et al., ProtTrans and multi-window scanning convolutional neural networks for the prediction of protein-peptide interaction sites, J. Mol. Graph. Model. 130 (2024), 108777.
[42] V.K. Sangaraju, N.T. Pham, L. Wei, et al., mACPpred 2.0: stacked deep learning for anticancer peptide prediction with integrated spatial and probabilistic feature representations, J. Mol. Biol. 436 (2024), 168687.
[43] N.T. Pham, Y. Zhang, R. Rakkiyappan, et al., HOTGpred: enhancing human O-linked threonine glycosylation prediction using integrated pretrained protein language model-based features and multi-stage feature selection approach, Comput. Biol. Med. 179 (2024), 108859.
[44] S. Basith, N.T. Pham, B. Manavalan, et al., SEP-AlgPro: an efficient allergen prediction tool utilizing traditional machine learning and deep learning techniques with protein language model features, Int. J. Biol. Macromol. 273 (2024), 133085.
[45] Z. Chen, X.H. Liu, P. Zhao, et al., iFeatureOmega: an integrative platform for engineering, visualization and analysis of features from molecular sequences, structural and ligand data sets, Nucleic Acids Res. 50 (2022) W434–W447.
[46] N.T. Pham, L.T. Phan, J. Seo, et al., Advancing the accuracy of SARS-CoV-2 phosphorylation site detection via meta-learning approach, Briefings Bioinf. 25 (2023), bbad433.
[47] J. Devlin, M.-W. Chang, K. Lee, et al., BERT: pre-training of deep bidirectional transformers for language understanding, in: Book BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Series BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019, pp. 4171–4186.
[48] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, in: Book Decoupled Weight Decay Regularization, Series Decoupled Weight Decay Regularization, 2019.
[49] K. Sohn, Improved deep metric learning with multi-class n-pair loss objective, in: Book Improved Deep Metric Learning with Multi-Class N-Pair Loss Objective, Series Improved Deep Metric Learning with Multi-Class N-Pair Loss Objective, 2016, pp. 1849–1857.
[50] T. Chen, S. Kornblith, M. Norouzi, et al., A simple framework for contrastive learning of visual representations, in: Book A Simple Framework for Contrastive Learning of Visual Representations, Series A Simple Framework for Contrastive Learning of Visual Representations, 2020, pp. 1597–1607.
[51] V. Morozov, C.H.M. Rodrigues, D.B. Ascher, CSM-Toxin: a web-server for predicting protein toxicity, Pharmaceutics 431 (2023), 431.
[52] L.S. Wei, X.C. Ye, T. Sakurai, et al., ToxIBTL: prediction of peptide toxicity based on information bottleneck and transfer learning, Bioinformatics 38 (2022) 1514–1524.
[53] L. McInnes, J. Healy, N. Saul, et al., UMAP: uniform manifold approximation and projection, J. Open Source Softw. 3 (2018), 861.
[54] T.L. Bailey, C. Elkan, Fitting a mixture model by expectation maximization to discover motifs in biopolymers, Proc. Int. Conf. Intell. Syst. Mol. Biol. 2 (1994) 28–36.
[55] J.H. Wang, T.Y. Sung, ToxTeller: predicting peptide toxicity using four different machine learning approaches, ACS Omega 9 (2024) 32116–32123.
[56] N.T. Pham, R. Rakkiyapan, J. Park, et al., H2Opred: a robust and efficient hybrid deep learning model for predicting 2'-O-methylation sites in human RNA, Briefings Bioinf. 25 (2023), bbad476.

[57] N.T. Pham, A.T. Terrance, Y.-J. Jeon, et al., ac4C-AFL: a high-precision identification of human mRNA N4-acetylcytidine sites based on adaptive feature representation learning, Mol. Ther. Nucleic Acids 35 (2024), 102192.
[58] S. Basith, G. Lee, B. Manavalan, STALLION: a stacking-based ensemble learning framework for prokaryotic lysine acetylation site prediction, Briefings Bioinf. 23 (2022), bbab376.
[59] L.T. Phan, H.W. Park, T. Pitti, et al., Mlacp 2.0: an updated machine learning tool for anticancer peptide prediction, Comput. Struct. Biotechnol. J. 20 (2022) 4473–4480.
[60] S. Basith, B. Manavalan, T.H. Shin, et al., Machine intelligence in peptide therapeutics: a next-generation tool for rapid disease screening, Med. Res. Rev. 40 (2020) 1276–1314.
[61] X. Zhangt, L.S. Weit, X.C. Ye, et al., SiameseCPP: a sequence-based Siamese network to predict cell -penetrating peptides by contrastive learning, Briefings Bioinf. 24 (2023), bbac545.
[62] P. Shrestha, J. Kandel, H. Tayara, et al., Post-translational modification prediction via prompt-based fine-tuning of a GPT-2 model, Nat. Commun. 15 (2024), 6699.