ACP-ESM2: Enhancing Anticancer Peptide Prediction With Pre-Trained Protein Language Models

Shun Gao, Yan Xia, Xingfeng Li, Feifei Cui, Qingchen Zhang, Quan Zou, and Zilong Zhang

***Abstract*—Anticancer peptide (ACP) are short peptides with anti-cancer properties that have generated increasing attention in recent years due to their low toxicity, minimal side effects, and their ability to precisely target and kill cancer cells. Traditionally, identifying ACP has relied on experimental methods, which are time-consuming and labor-intensive. While deep learning-based prediction methods have made significant progress, there is still room for improvement in achieving optimal performance. In this study, we present ACP-ESM2, a deep learning framework based on the Evolutionary Scale Modeling 2 (ESM2) pre-trained model, which captures rich evolutionary information from protein sequences. By combining ESM2 with convolutional neural network (CNN) that excels at detecting local patterns, ACP-ESM2 offers a highly accurate tool for ACP prediction. The experimental results indicate that ACP-ESM2 shows significant improvements over best-existing recognition techniques on the Test1 set, with enhancements of 2.3% , 7.2% , 12.6% , and 5% in ACC, SN, SP, and MCC, respectively. Notably, on the Test2 set, ACP-ESM2 achieves an accuracy of 97.6% , showcasing its exceptional robustness. This establishes ACP-ESM2 as an efficient and precise tool for predicting anticancer peptides.**

***Index Terms*—Anticancer peptide, ESM2, CNN-attention, ACP predictor.**

I. Introduction

CANCER is a disease caused by pathological changes in cell division and has become a leading cause of death worldwide [1], [2], [3], [4]. It not only poses a severe threat to human health but also greatly diminishes patients' quality of life. Conventional treatment methods mainly employ methods such as chemotherapy and radiotherapy, in which the spread of cancer cells is inhibited by the use of chemical drugs, or local treatment is carried out using radiation. However, although these methods can play a therapeutic role to a certain extent, they also bring serious side effects to patients and are expensive to treat [5], [6]. In recent years, Anticancer peptide (ACP) have attracted much attention due to their highly selective killing effect on cancer cells while causing less damage to normal cells [7]. ACP are able to precisely target cancer cells, thus improving the therapeutic effect and reducing the adverse effects on patients in the process of treatment, which provides new ideas and methods for cancer treatment [8], [9].

ACP are proteins with anticancer properties, usually short peptides containing 5–50 amino acids. ACP exerts its main effects by interacting with cell membranes and inducing their cleavage or permeation [10]. In addition, some ACP exhibit a variety of other mechanisms of action, including inhibition of angiogenesis, induction of apoptosis in tumor cells, targeting of essential cell proteins, and recruitment of immune cells [11]. Traditionally, identifying ACP has relied on experimental methods, which are time-consuming and labor-intensive. With the rapid development of machine learning techniques, numerous prediction tools have been developed to accelerate research in this field [12].

Most of these methods use manual feature extraction, such as ACPred which utilizes the manual sequence-based feature combination combined with a support vector machine (SVM) to predict ACP [13]. Similarly, Vinothini Boopathi et al. developed mACPred, which uses sequence composition information and physicochemical properties as feature sets, leveraging SVM and Random Forest (RF) algorithms for predictive analysis [14]. In addition, Huang et al. developed a method for detecting ACP by analyzing sequence features and physicochemical properties, again using SVM as the underlying algorithm [15]. MLACP 2.0 constructs a meta-model by combining multiple encoding methods and various machine learning classifiers to predict anticancer peptides [4]. mACPpred 2.0 employs a stacked deep learning strategy, integrating 1D convolutional neural network modules and hybrid features to enhance the accuracy of anticancer peptide prediction [16]. ACPredStackL utilizes a stacking ensemble strategy, combining SVM Naive Bayes, LightGBM, and KNN to optimize the performance and stability of anticancer peptide recognition [17].

Received 7 October 2024; revised 31 December 2024; accepted 21 February 2025. Date of publication 4 March 2025; date of current version 5 June 2025. The work was supported by the National Natural Science Foundation of China under Grant 62101100 and Grant 62262015. *(Corresponding author: Zilong Zhang.)*

Shun Gao, Yan Xia, Xingfeng Li, Feifei Cui, Qingchen Zhang, and Zilong Zhang are with the School of Computer Science and Technology, Hainan University, Haikou 570228, China (e-mail: gaoshun@hainanu.edu.cn; yanxia@hainanu.edu.cn; lixingfeng@hainanu.edu.cn; feifeicui@hainanu.edu.cn; zhangqingchen@hainanu.edu.cn; zhangzilong@hainanu.edu.cn).

Quan Zou is with the Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 610054 USA, and also with the Yangtze Delta Region Institute (Quzhou), University of Electronic Science and Technology of China, Quzhou 324000, China (e-mail: zouquan@nclab.net).

To facilitate predictions for researchers, we have set up a web server for ACP-ESM2, which is available at http://www.bioai-lab.com/ACP-ESM2.

Digital Object Identifier 10.1109/TCBBIO.2025.3547952

TABLE I
THE TRAINING SET AND TEST SET FOR ACP

Dataset	Negative sample	Positive samples	Total
Training dataset	1479	487	1966
Test1 Set	360	135	495
Test2 Set	157	99	256

In recent years, with the improvement of computational performance, deep learning methods have been developed, which do not require tedious manual fetching and extraction of features to represent the input data as compared to machine learning methods [18], [19], [20]. With its good scalability and adaptability, deep learning has been developed to solve problems related to biological sequences [21], [22]. For example, ACP-MHCNN uses a multi-head deep convolutional neural network to effectively capture and combine discriminative features from diverse data inputs in an interactive manner for ACP recognition [23]. Han et al. proposed a deep learning-based ACPred-BMF predictor for ACP prediction based on a peptide sequence representation and a bidirectional LSTM neural network framework [24]. Furthermore, ACP-2DCNN utilizes the Dipeptide Deviation from Expected Mean (DDE) method to extract key features, followed by a two- dimensional convolutional neural network (2DCNN) technique for ACP classification [25]. Recently, Yang et al. proposed a Comparative ACP Predictor (CACPP) approach, which is an innovative method that utilizes CNN and employs the principle of comparative learning for predicting ACP [26]. Wang et al. developed iACP-DFSRA, an anticancer peptide recognition model based on a dual-channel fusion strategy combining ResCNN and Attention mechanisms [27].

Although existing machine-learning methods have made some progress in ACP prediction, there is still room for improvement. Most approaches rely on manually extracted features, which require extensive domain knowledge and are time-consuming, as selecting the best features for a specific problem can be challenging. Additionally, handcrafted features often fail to capture the full complexity of the data, limiting their applicability and accuracy. While deep learning techniques have shown promise in automating feature extraction and handling complex data, current research mainly focuses on traditional methods and has yet to leverage the potential of the latest pre-trained models fully. Therefore, we propose a novel approach: the ACP-ESM2 architecture, which is based on the pre-trained ESM2 model [28] with CNN to more accurately differentiate between ACPs and non-ACPs. First, we utilized ESM2 to process ACP sequences, extracting key sequence features rather than relying on traditional manual feature selection. This approach enables us to gain a more comprehensive understanding of the important information in ACP sequences, laying a solid foundation for subsequent prediction tasks. Secondly, we introduced these extracted features into CNN for further deep learning, effectively capturing local patterns and improving the accuracy of ACP prediction. In addition, we incorporated an attention mechanism, allowing the model to focus on features most relevant to the prediction task. This method of integrating the ESM2 model with the CNN network maximizes their respective advantages, achieving accurate predictions of ACP. Empirical evidence has demonstrated that our approach yields good results in ACP prediction.

II. METHODS AND MATERIALS

A. Datasets

To assess the performance of our model against state-of-the-art methods, we utilized the dataset compiled by Bian et al., which consists of a training set, test set one, and test set two [29]. The training set and test set one were sourced from CancerPPD [30], APD3 [31], and SATPdb [32]. Next, the Seqkit tool was used to extract sequences ranging from 5 to 50 in length, retaining only those sequences that could access the PSSM matrix through the PSI-BLAST[33] tool, and CD-Hit [34] was employed to eliminate sequences with over 90% homology. In order to test the generalization ability of the model, the test set constructed by Agrawal et al. was used as Test Set Two, as done by Bian et al., and it was similarly processed as described above [35]. Table I summarizes the training and test sets used for ACP prediction.

B. The Architecture of the Proposed Method ACP-ESM2

Fig. 1 illustrates the comprehensive framework of our proposed model, which is divided into four main modules: The ESM2 module, the CNN with self-attention module [36], the classification module, and the web server module. Firstly, the ESM2 module receives the original amino acid sequenceas input and encodes and extracts features from it. With the ESM2 module, we can efficiently extract critical feature details from the sequence, and this step lays the foundation for subsequent processing. Next, after connecting the CNN to the ESM2 module, local information and patterns in the features are further extracted, thus enhancing the model's ability to represent the data. Next, a self-attention mechanism is used to capture global dependencies in the sequence and dynamically adjust the feature representation according to the importance of different locations. This approach enhances the model's ability to discern sequence correlation information. Finally, the feature vectors are fed into a fully connected layer to perform the final classification task. Through this column process, we can make better use of sequence features for classification, thus providing strong support for the prediction of ACP.

C. ESM2 Module

ESM2 is a protein sequence pre-training model based on the Transformer [37] architecture, which is pre-trained on the UniProt dataset through the Masked Language Model Learning pre-training task [38], where the model learns to capture complex models and representations from protein sequences during the pre-training phase. ESM2 aims to leverage large-scale

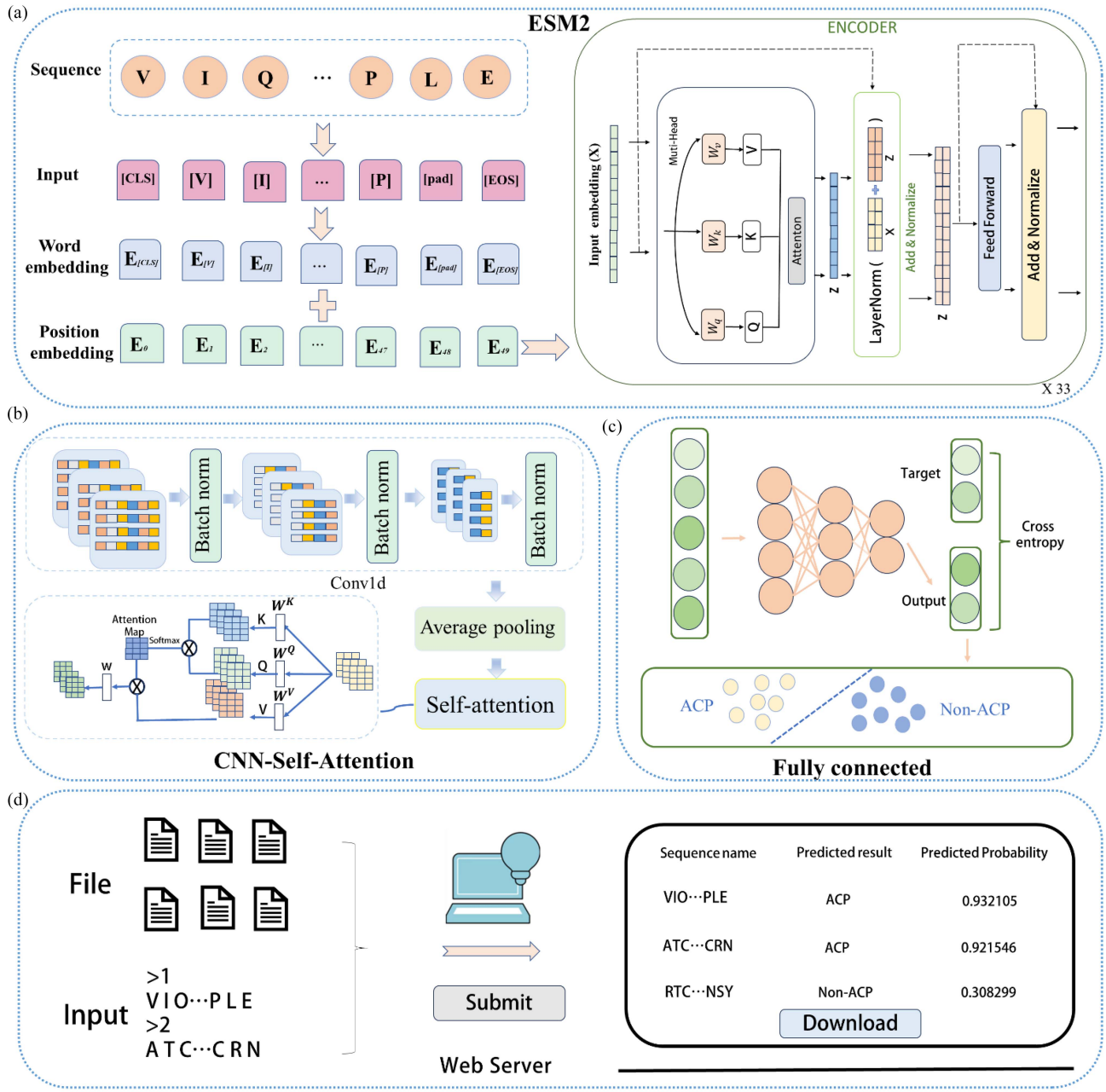

Fig. 1. The model architecture of ACP-ESM2 contains four main modules. (a) ESM2 module. (b) CNN-self-attention module. (c) Classification module. (d) Web Server module.

sequence data and evolutionary information to enhance the understanding of proteins and their functions and structures, which in turn provides more accurate and comprehensive insights into proteins. After pre-training, ESM2 models can be fine-tuned for specific protein-related tasks, and the fine-tuning enables the model to prioritize these tasks and make accurate predictions.

Theoretically, larger models typically achieve better performance due to their ability to capture more complex features and patterns. However, larger models also require more computational resources. During the adjustment process of ACP-ESM2, we sequentially utilized four pre-trained ESM-2 models of varying versions: ESM2_t6_8 M_UR50D, ESM2_t12_35 M_UR50D, ESM2_t30_150 M_UR50D, and ESM2_t33_650 M_UR50D. As shown in Table II, after evaluating the performance of all versions, we selected the ESM2_t33_650M_UR50D version. Its superior ability to capture complex features and comprehend biological information made it the most suitable choice for our research. Initially, we employed ESM2 for encoding, treating each peptide sequence as a text sentence and each amino acid as a word in the input. Next, these encoded peptide sequences were inputted into a pre-trained ESM2 model. Specifically, these peptide sequences were turned into a labeled input consisting of the "cls" character (indicating the start of the sequence), all amino acid characters

TABLE II
THE PERFORMANCE OF DIFFERENT VERSIONS OF ESM2 ON THE TRAINING SET

Versions	ACC	SN	SP	MCC
ESM2_t6_8 M_UR50D	0.914	0.795	0.949	0.754
ESM2_t12_35 M_UR50D	0.919	0.761	0.966	0.764
ESM2_t30_150 M_UR50D	0.927	0.750	0.980	0.786
ESM2_t33_650 M_UR50D	**0.942**	**0.810**	**0.982**	**0.832**

Note: The highest score in each column is shown in bold.

in the amino acid sequence, and the "eos" character (indicating the end of the sequence), where the ESM2 generated word embedding and position embedding, which enables the model to understand the relationships between the different amino acids in the sequence. Finally, an encoder is utilized to process these embedding vectors.

The encoder layer enhances the embedding vectors of input sequences by utilizing a multi-head self-attention mechanism and feed-forward neural network sublayers, resulting in more comprehensive feature representations. The multi-head self-attention mechanism enables the model to compute attention across all positions of each input sequence, capturing associations between positions and allowing simultaneous consideration of different parts of the sequence. The formula is as follows:

$$\begin{cases} MultiHead\,(Q,K,V) = \text{Concat}\ (head_1,\ldots,\ head_h)\,W^O \\ head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right) \\ \text{Attention}\ (Q,K,V)\ =\ softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \end{cases} \tag{1}$$

Where Q, K and V denote the Query, Key and Value matrices computed from different linear transformations of the input sequence. respectively, W_i^Q, W_i^K and W_i^V denote the weight matrices. d_k denotes the dimensionality of the key vectors for each attention header in the Key matrix.

Feedforward neural network sublayer which performs a nonlinear transformation of the contextual representation of each position. This sublayer usually consists of an activation function and two linear transformations, which are computed as follows [39]:

$$output\,(x) = \max\left(0, xW_1 + b_1\right)W_2 + b_2 \tag{2}$$

where W_1 and W_2 are the weight parameters, b_1 and b_2 are the bias terms, respectively, and x is the representation obtained after processing of the multi-head self-attention sublayer.

Each amino acid character in the Encoder layer is represented as a 1280-dimensional hidden vector, which is rich in information that reflects the features at each position in the sequence. In our study, we chose the output of the last layer as the feature we used because this layer usually contains more abstract and advanced feature representations that help us in further analysis and processing.

D. CNN-Self-Attention Module

CNN as a deep learning model, plays a critical role in the analysis of protein sequences [40]. Its unique local sensing and feature learning capabilities make it a powerful tool for processing sequence data. Through the filtering operations of the convolutional layers, CNN can capture local features at different positions, enabling the extraction of key information from protein sequences. This local awareness allows CNN to efficiently identify patterns and structures in protein sequences, laying a solid foundation for further analysis and prediction. In addition, CNN possesses multi-level feature abstraction capabilities, enabling it to automatically learn and extract abstract features from sequence data, thereby gaining a better understanding of the overall structure and properties of the sequences. This comprehensive feature learning ability gives CNN a wide range of applications in protein sequence analysis, providing robust tools and methods for research in the field of bioinformatics.

We chose CNN as a part of our model primarily due to its superior ability to capture local features and its multi-level feature representation advantages. Compared to other models, CNN demonstrates higher computational efficiency and feature extraction capabilities when handling sequence data, while maintaining a relatively simple and easy-to-implement structure. However, CNN also has certain limitations, such as its limited ability to model the relationships between features, making it unable to directly capture the interactions between distant residues in a sequence. To address these limitations, we developed a framework that integrates CNN with attention mechanisms. The attention mechanisms adaptively assign weights to flexibly capture global dependencies, enabling a more comprehensive understanding of the complex structures in protein sequences. By incorporating attention mechanisms, we aim to enhance the model's ability to capture global features, improve feature extraction, and strengthen its expressive power.

To further optimize the model and reduce the risk of overfitting, we employed the dropout technique [41], which randomly sets the output of certain neurons to zero during training. Following this, we applied three one-dimensional convolutional layers to process the extracted features, leveraging convolutional operations to learn relevant patterns. To ensure stability and efficiency, we also incorporated Batch Normalization [42] after each convolutional layer to accelerate the training process. Given an input x, with the convolution kernel of the i-th layer represented by W_i and the bias be b_i, the convolution operation is defined as:

$$Convolution_i = x * W_i + b_i \tag{3}$$

To obtain more accurate results, it is important to use pooling layers to reduce the dimensionality of the final convolutional layer output. The pooling layer usually contains an average pooling layer and a maximum pooling layer. After three convolutional layers, we apply average pooling to the obtained features so that we have a lower dimensional representation with the

following formula:

$$Pooling = \frac{1}{m}\sum_{i=1}^{m} y_i \tag{4}$$

Where $Y = \{y_1, y_2, \ldots, y_m\}$ are the samples in a pooled region. m is the number of samples in the pooled region. The final output is the average of all the values in the pooled region.

In the Self-Attention process, sequence features are transformed into three key representations: query vectors, key vectors, and value vectors. The attention weights are computed by measuring the similarity between the query vectors and the key vectors at each position, followed by normalization using the softmax function. These attention weights are then applied to the value vectors through weighted summation, resulting in a feature representation that highlights the most important sequence elements. Mathematically, Self-Attention can be expressed as follows:

$$\begin{cases} Q = XW^Q \\ K = XW^K \\ V = XW^V \end{cases} \tag{5}$$

$$Self - Attention\,(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{6}$$

where x is the output after pooling, W^Q, W^Q, W^V are the weight matrices that map X to Q, K, V respectively.

E. Classification Module

After processing in the self-attention module, the generated feature vector will be passed to a fully connected classification module for subsequent processing. In this classification module, the feature vectors will be transformed through two fully connected layers [43]. The result is then mapped to a range between 0 and 1 by applying a sigmoid function to perform the final binary classification task. For samples with an output probability lower than 0.5, they will be classified as non-ACP, while for samples with an output probability equal to or greater than 0.5, they will be classified as ACP.

$$\begin{cases} Z = Self - Attention\,(Pooling\ ((\ Conv\ \ (x)))) \\ \text{output} = \text{FC}\ \ (Z) \\ y = sigmoid\,(output) \end{cases} \tag{7}$$

where the fully connected (FC) function is defined as:

$$q_i = q_{i-1}W_i + b_i \tag{8}$$

Where q_{i-1} is the input data of layer i, and q_i is the output data of that layer. w_i is the weight matrix of this layer and b_i is the bias of this layer. This function serves to pass the input real values through a mapping so that they fall within the interval (0, 1). It is usually used in the output layer of a binary classification problem. In the FC layer, the sigmoid function processes the result of the linear transformation (multiplying the inputs by the weights and adding a bias) and the output value will range between 0 and 1, representing the level of activation for each output unit.

F. Web Server Module

To facilitate the research community, we have established a user-friendly ACP-ESM2 web server accessible at http://www.bioai-lab.com/ACP-ESM2. This platform welcomes user-submitted FASTA format sequence data and is purposefully crafted to analyze whether a provided peptide sequence demonstrates potential anticancer properties. The primary focus of this tool is to assist in the prediction and identification of peptides that could potentially combat cancer, aiding researchers in their exploration of novel therapeutic avenues against this disease.

G. Measures To Prevent Overfitting

To cope with the risk of overfitting, we adopt several methods. First, we introduce the dropout mechanism into the architecture of the neural network. By randomly switching off some neurons during the training process, dropout helps to reduce the dependence on specific neurons, thus preventing model overfitting and facilitating the network to develop more robust features [44].

Secondly, we use L2 regularization, implemented through weight decay in the optimizer [45]. Specifically, L2 regularization is introduced by adding a term to the loss function that penalizes large model weights, limiting the model's complexity and preventing overfitting. The loss($\mathcal{L}_{\text{CE}}$) function used is the cross-entropy loss, defined as:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log\left(\hat{y}_i\right) + \left(1 - y_i\right)\log\left(1 - \hat{y}_i\right)\right] \tag{9}$$

where N is the number of samples, y_i is the true label of the i-th sample (either 0 or 1), and $\hat{y}_i$ is the predicted probability for the i-th sample (the probability of being class 1). The cross-entropy loss measures the difference between the predicted probability distribution and the true labels.

To incorporate L2 regularization, we add a penalty term proportional to the squared values of the model weights to the loss function, resulting in the following total loss function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda\sum_{j=1}^{M} w_j^2 \tag{10}$$

where λ is the regularization hyperparameter, and w_j is the j-th model weight. This penalty term helps to prevent the model from overly relying on specific features, limits the growth of the weights, and mitigates overfitting.

In the implementation, we use the weight decay parameter in the optimizer to apply L2 regularization directly during the training process. This method penalizes large weights by adding a term proportional to the weight magnitude during each parameter update, helping to prevent overfitting by controlling the size of the model weights.

Finally, we use a validation set to assess the performance of the model during the training process. By regularly evaluating the model's performance on the validation set and halting training if performance starts to decline, we effectively control the model's generalization ability, ensuring its accuracy and stability in real-world applications [46].

TABLE III
COMPARISON OF EXISTING METHODS ON TEST1

Methods	ACC	SN	SP	MCC
ACP-MHCNN	0.590	0.783	0.338	0.136
ACPred-BMF	0.668	0.837	0.429	0.296
ACP-Pred	0.856	0.914	0.719	0.648
MACPpred	0.870	0.879	0.838	0.659
ACP-ML	0.908	**0.938**	0.829	0.770
ACP-ESM2(our)	**0.931**	0.866	**0.955**	**0.825**

Note: The highest score in each column is shown in bold.

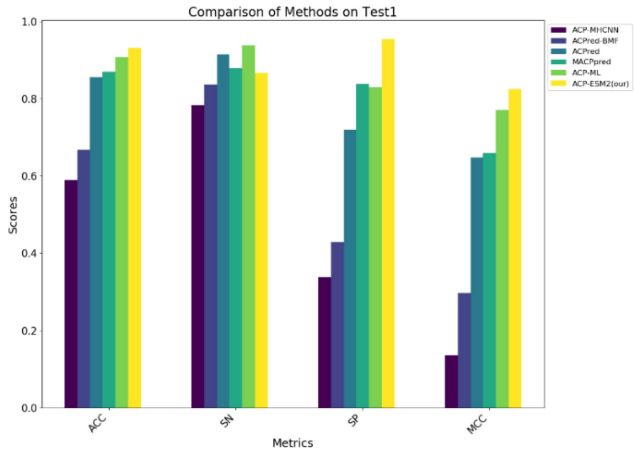

Fig. 2. The performances of ACP-ESM2 and existing methods on the Test1 set.

H. Evaluation Metrics

In this study, we used four commonly used evaluation metrics to evaluate the performance of our model and other existing models [47], [48], [49]: ACC (accuracy), SN (sensitivity), SP (specificity), and MCC (Matthews correlation coefficient). The formulae for these indicators are given below:

$$\begin{cases} ACC = \frac{TP+TN}{TP+FN+TN+FP} \\ SN = \frac{TP}{TP+FN} \\ SP = \frac{TN}{TN+FP} \\ MCC = \frac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \end{cases} \quad (11)$$

TP (True Positive), FN (False Negative), TN (True Negative), and FP (False Positive) denote the number of times the model correctly identifies an ACP as an ACP, incorrectly identifies an ACP as a non-ACP, correctly identifies a non-ACP as a non-ACP, and incorrectly identifies a non-ACP as an ACP, respectively. The ACC measures the model's predicted overall accuracy. The MCC evaluates the performance of binary classification models by considering true positives, false positives, and false negatives. It quantifies the quality of classification results on a scale from -1 to $+1$, where 1 means that the predicted value is in perfect agreement with the actual value, 0 means no better than random, and -1 means no agreement at all. SN (True Positive Rate) measures the ability of the model to correctly identify actual positive samples, and the ability of the model to detect positive classes. SP (true negative rate) measures the ability of the model to correctly identify actual negative samples, and the ability of the model to detect negative classes. Typically, the higher these metrics are, the better the predictive performance of the model.

III. RESULTS

A. Comparison With Existing Methods on the Test Sets

To evaluate the performance of our proposed deep learning model for ACP recognition, we conducted an extensive comparative study using two independent test datasets and compared them with current leading predictors. We considered metrics such as ACC, SN, SP, and MCC to evaluate the overall performance of the model.

As shown in Table III and Fig. 2, by comparing with existing ACP prediction models ACP-ML [29], MACPpred [14], ACPred [13], ACP-MHCNN [23], ACPred-BMF [24]. ACP-ESM2 exhibits a significant advantage in all the metrics. For example, in terms of SP, the ACP-ESM2 model outperforms the current best method by about 12.6%. In terms of ACC, our model also shows excellent performance, with a minimal gap of only about 2.3% compared to other models. In addition, the ACP-ESM2 model improves the performance over the ACP-ML model by about 5.5%, as assessed by the MCC. Combining these results, our study demonstrates that the proposed deep learning model ACP-ESM2 has a clear competitive advantage in ACP prediction, providing important support and insights for the progress of the ACP research field.

TABLE IV
COMPARISON OF EXISTING METHODS ON TEST2

Methods	ACC	SN	SP	MCC
ACP-MHCNN	0.699	0.738	0.625	0.354
ACPred-BMF	0.757	0.920	0.629	0.561
ACPrd	0.863	0.896	0.813	0.714
MACPpred	0.894	0.882	0.918	0.777
ACP-ML	0.925	0.936	0.908	0.843
ACP-ESM2(our)	**0.976**	**0.969**	**0.980**	**0.950**

Note: The highest score in each column is shown in bold.

We further validated the constructed ACP-ESM2 model by using the Test2 set for testing. As shown in Table IV and Fig. 3, the results show that ACP-ESM2 exhibits a very satisfactory generalization ability on the Test2 set with outstanding performance. Specifically, in terms of SP, our ACP-ESM2 model outperforms the other four methods, with a performance improvement ranging from about 7.2% to 35.5%. In addition, in terms of ACC and MCC, ACP-ESM2 also performs more superiorly, significantly outperforming the other predictors. These results further validate the robustness of our proposed ACP-ESM2 model on different datasets, providing strong support for its reliability in practical applications.

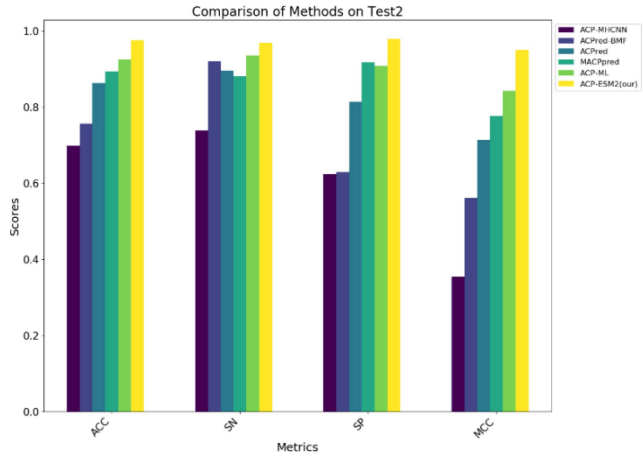

Fig. 3. The performances of ACP-ESM2 and existing methods on the Test2 set.

B. UMAP Visualization and Motif Analysis of Model Attention

To visualize the feature learning capability of our model, we chose UMAP (Uniform Manifold Approximation and Projection) [50] as the dimensionality reduction technique to visualize the distribution of features extracted by our model on the Test1 set and Test2 set. UMAP is a dimensionality reduction technique used to transform high-dimensional data into a more manageable and interpretable format for visualization and analysis. In our visualization graph, each point represents a peptide sequence, while different colored points indicate different ACP and non-ACP categories. As shown in Fig. 4(a)–(b), on the Test1 set, before the model was applied, the data points were entangled and overlapping, making it difficult to differentiate between ACP and non-ACP. However, after model inference, our ACP-ESM2 model successfully partitions the data into two distinguishable clusters, indicating that the model has a strong feature learning ability and can effectively differentiate between ACP and non-ACP sequences. Fig. 4(c)–(d) shows similar results on the Test2 set, further validating the model's generalization ability.

Moreover, to further explore the model's capability in learning sequence features, we conducted an in-depth analysis of the motifs in peptide sequences. Specifically, we extracted the attention features of individual peptide sequences through the self-attention mechanism and visualized them as heatmaps, as shown in Fig. 5 (left). In the heatmap, the intensity of the colors represents the model's focus on different amino acid positions, with darker colors indicating higher importance for the classification task. Simultaneously, we used the traditional tool STREME [51] to perform motif analysis on the same sequences and generated corresponding motif diagrams, as shown in Fig. 5 (right). In the motif diagram, the size of the letters reflects the frequency of specific amino acids at each position, with larger letters indicating higher conservation of that amino acid at the position. By comparing the attention heatmaps generated by the model with the motifs identified by STREME, we observed a high degree of consistency in specific key regions, demonstrating that the model can not only capture critical sequence features

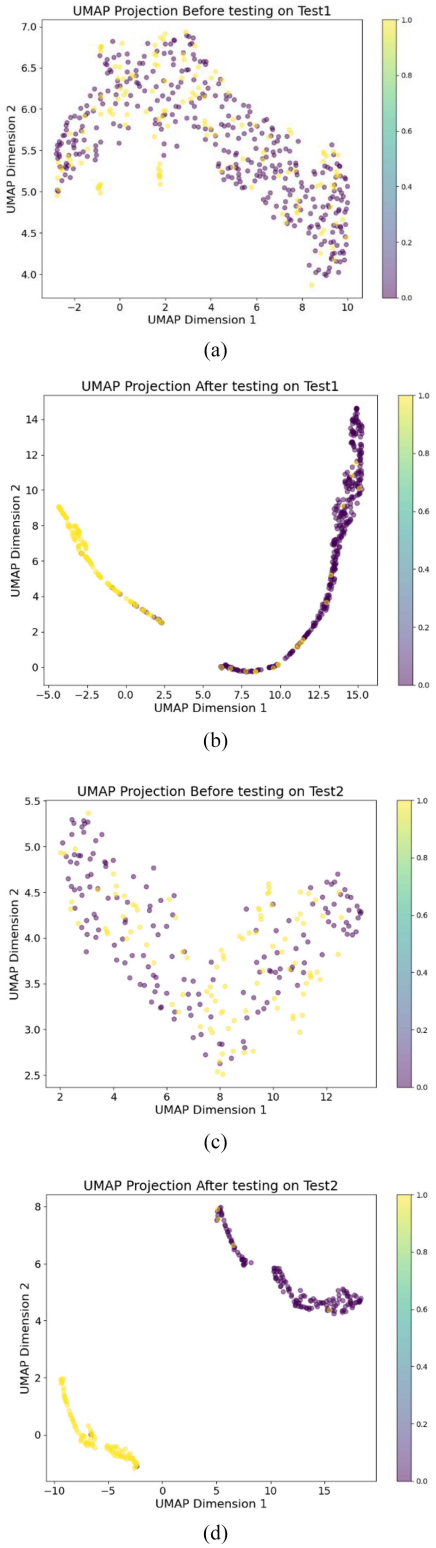

Fig. 4. UMAP visualization of the feature space distribution of our model on the Training set. (a) Input data before inference on Test1 set by ACP-ESM2. (b) Data after inference on Test1 set by ACP-ESM2. (c) Input data before inference on Test2 set by ACP-ESM2. (d) Data after inference on Test2 set by ACP-ESM2.

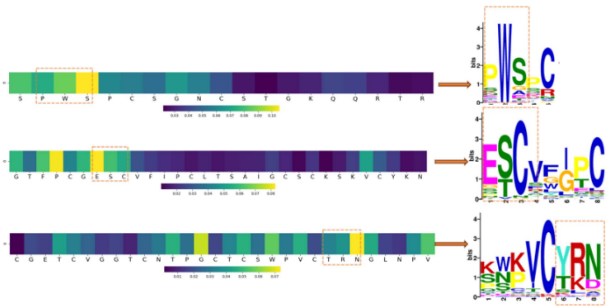

Fig. 5. Attention maps and motifs of the peptide sequences.

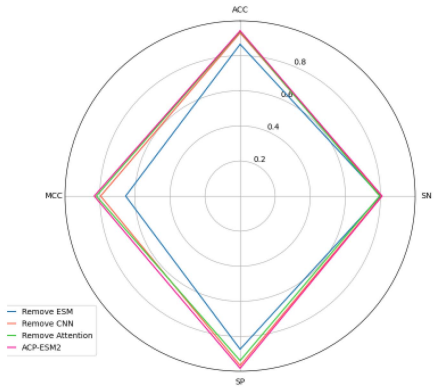

Fig. 6. The performance of model ablation on the Training set.

but also uncover significant regions identified by traditional motif recognition tools. The attention mechanism plays an indispensable role in this process, as it adaptively focuses on the most important parts of the sequence for prediction, thereby enhancing the ability to identify critical regions.

In summary, the combined use of UMAP and motif analysis not only demonstrates the model's ability to effectively learn high-dimensional features but also further validates its capability to capture and interpret key sequence characteristics. UMAP visualization clearly illustrates the model's discriminative performance in classification tasks, while motif analysis reveals biologically meaningful regions at a finer scale. This multi-level analytical approach provides strong support for validating the model's performance and interpretability, laying a solid foundation for future research in related fields.

C. Ablation Experiments for ACP-ESM2

In this section, we conducted a series of ablation experiments to analyze the independent contributions of each component of the ACP-ESM2 model to its predictive performance. To comprehensively evaluate the performance of different models, we applied five-fold cross-validation on the training set. During the experiments, we recorded and analyzed the performance of each model across various evaluation metrics, with the detailed results presented in Table V and Fig. 6. The evaluation included four model variants: removing the ESM module, removing the CNN structure, removing the self-attention mechanism, and the complete ACP-ESM2 model, which integrates ESM2, CNN, and the self-attention mechanism.

TABLE V
THE PERFORMANCE OF MODEL ABLATION ON THE TRAINING SET

Model	ACC	SN	SP	MCC
Remove ESM	0.865	0.801	0.873	0.653
Remove CNN	0.930	0.807	0.966	0.796
Remove Attention	0.938	0.796	0.938	0.820
ACP-ESM2(our)	**0.942**	**0.810**	**0.982**	**0.832**

Note: The highest score in each column is shown in bold.

TABLE VI
THE PERFORMANCE COMPARISON OF DIFFERENT KERNEL SIZES ON THE TRAINING SET

Kernel size	ACC	SN	SP	MCC
3	0.940	0.794	**0.983**	0.824
5*	**0.942**	0.810	0.982	**0.832**
7	0.939	**0.819**	0.976	0.825

Note: The highest score in each column is shown in bold, * represents the kernel size of our choice.

The results show that removing the ESM module leads to a significant performance decline, with ACC dropping to 0.865 and MCC decreasing to 0.653. This indicates that the ESM module, as a pre-trained model, effectively captures deep semantic information and global features of the sequences. It serves as the core component for feature extraction, and its absence results in lower-quality input features, severely impacting the model's predictive accuracy.

When the CNN structure is removed, the model still achieves relatively good performance, with an ACC of 0.930 and MCC of 0.796, but there is a noticeable drop compared to the complete model. This suggests that CNN plays a critical role in capturing local patterns and spatial dependencies within the sequences. By applying convolution operations, CNN extracts local contextual relationships, complementing the ESM module by enhancing fine-grained feature representations.

Removing the self-attention mechanism results in a slight performance decline, with ACC of 0.938 and MCC of 0.820, highlighting the importance of self-attention in modeling global dependencies and long-range interactions. Unlike CNN, which focuses on local feature extraction, the self-attention mechanism captures relationships between different positions in the sequence, enabling the model to learn broader contextual information and improve its handling of complex sequences.

By comparison, the complete ACP-ESM2 model achieves the best performance, with an ACC of 0.942 and MCC of 0.832. This demonstrates the complementary and synergistic effects of the ESM module, CNN structure, and self-attention mechanism. The ESM module provides high-quality initial features, CNN enhances the learning of local features, and the self-attention mechanism strengthens global feature modeling. Together, these components improve the predictive performance of the model. These ablation experiments provide deeper insights into the

unique roles of each component and offer important theoretical support and practical guidance for further model optimization.

D. Model Parameter Optimization

In order to investigate the effect of different CNN layers on the performance of our model, we have conducted a comparative analysis of the model with three different convolutional kernel sizes, and the specific results can be seen in Table VI. As can be seen from the table, the model performs optimally in terms of ACC and MCC when the convolutional kernel size is 5. Although there is a slight decrease in SN and SP relative to the models with convolutional kernel size 7 and convolutional kernel size 3, the difference is very small. Therefore, considering all the metrics, we chose to use a convolutional kernel size of 5 in the CNN structure, a choice that maximizes the model's performance and makes it ideal in different aspects.

IV. Discussion and Limitations

Although our ACP predictor performs well in predicting ACPs, we must acknowledge several limitations and areas for improvement in this study. First, our work does not yet incorporate more detailed structural information, which could be crucial for ACP functionality. For example, the precise arrangement of amino acid residues or the structure of fusion proteins might provide valuable insights into ACP function and activity. Including such structural data could significantly enhance the accuracy and predictive power of the model.

In addressing the data imbalance problem, we attempted to use the focal loss function, a commonly used approach for handling class imbalance. However, despite extensive trials and adjustments, we were unable to find a suitable hyperparameter configuration that substantially improved the classification performance. While focal loss is generally effective for highly imbalanced datasets, it did not significantly enhance performance in our case. This suggests that we may need to explore alternative strategies to better address the data imbalance issue. Potential approaches could include resampling, data augmentation, or adopting other weighted loss functions tailored to our specific dataset and model structure.

Our experimental results also highlight the importance of the different components of the ACP-ESM2 model. Specifically, removing the ESM module, CNN structure, or self-attention mechanism significantly degrades the model's performance. The ESM module, which captures deep semantic information and global features of the sequence, plays a crucial role in maintaining high accuracy. The CNN structure, on the other hand, helps the model learn local patterns and capture short-range interactions critical for ACP identification. Although the model still performs reasonably well without these components, the complete model consistently outperforms alternatives, demonstrating the importance of the synergy between these modules.

Future work will focus on improving the handling of data imbalance. In addition to fine-tuning the hyperparameters of focal loss, we plan to explore other strategies such as custom loss functions or combining different types of weighted loss functions. Furthermore, we aim to integrate more structural information, such as sequence-specific structural data or protein-protein interaction information, to further enhance the model's predictive power and generalizability.

V. Conclusion

Neural network technology plays a crucial role in accurately identifying ACP, providing valuable support for disease treatment and intervention. However, current ACP prediction methods still face several challenges, including limited feature extraction capabilities, insufficient handling of complex sequence data, and a lack of effective strategies for capturing both local and global dependencies. These limitations present significant opportunities for future improvement and innovation.

In this study, we propose a novel deep learning model, ACP-ESM2, which integrates the ESM2 module, CNN structure, and self-attention mechanism to address the aforementioned challenges. Compared to traditional ACP prediction methods, ACP-ESM2 offers significant advantages in feature extraction. The ESM2 module, by utilizing pre-trained embeddings, effectively captures semantic and structural information from sequence data, providing a solid foundation for feature extraction. The CNN structure further enhances the model by learning local patterns and spatial dependencies within the sequences, enabling it to focus on crucial short-range interactions. Meanwhile, the self-attention mechanism helps the model capture long-range dependencies by dynamically adjusting weights, allowing it to better understand the overall sequence structure.

Compared to existing models, ACP-ESM2 demonstrates clear advantages in multiple aspects. Through comprehensive experimental evaluations, we found that ACP-ESM2 consistently outperforms other models in terms of ACC and MCC. In particular, during ablation studies, the complete ACP-ESM2 model, with all its key components intact, achieved the best performance, reaching an ACC of 0.942 and an MCC of 0.832. This result is significantly better than any model that lacks one of its components, highlighting the importance of the synergistic effect of the ESM2 module, CNN, and self-attention mechanism in improving ACP prediction performance.

This study not only advances the field of ACP prediction but also provides valuable insights for bioinformatics, particularly in cancer research. The successful application of ACP-ESM2 demonstrates the potential of deep learning in handling complex biological sequence data, particularly in its ability to extract both local and global features effectively. Future work will focus on further optimizing the ACP-ESM2 model, incorporating more sequence-specific structural information, and exploring other cutting-edge techniques to enhance its accuracy and generalizability. We believe that ACP-ESM2 will not only provide more accurate support for ACP prediction but also serve as a powerful theoretical and technical foundation for the development of ACP-based cancer therapies.

VI. Availability and Implementation

The data and methods are available at https://github.com/birdsmart/ACP-ESM2. For the convenience of the research

community, a web server has been established at http://www.bioai-lab.com/ACP-ESM2.

REFERENCES

[1] M. Eghtedari, S. J. Porzani, and B. Nowruzi, "Anticancer potential of natural peptides from terrestrial and marine environments: A review," *Phytochemistry Lett.*, vol. 42, pp. 87–103, 2021.

[2] B. S. Chhikara and K. Parang, "Global cancer statistics 2022: The trends projection analysis," *Chem. Biol. Lett.*, vol. 10, no. 1, pp. 451–451, 2023.

[3] H. Duan et al., "Machine learning-based prediction model for distant metastasis of breast cancer," *Comput. Biol. Med.*, vol. 169, 2024, Art. no. 107943.

[4] H. W. Park, T. Pitti, T. Madhavan, Y. -J. Jeon, and B. Manavalan, "MLACP 2.0: An updated machine learning tool for anticancer peptide prediction," *Comput. Struct. Biotechnol. J.*, vol. 20, pp. 4473–4480, 2022.

[5] B. Han et al., "Cancer incidence and mortality in China, 2022," *J. Nat. Cancer Center*, vol. 4, pp. 47–53, 2024.

[6] Z. Zhang, F. Cui, M. Zhou, S. Wu, Q. Zou, and B. Gao, "Single-cell RNA sequencing analysis identifies key genes in brain metastasis from lung adenocarcinoma," *Curr. Gene Ther.*, vol. 21, no. 4, pp. 338–348, 2021.

[7] Y. Deng, S. Ma, J. Li, B. Zheng, and Z. Lv, "Using the random forest for identifying key physicochemical properties of amino acids to discriminate anticancer and non-anticancer peptides," *Int. J. Mol. Sci.*, vol. 24, no. 13, 2023, Art. no. 10854.

[8] G. Ghaly et al., "Anti-cancer peptides: Status and future prospects," *Molecules*, vol. 28, no. 3, 2023, Art. no. 1148.

[9] S. Marqus, E. Pirogova, and T. J. Piva, "Evaluation of the use of therapeutic peptides for cancer treatment," *J. Biomed. Sci.*, vol. 24, pp. 1–15, 2017.

[10] E. S. Okeke, I. U. Okagu, K. Chukwudozie, T. C. Ezike, and T. P. C. Ezeorba, "Marine-derived bioactive proteins and peptides: A review of current knowledge on anticancer potentials, clinical trials future prospects," *Natural Product Commun.*, vol. 19, no. 3, 2024, Art. no. 1934578X241239825.

[11] D. Wu, Y. Gao, Y. Qi, L. Chen, Y. Ma, and Y. Li, "Peptide-based cancer therapy: Opportunity and challenge," *Cancer Lett.*, vol. 351, no. 1, pp. 13–22, 2014.

[12] C. Ao, S. Jiao, Y. Wang, L. Yu, and Q. Zou, "Biological sequence classification: A review on data and general methods," *Research*, vol. 2022, 2022, Art. no. 0011.

[13] N. Schaduangrat, C. Nantasenamat, V. Prachayasittikul, and W. Shoombuatong, "ACPred: A computational tool for the prediction and analysis of anticancer peptides," *Molecules*, vol. 24, no. 10, 2019, Art. no. 1973.

[14] V. Boopathi, S. Subramaniyam, A. Malik, G. Lee, B. Manavalan, and D. -C. Yang, "mACPpred: A support vector machine-based meta-predictor for identification of anticancer peptides," *Int. J. Mol. Sci.*, vol. 20, no. 8, 2019, Art. no. 1964.

[15] K. -Y. Huang, Y. -J. Tseng, H. -J. Kao, C. -H. Chen, H. -H. Yang, and S. -L. Weng, "Identification of subtypes of anticancer peptides based on sequential features and physicochemical properties," *Sci. Rep.*, vol. 11, no. 1, 2021, Art. no. 13594.

[16] V. K. Sangaraju, N. T. Pham, L. Wei, X. Yu, and B. Manavalan, "mACPpred 2.0: Stacked deep learning for anticancer peptide prediction with integrated spatial and probabilistic feature representations," *J. Mol. Biol.*, vol. 436, no. 17, 2024, Art. no. 168687.

[17] X. Liang et al., "Large-scale comparative review and assessment of computational methods for anti-cancer peptide identification," *Brief. Bioinf.*, vol. 22, no. 4, 2021, Art. no. bbaa312.

[18] X. Fu et al., "AGF-PPIS: A protein–protein interaction site predictor based on an attention mechanism and graph convolutional networks," *Methods*, vol. 222,pp. 142–151, 2024.

[19] Z. Zhou et al., "PSAC-6mA: 6mA site identifier using self-attention capsule network based on sequence-positioning," *Comput. Biol. Med.*, vol. 171, 2024, Art. no. 108129.

[20] F. Cui, Z. Zhang, and Q. Zou, "Sequence representation approaches for sequence-based protein prediction tasks that use deep learning," *Brief. Funct. Genomic.*, vol. 20, no. 1, pp. 61–73, 2021.

[21] C. Xiao, Z. Zhou, J. She, J. Yin, F. Cui, and Z. Zhang, "PEL-PVP: Application of plant vacuolar protein discriminator based on PEFT ESM-2 and bilayer LSTM in an unbalanced dataset," *Int. J. Biol. Macromolecules*, vol. 277, 2024, Art. no. 134317.

[22] C. Yan, A. Geng, Z. Pan, Z. Zhang, and F. Cui, "MultiFeatVotPIP: A voting-based ensemble learning framework for predicting proinflammatory peptides," *Brief. Bioinf.*, vol. 25, no. 6, 2024, Art. no. bbae505.

[23] S. Ahmed et al., "ACP-MHCNN: An accurate multi-headed deep-convolutional neural network to predict anticancer peptides," *Sci. Rep.*, vol. 11, no. 1, 2021, Art. no. 23676.

[24] B. Han, N. Zhao, C. Zeng, Z. Mu, and X. Gong, "ACPred-BMF: Bidirectional LSTM with multiple feature representations for explainable anticancer peptide prediction," *Sci. Rep.s*, vol. 12, no. 1, 2022, Art. no. 21915.

[25] A. Ghulam, F. Ali, R. Sikander, A. Ahmad, A. Ahmed, and S. Patil, "ACP-2DCNN: Deep learning-based model for improving prediction of anticancer peptides using two-dimensional convolutional neural network," *Chemometrics Intell. Lab. Syst.*, vol. 226, 2022, Art. no. 104589.

[26] X. Yang, J. Jin, R. Wang, Z. Li, Y. Wang, and L. Wei, "CACPP: A contrastive learning-based Siamese network to identify anticancer peptides based on sequence only," *J. Chem. Inf. Model.*, vol. 64, pp. 2807–2816, 2023.

[27] X. Wang, Z. Zhang, and C. Liu, "iACP-DFSRA: Identification of anticancer peptides based on a dual-channel fusion strategy of ResCNN and attention," *J. Mol. Biol.*, vol. 436, no. 22, 2024, Art. no. 168810.

[28] Z. Lin et al., "Language models of protein sequences at the scale of evolution enable accurate structure prediction," *BioRxiv*, vol. 2022, 2022, Art. no. 500902.

[29] J. Bian, X. Liu, G. Dong, C. Hou, S. Huang, and D. Zhang, "ACP-ML: A sequence-based method for anticancer peptide prediction," *Comput. Biol. Med.*, vol. 170, 2024, Art. no. 108063.

[30] A. Tyagi et al., "CancerPPD: A database of anticancer peptides and proteins," *Nucleic Acids Res.*, vol. 43, no. D1, pp. D837–D843, 2015.

[31] G. Wang, X. Li, and Z. Wang, "APD3: The antimicrobial peptide database as a tool for research and education," *Nucleic Acids Res.*, vol. 44, no. D1, pp. D1087–D1093, 2016.

[32] S. Singh et al., "SATPdb: A database of structurally annotated therapeutic peptides," *Nucleic Acids Res.*, vol. 44, no. D1, pp. D1119–D1126, 2016.

[33] W. Shen, S. Le, Y. Li, and F. Hu, "SeqKit: A cross-platform and ultrafast toolkit for FASTA/Q file manipulation," *PLoS One*, vol. 11, no. 10, 2016, Art. no. e0163962.

[34] L. Fu, B. Niu, Z. Zhu, S. Wu, and W. Li, "CD-HIT: Accelerated for clustering the next-generation sequencing data," *Bioinformatics*, vol. 28, no. 23, pp. 3150–3152, 2012.

[35] P. Agrawal, D. Bhagat, M. Mahalwal, N. Sharma, and G. P. Raghava, "AntiCP 2.0: An updated model for predicting anticancer peptides," *Brief. Bioinf.*, vol. 22, no. 3, 2021, Art. no. bbaa153.

[36] B. Yang, L. Wang, D. Wong, L. S. Chao, and Z. Tu, "Convolutional self-attention networks," 2019, *arXiv:1904.03107*.

[37] Z. Lin et al., "Evolutionary-scale prediction of atomic-level protein structure with a language model," *Science*, vol. 379, no. 6637, pp. 1123–1130, 2023.

[38] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[39] U. Consortium, "UniProt: A hub for protein information," *Nucleic Acids Res.*, vol. 43, no. D1, pp. D204–D212, 2015.

[40] L. Dou, Z. Zhang, L. Xu, and Q. Zou, "iKcr_CNN: A novel computational tool for imbalance classification of human nonhistone crotonylation sites based on convolutional neural networks with focal loss," *Comput. Struct. Biotechnol. J.*, vol. 20, pp. 3268–3279, 2022.

[41] J. Xie et al., "Advanced dropout: A model-free methodology for bayesian dropout optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 4605–4625, Sep. 2022.

[42] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 2488–2498.

[43] A. G. Schwing and R. Urtasun, "Fully connected deep structured networks," 2015, *arXiv:1503.02351*.

[44] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[45] T. Van Laarhoven, "L2 regularization versus batch and weight normalization," 2017, *arXiv:1706.05350*.

[46] L. Prechelt, "Automatic early stopping using cross validation: Quantifying the criteria," *Neural Netw.*, vol. 11, no. 4, pp. 761–767, 1998.

[47] R. Liu, Z. Zhang, X. Fu, S. Yan, and F. Cui, "AIPPT: Predicts anti-inflammatory peptides using the most characteristic subset of bases and sequences by stacking ensemble learning strategies," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, 2023, pp. 23–29.

[48] Y. Li et al., "msBERT-promoter: A multi-scale ensemble predictor based on BERT pre-trained model for the two-stage prediction of DNA promoters and their strengths," *BMC Biol.*, vol. 22, no. 1, 2024, Art. no. 126.

[49] Y. Wang, Y. Zhai, Y. Ding, and Q. Zou, "SBSM-Pro: Support bio-sequence machine for proteins," *Sci. China Inf. Sci.*, vol. 67, no. 11, 2024, Art. no. 212106.
[50] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," 2018, *arXiv:1802.03426 1802*.
[51] T. L. Bailey, "STREME: Accurate and versatile sequence motif discovery," *Bioinformatics*, vol. 37, no. 18, pp. 2834–2840, 2021.

Shun Gao is currently working toward the master's degree with Hainan University, Haikou, China. His research interests include bioinformatics, Big Data, and machine learning.

Yan Xia is currently working toward the doctoral degree with Hainan University, Haikou, China. Her research interests include bioinformatics, medication, and machine learning.

Xingfeng Li received the master's degree from Tianjin University, China, in 2016, and the PhD degree from the Japan Advanced Institute of Science and Technology, in 2019. He is an associate professor with the School of Computer Science and Technology, Hainan University. His research interests include speech emotion recognition and machine learning.

Feifei Cui received the MS degree in computer application technology from Shandong University, Jinan, China, in 2012, and the PhD degree in bioinformatics from the University of Tokyo, Japan, in 2020. She is currently an associate professor with the School of Computer Science and Technology, Hainan University. Her research interests include bioinformatics, deep learning, and biological data mining.

Qingchen Zhang received the PhD degree in software engineering from the Dalian University of Technology, China, in 2015. He is currently a professor with the School of Computer Science and Technology, Hainan University. His main research interests include machine learning, medical Big Data, and blockchain.

Quan Zou received the bachelor's, master's, and PhD degrees from the Harbin Institute of Technology, in 2009. He is an associate professor with the School of Computer Science and Technology, University of Electronic Science and Technology of China. He was awarded the National Science Fund for Excellent Young Scholars in 2019. His research interests include applying machine learning methods to solve bioinformatics problems and using parallel/high-performance computing to address bioinformatics challenges.

Zilong Zhang received the PhD degree in bioinformatics from the University of Tokyo, Japan, in 2020. He is currently an associate professor with the School of Computer Science and Technology, Hainan University. He worked as a postdoctoral researcher with the University of Electronic Science and Technology of China. His research interests include bioinformatics, machine learning, and graph neural network.