Neural Networks 195 (2026) 108261

Contents lists available at ScienceDirect

Neural Networks

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

Full Length Article

CNNCaps-DBP: Leveraging protein language models with attention-augmented convolution for DNA-binding protein prediction

Ziyuan Yan [a,1], Aoyun Geng [a,1], Yazi Li [b], Jiajing Wang [a], Junlin Xu [c], Yajie Meng [d], Leyi Wei [e], Quan Zou [f], Zilong Zhang [a], Feifei Cui [a,*]

[a] School of Computer Science and Technology, Hainan University, Haikou, 570228, China
[b] School of Mathematics and Statistics, Hainan University, Haikou, 570228, China
[c] School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan, 430081, Hubei, China
[d] School of Computer Science and Artificial Intelligence, Wuhan Textile University, Wuhan, 430200, Hubei, China
[e] Centre for Artificial Intelligence driven Drug Discovery, Faculty of Applied Science, Macao Polytechnic University, Macao SAR, 999078, China
[f] Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu, 610054, China

ARTICLE INFO

Keywords:
DNA-binding proteins
Deep learning
ESM C
Attention augmented convolution
Capsule network

ABSTRACT

DNA-binding proteins (DBPs) are fundamental to many key cellular processes, possessing distinct binding domains, differential affinities for single- and double-stranded DNA structures, and playing roles in fundamental biological functions such as DNA replication and gene regulation. They are intimately linked to the pathological mechanisms of diseases like neurodegenerative disorders and cancers, making their prediction pivotal for unraveling protein function and disease mechanisms. However, conventional experimental techniques for DBP identification are temporally inefficient, labor-intensive, material-intensive and pricey. Existing DNA-binding protein prediction models either lack integration with pre-trained protein language models, primarily relying on manually constructed features, or despite utilizing pre-trained language models, fail to extract sufficiently effective information from the features generated by these models. Hence, a pressing imperative persists to devise robust computational frameworks capable of precise and efficient DBP delineation. In this study, we propose a novel deep learning-based method named CNNCaps-DBP for the accurate prediction of DBPs from primary sequence information. Our methodology incorporates the pre-trained protein language model ESM C and enhances the embeddings via an attention augmented convolution module. The extracted features are then passed through a hybrid deep learning framework consisting of Capsule network and MLP to construct the final predictive model. To optimize the model's training process, we applied a dynamic learning rate scheduler utilized for lessening the risk of premature convergence and enhance the robustness of the learning process. Experimental results show that CNNCaps-DBP significantly outperforms previous models in terms of predictive performance. To further validate the robustness of the proposed model, we evaluated it on additional independent datasets, where CNNCaps-DBP consistently outperformed state-of-the-art methods. In addition, we conducted two case studies to interpret the predictions of our model, which demonstrates the strong predictive capability for DBP identification. The source code used in this study is available at: https://github.com/YZYAlex/CNNCaps-DBP.

1. Introduction

The fundamental basis of many critical cellular processes lies in DNA-protein interactions (Zhang et al., 2022). Approximately 6–7 % of proteins within eukaryotes partake in interactions with DNA. These proteins feature unique DNA - binding domains and display distinct affinities toward single - stranded and double - stranded DNA (Attali et al., 2021; Gupta et al., 2021). The mechanisms for DNA - protein recognition comprise direct base–amino acid interactions as well as indirect contributions derived from conformational energy, which arises from DNA deformations and elasticity (Arora et al., 2023). DNA-binding proteins (DBPs) occupy extremely significant positions in the biology of living organisms (Dai et al., 2018; Castello et al., 2013; Wei et al., 2021; Shao et al., 2024), which engage in multiple key biological processes,

* Corresponding author.
E-mail address: feifeicui@hainanu.edu.cn (F. Cui).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.neunet.2025.108261
Received 12 August 2025; Received in revised form 10 October 2025; Accepted 25 October 2025
Available online 26 October 2025

encompassing DNA replication, gene regulation, repair, and modification (Liu & Yan, 2012; Ren et al., 2000; Abdelkader & Kim, 2024). The investigation into DNA - binding proteins, including TDP - 43, helicase chromodomain proteins, and those featuring methyl - CpG - binding domains, has significantly advanced the identification of pathologies such as neurodegenerative diseases and cancers (Lye & Chen, 2022; Alendar & Berns, 2021; Zhang et al., 2021; Wei et al., 2024; Jin et al., 2022). Therefore, accurate prediction of DBPs is paramount for elucidating the functional characterization of proteins. Substantial proteomic endeavors at scale, predicated upon biochemical paradigms, have been spearheaded to prognosticate DNA-binding proteins (DBPs), encompassing X-ray crystallography, microarray analysis, and chromatin immunoprecipitation. While these methods increased a number of experimentally determined DBPs, they are time-prohibitive, personnel-intensive, material-intensive and expensive, greatly limiting their scalability and applicability.

To overcome these constraints, algorithmic modalities have been increasingly employed to predict DBPs, proving accurate and cost – effective (Xu et al., 2021; Lin, 2024). Thus far, computational algorithms have been developed to predict DBPs, encompassing approaches grounded in machine learning and deep learning. Methods such as DPP-PseAAC (Rahman et al., 2018), iDNA-Prot|dis (Liu et al., 2014), iDNA-Prot (Lin et al., 2011), PseDNA-Pro (Liu et al., 2015), IKP-DBPPred (Qu et al., 2017), PSFM-DBT (Zhang & Liu, 2017), TargetDBP (Hu et al., 2019), Deep-WET (Mahmud et al., 2024), employ various feature extraction methods and model architectures to predict DBPs. To address the issue of high computational cost associated with using traditional evolutionary information, some research used PSSM as an evolutionary feature for prediction tasks. Sun et al. (Sun et al., 2024) combined a novel protein feature extraction approach with PSSM matrices and commonly used feature extraction methods for accurately predicting DBPs. In LBi-DBP (W. Zeng et al., 2024), authors fed the PSSM and other features into a model leveraging Bi-LSTM for DBP prediction. However, the process of extracting the protein PSSM matrix is highly complicated and requires considerable computational power.

In addition to evolutionary information, some studies also utilized other features (Wang et al., 2015). In DRBPPred-GAT (Zhang et al., 2024), one evolutionary information feature, sequence information feature and six physicochemical property features serve to capture the characteristics of DNA-binding. Deep-WET (Mahmud et al., 2024) is another proposed DBP predictor in which three word embedding approaches are used for feature encoding, with these features dynamically fused by the DE algorithm. These researches have underscored that choosing appropriate feature extraction methods is a crucial factor in accurately predicting DBPs. Many studies have utilized pre-trained models like ESM 2 for the classification of proteins (Luo et al., 2025; Geng et al., 2025; Lv et al., 2025; Li et al., 2024; Chen et al., 2025). Cordoves-Delgado et al. (Cordoves-Delgado & Garcia-Jacas, 2024) used a framework based on ESMFold and ESM 2 to predict antimicrobial peptides. In predicting IL13-inducing peptides, Kurata et al. (Kurata et al., 2024) integrated various machine learning and deep learning methods with ESM 2. Zeng et al. (S. Zeng et al., 2024) used ESM 2 to fine-tune the parameters, improving the prediction of signal peptides. And Sanaboyana et al. (Sanaboyana & Elcock, 2024) used AlphaFold2-predicted protein structures to improve the signal and transit peptide predictions. Although these studies have extracted rich protein features using pre-trained models or other methods, the traditional convolutional modules they employed in the subsequent extraction of effective information from these features have resulted in some loss of information.

Since the pooling layers in convolutional networks can't preserve the hierarchical structure information of proteins, some studies have started using capsule networks for protein classification to tackle this issue. EmbedCaps-DBP (Naim et al., 2023) leveraged protein sequence embedding combined with a capsule network for DBP prediction. Conversely, Mursalim et al. (Mursalim et al., 2025) adopted a hybrid approach integrating GRU, Bi-LSTM, and capsule network architectures to enhance the predictive performance of DBPs. In ACP-CapsPred (Yao et al., 2024), authors used a capsule network-based framework to improve the identification and functional prediction of anti-cancer peptides. And pNPs-CapsNet (Akbar et al., 2025) integrated the pre-trained model and weighted multi-image features with capsule network. These studies utilized capsule networks to ensure the preservation of spatial location and hierarchical relationships of features; however, the characteristics they extracted limited the performance of their models, as the methods of feature extraction were overly traditional or the extracted features were too simplistic.

Although existing methods have achieved promising results in DBP prediction, they still largely lack the integration of pre-trained protein language models, which have been broadly adopted for the prediction of other proteins. Most approaches rely primarily on handcrafted features derived from sequence information, physicochemical properties and evolutionary information, without fully exploiting the powerful sequence representation and contextual understanding capabilities offered by large-scale protein language models. And research that utilizes pre-trained language models have failed to extract sufficiently effective information from the features derived from these models. To tackle this shortcoming, we design an innovative predictor named CNNCaps-DBP, which incorporates the pre-trained protein language model ESM C with attention-augmented convolution modules (augmented Conv) and a capsule network (CapsNet). Initially, we use ESM C to encode protein sequences and extract sequence features as input for augmented Conv. Augmented Conv (Bello et al., 2019) is capable of capturing both the local and global information of protein sequence that contribute to advancing the prediction workflow. These outputs are then, these outputs are passed to CapsNet, which serves to model the mutual relationships between features. Finally, input the output of the CapsNet into a multi-layer perceptron (MLP) to obtain the prediction result. Compared to previous studies, the features extracted using ESM C are richer and more comprehensive. And the augmented Conv we employed allows for the simultaneous extraction of both local and global features of proteins, addressing the issue of effective feature loss that arose from the sole use of traditional convolutions in earlier studies. Additionally, the CapsNet is capable of fully utilizing the extracted features from the augmented Conv while preserving their spatial positions and hierarchical relationships, which were overlooked in prior research. The seminal innovations of the present work can encapsulate:

1) A pre-trained language model tailored for proteins is employed to encode protein sequences, fully leveraging its representational power to automatically capture structural and evolutionary information embedded within the sequences.
2) The improved attention-augmented convolution module addresses the limitation of conventional convolutional networks that primarily focus on local features. By incorporating an attention mechanism, it captures deeper relationships among embeddings.
3) On independent test datasets, CNNCaps-DBP exhibits elevated functionality relative to the state-of-the-art methods (SOTA) models across multiple evaluation metrics. Furthermore, its predictive capability is validated on an additional dataset.
4) To enhance model interpretability, we conducted two case studies by visualizing attention weights associated with DBPs, providing insight into the decision-making process of the model.

2. Materials and methods

2.1. The framework of cnncaps-dbp

CNNCaps-DBP is constructed through a sequence of steps, including data preprocessing, ESM C feature encoding, model training, and final prediction, as illustrated in Fig. 1. At the initial phase, ESM C embedding

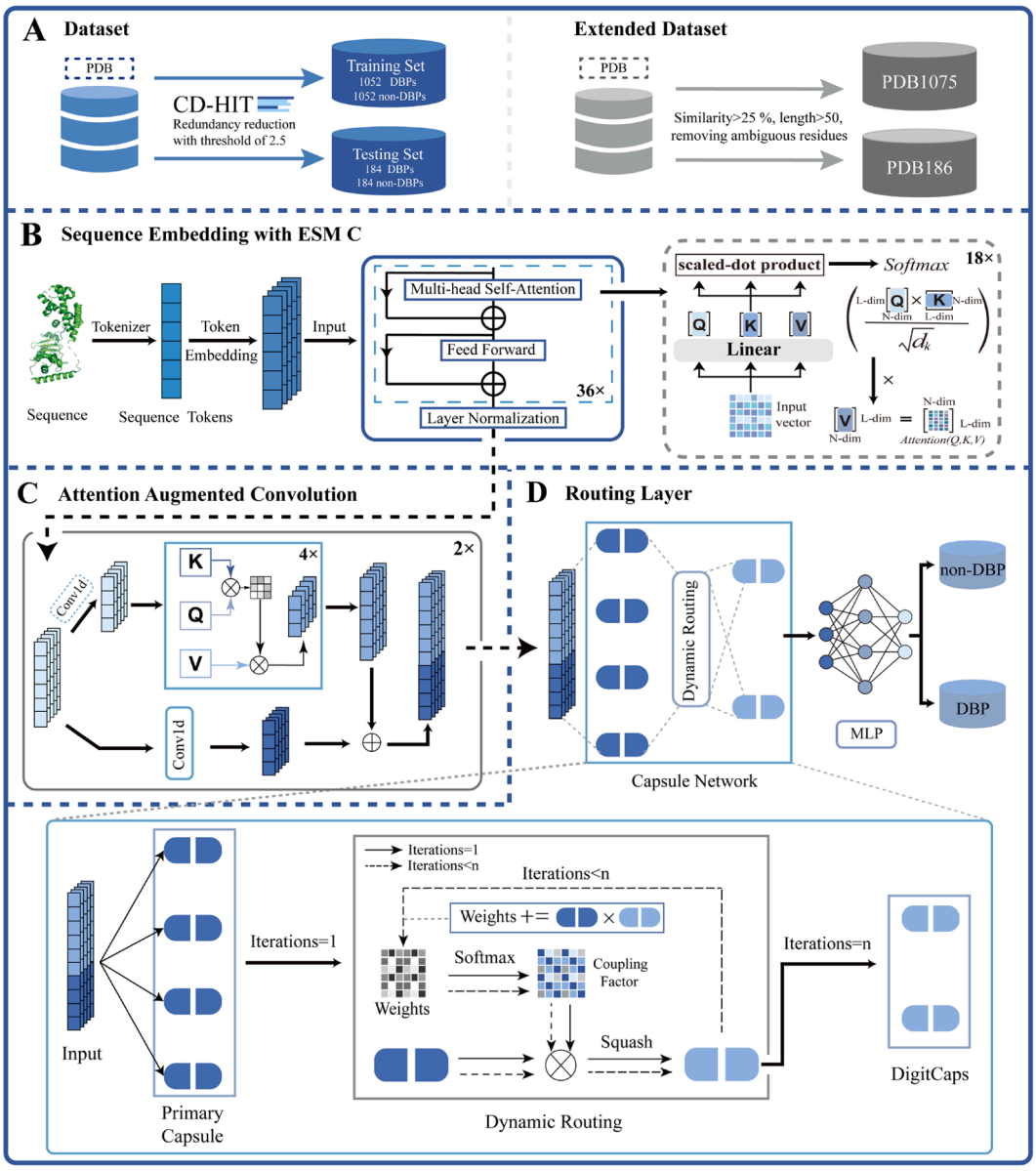

Fig. 1. The overall architecture of CNNCaps-DBP. (A) Dataset and Extended Dataset: The Dataset employed a lower CD-HIT threshold of 0.25 to eliminate redundant protein chains, while also excluding protein chain sequences with fewer than 50 residues or unknown residues. Positive and negative datasets were randomly chosen to form training datasets containing 1052 DBPs and 1052 non-DBPs, alongside independent test datasets with 148 DBPs and 148 non-DBPs. The Extended Dataset includes two subsets designated as PDB1075 and PDB186. Protein sequences derived from these datasets were retrieved from the PDB database and subjected to rigorous filtering procedures. (B) Sequence Embedding with ESM C: After entering ESM C, protein sequences are first tokenized, then passed into 36 Transformer layers consisting of multi-head attention mechanisms, FNN and ResNet, and the final sequence features are obtained through layer normalization. (C) Attention Augmented Convolution: This module consists of a 1D convolution which captures local features and a multi-head attention that captures global features, and fuses the captured features together, enabling the module to fully capture sequence characteristics. (D) Routing Layer: This module consists of CapsNet and MLP. The CapsNet consists of a primary capsules layer and a digitcaps layer, with the two capsule layers connected by dynamic routing that can continuously optimize the connection strength between the two layers of capsules during model training, preserving the spatial hierarchy of features through weighted summation. The MLP receives the output of the capsule network, and the input data goes through multiple fully connected layers to obtain the prediction result.

feature encoding technique were adopted. And then the features were fed to the deep learning model based on augmented Conv, CapsNet and MLP to construct the ultimate prediction models by leveraging the training and independent test datasets. Specifically, the augmented Conv incorporates a multi-head attention mechanism and 1D convolution, combining the ability of convolution to capture local features with the attention mechanism's ability to capture global features. The pooling layer of conventional convolution losses the spatial hierarchical structure of proteins, but CapsNet effectively addresses this issue. Eventually, the model's performance was appraised by diverse metrics, encompassing accuracy, sensitivity, specificity, MCC (Matthews Correlation Coefficient), AUC (Area Under the Curve) and F1 score.

2.2. Datasets

Developing a reliable, comprehensive and high-quality dataset is extremely crucial for constructing an effective and stable model. In this research, we compiled datasets from previous research which were gathered and mainly utilized by Hu et al. (Hu et al., 2019), who retrieved both DBP and non-DBP chains from the Protein Data Bank (PDB). The dataset was first processed by applying a CD-HIT (Chen et al., 2025) threshold of 0.25 to eradicate redundant protein chains, which

can effectively reduce the interference of similarity between sequences, ensuring the diversity of the dataset and providing more representative samples for model training. Sequences with <50 residues or unknown residues were then excluded. Positive and negative samples were allocated to the training set and an independent test set in a random manner, that reduces the noise in the sequential data, ensuring that the information learned by the model from the features is more reliable. In addition, compared to the currently widely used benchmark datasets of DNA-binding proteins, the DNA-binding proteins in this dataset are more novel. The training set included 1052 DBPs and 1052 non-DBPs, whereas the independent set contained 148 DBPs and 148 non-DBPs. Further details about the dataset partitioning are available in Hu et al. (Hu et al., 2019). Additionally, two widely acknowledged benchmark datasets from earlier studies, PDB1075 and PDB186, were utilized in the generalization experiment. Protein sequences within these datasets retrieved from the latest iteration of the Protein Data Bank (PDB) (Naim et al., 2023). PDB1075 was established through the removal of protein sequences with a length of <50 amino acids and the PISCES 40 software was used to filter out protein sequences exhibiting a similarity exceeding 25 %. PDB186 was constructed via deletion of protein sequences ⟨ 60 amino acids in length and NCBI's BLASTCLUST software was utilized to filter out protein sequences with ⟩ 25 % similarity. Using these widely recognized benchmark datasets in the generalization experiments will better illustrate the generalizability of our model. Detailed information about these datasets is presented in Table 1.

2.3. Feature representation using ESM C

Developed by Meta AI Research, the ESM C protein language model represents a deep learning-driven solution for forecasting protein structure and function. By integrating transformer architectures and evolutionary information, this method processes raw sequence data to decode protein folding patterns and functional roles, aligning with state-of-the-art advancements in computational structural biology. Compared to the previous generation ESM 2 (Lin et al., 2023), ESM C's inference speed has significantly improved while maintaining higher accuracy. It is a pre-trained model for biological sequences, such as the amino acid sequences of peptides or proteins. The ESM C is based on the Transformer architecture which includes Multi-head Attention Mechanism, Feedforward Neural Network (FNN) and Residual Network (ResNet) and undergoes unsupervised pre-training on a large-scale protein sequence dataset. Via learning the evolutionary patterns of protein sequences and the relationship between structure and function, the ESM C can generate corresponding latent vectors for protein sequences, encoding their structural and functional attributes. Additionally, the representations can serve as computational substrates for specialized biological workflows. ESM C effectively instantiates deep learning paradigms to assimilate protein sequences in a manner isomorphic to natural language processing (NLP), leveraging transformer architectures to encode sequential dependencies and extract contextual embeddings. It is a new powerful and versatile tools for protein scientific studies. In this study, we use ESM C-600 m to encode DBPs, where each amino acid in the sequence is represented by a 1152-dimensional vector. These encoded representations are subsequently input into our model. And we did not perform fine-tuning on ESM C in any training processes or experiments.

Table 1
Datasets used for training, independent test and generalization experiment.

Dataset	DBPs	Non-DBPs
Training Data	1052	1052
Independent Test Data	148	148
PDB1075	525	550
PDB186	93	93

2.4. Attention augmented convolution (augmented conv)

As a form of deep learning model, Convolutional Neural Networks (CNNs) are frequently utilized in applications like image and video recognition, alongside natural language processing tasks (LeCun et al., 2015; Grinblat et al., 2016; Yan et al., 2025; D. Zhang et al., 2025; Liu et al., 2019). Within CNN frameworks, deeper convolution layers facilitate the extraction of high-dimensional features through sliding convolutional kernels that adopt different hyperparameter setups on the upper layers' outputs. However, ordinary CNNs can only capture local features of the data, which still limits their effectiveness in addressing certain problems. Thus, we use the Attention augmented Convolution (augmented Conv) (Bello et al., 2019) to replace the ordinary CNNs.

The Attention augmented Convolution published by Bello et al. (Bello et al., 2019) is used for processing two-dimensional data, which is not suitable for the protein sequences. Therefore, we replaced the 2D convolutions with 1D convolutions and slightly modified the multi-head attention mechanism to ensure that the data from the two channels can be concatenated correctly.

The following notation is adopted: L and F_{in} stand for the length and input filter count of an activation map. For multihead-attention (MHA), N_h, d_v and d_k respectively indicate the number of heads, value depth, and query/key depth. It is further assumed that N_h evenly divides both d_v and d_v with d_v^h and d_k^h representing the depth of values and queries/keys per attention head.

Receiving an input tensor with shape (L, F_{in}), we reshape it into a matrix $X \in R^{L\times F_{in}}$ and implement the multi-head attention mechanism as described in the Transformer architecture (Vaswani et al., 2017). For an individual attention head h, the output of the self-attention operation can be defined as:

$$O_h = Softmax\left(\frac{(XW_q)(XW_k)^T}{\sqrt{d_k^h}}\right)(XW_v) \tag{1}$$

where W_q, $W_k \in R^{F_{in}\times d_k^h}$ and $W_v \in R^{F_{in}\times d_v^h}$ denote trainable linear projection matrices that convert the input X into queries vectors $Q{=}XW_q$, keys vectors $K{=}XW_k$ and values vectors $V{=}XW_v$. The outputs of all attention heads are then aggregated via concatenation and further processed through an additional projection step:

$$MHA(X) = Concat[O_1, \ldots, O_{Nh}]W^O \tag{2}$$

where $W^O \in R^{d_v\times d_v}$ is a learned linear transformation. $MHA(X)$ is reconfigured as (L, d_v) tensors for spatial coherence. Notably, multi-head attention demands O ($L^2 d_k$) computation and memory O ($L^2 N_h$) due to per-head attention map retention. The output of the 1D convolution is formulated as:

$$Conv1d(X) = X{\cdot}K + b \tag{3}$$

where $*$ represents the convolution operation, K is convolution kernel weights and b is bias term. Mathematically, consider a standard convolution operator defined by kernel K, with F_{in} input channels and F_{out} output channels. Derived attention-augmented convolution is defined by:

$$AAConv(X) = Concat[Conv1d(X), MHA(X)] \tag{4}$$

where Concat represents concatenate the outputs of 1D convolution and multi-head attention.

In this study, we connect two enhanced convolutional modules linearly. The first convolutional module has a kernel size of 10, an input dimension of 1152, and an output dimension of 510. The second module has a kernel size of 5, an input dimension of 510, and an output dimension of 200. Two modules both have 4 attention heads.

2.5. Capsule network (CapsNet)

Capsule network is a deep neural network architecture put forth by Jaiswal et al. (Jaiswal et al., 2018). The typical architecture incorporates an input layer, followed by a convolutional layer, a primary capsule layer, a digitcaps layer, a fully connected layer, and culminates in an output layer. Within the CapsNet architecture, to mitigate error propagation, global parameters are universally employed across layers, whereas parameter updates are governed by an iterative dynamic routing mechanism (Zhou et al., 2024). In this research, we use the primary-caps layer and digit-caps layer of CapsNet.

The forward propagation across capsule layers follows these equations:

$$u_{j|i} = w_{ij} \cdot u_i \tag{5}$$

$$t_i = \sum_i c_{ij} \cdot u_i \tag{6}$$

where u_i denotes the i- th capsule within the primary capsule layer, w_{ij} corresponds to the associated weight matrix, and $u_{j|i}$ signifies the input prediction vector input from the primary capsule layer to the digit capsule layer, which is generated through the dynamic routing process. t_i serves as the weighted aggregation matrix of $u_{j|i}$ and c_{ij}, which can be through the adjustment of c_{ij}. After applying a squashing function to compress t_i, an output vector v_j with a magnitude between 0 and 1 is derived, which can be formulated as:

$$v_j = \frac{\| t_i\|^2}{1+\| t_i\|^2} \cdot \frac{t_i}{\| t_i\|} \tag{7}$$

CapsNet evaluates capsule agreement through the dot product $u_{j|i} \cdot v_j$, while simultaneously refining connection weights b_{ij} between primary and digit capsule layers. A larger value of the coupling coefficient u_{ij} indicates a higher degree of similarity between the representative capsules. The calculation formulas of b_{ij} and u_{ij} can be expressed as:

$$b_{ij} = b_{ij} + u_{j|i} \cdot v_j \tag{8}$$

$$u_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})} \tag{9}$$

Compared to traditional convolution, which compresses features through max pooling, leading to the loss of spatial hierarchical information, making it difficult to capture the structural hierarchical relationships of complex biomolecules, the CapsNet replaces pooling with a dynamic routing algorithm, utilizing the inner product of the prediction vectors from the primary capsule network $u_{j|i}$ and the vectors v_j generated by the digit capsule layer to dynamically adjust the weights of feature transmission c_{ij}, preserving the spatial locations and hierarchical relationships of features. It provides an efficient solution for high-dimensional feature classification problems in bioinformatics.

The algorithm introduced in Algorithm 1 outlines the process of dynamic routing. This algorithm takes primary capsules as input while introducing an initial weight b. It then uses Softmax to convert b into coupling factors. Next, the primary capsules are multiplied by the coupling factors and normalized using the Squash function to obtain digit capsules. If the number of routing iterations has not reached n, the initial weight b ($b = b +$ *primary capsules* $\times$ *digit capsules*) is updated, and the above operation is repeated.

In this study, the number of input capsules used in the capsule network is 200, the number of output capsules is 16, the dimension of the input capsules is 122, the dimension of the output capsules is 16, and the number of dynamic routing iterations is 3.

2.6. Model training

2.6.1. Dynamic learning rate scheduling

In deep learning, the choice of optimizer and its learning rate are crucial to performance: excessive rates impede learning, while insufficient rates delay convergence (Donâncio et al., 2024; H.-Q. Zhang et al., 2025). In mathematics, the learning rate is usually represented by η, and its update formula is:

$$\theta = \theta - \eta \cdot \nabla J(\theta) \tag{10}$$

where θ is the model parameter, and $\nabla J(\theta)$ represents the gradient of the loss function $J(\theta)$.

In this research, a dynamic learning rate scheduler is employed to further optimize the model's training process. ReduceLROnPlateau in torch is used as a dynamic learning rate scheduler, reducing the learning rate to half of the original when the number of training rounds surpasses five, yet the loss score on the validation set fails to decrease. This learning rate scheduler mechanism facilitates the model's ability to surmount local minima in the optimization landscape. Consequently, it enables our model to explore broader regions of the parameter space, enhancing the likelihood of converging to more globally optimal solutions. This iterative adjustment of the learning rate serves as a crucial

Algorithm 1
Dynamic Routing for Capsule Layer.

```
Input: Vector of each input capsule u_predict
       Routing iteration count n_iterations
Output: v ←Activation vector of the output capsule
b ← zeros[input_caps, output_caps]
for each forward pass do
  b.zero_()
  Initialize b = 0
  for r = 1 : n_iterations do
    compute coupling coefficients: c ← softmax(b)
    c_expanded ← c.unsqueeze(2)
    s ← ∑(c ⊗ u_predict)
    v ← squash(s)
    if r < n_iterations then
      for i in 1 to input_caps do
        for j in 1 to output_caps do
          agreement ← dot(u_predict, v)
          b ← b + sum(agreement)
        end for
      end for
    end if
  end for
end for
return v
```

regularization strategy, mitigating the risk of premature convergence and promoting more robust and efficient optimization during the training process.

2.6.2. Loss function

While using deep learning models to deal with binary classification problems, the choice of loss function plays a pivotal role in the optimization direction and final performance of the model. In this research, we use the Binary Cross-Entropy (BCE) as the loss function and sigmoid as the activation function. The expression of BCE loss is linearly separable, which makes it easy to efficiently compute gradients through back-propagation and it works well with the layered nonlinear transformations of neural networks. In addition, BCE measures the cross-entropy between the true distribution and the predicted distribution. Minimizing cross-entropy minimizes the KL divergence between the data distribution and the model distribution, allowing the model process to not just center on training set adaptation, while implicitly optimizing the overall divergence within the predicted distribution and the true distribution during the optimization, enhancing generalization ability. The formula for the binary cross-entropy loss function for batch data is as follows:

$$\mathscr{L}_{BCE} = -\frac{\sum_{i=1}^{N}[y_i \cdot \log\widehat{y}_i + (1-y_i)\cdot\log(1-\widehat{y}_i)]}{N} \tag{11}$$

where N denotes the overall count of samples within the batch, i serves to index each sample, and y_i stands for the actual label of the i th sample, $\widehat{y}_i$ represents the predictive probability output by the model for the i th sample.

2.6.3. Hyperparameter

In this research, the deep-learning-based predictor was employed for model training and performance evaluating on Python 3.10.0 and pytorch 2.2.0 to implement them. We adopted refined parameter configurations for the prediction algorithm, including a batch size 128, 4 hidden layers, learning rate 0.0005 and dropout rate 0.4. All experiments utilized an NVIDIA A100 GPU.

2.7. Evaluation metrics

The performance of CNNCaps-DBP was evaluated using six standard binary classification metrics (Wang et al., 2024; Xie et al., 2025; Huang et al., 2025; Huang et al., 2024; Huang et al., 2023): accuracy (ACC), sensitivity (Sen), specificity (Spe), Matthew's Correlation Coefficient (MCC), precision (Pre), F1 Score (F1) and Area Under the Curve (AUC). In this research, all our indicators were obtained through five-fold cross-validation (5-fold-CV).

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

$$Sen = \frac{TP}{TP + FN} \tag{13}$$

$$Spe = \frac{TN}{FP + TN} \tag{14}$$

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}} \tag{15}$$

$$Pre = \frac{TP}{TP + FP} \tag{16}$$

$$F1\ score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{17}$$

$$TPR_i = \frac{TP_i}{TP_i + FN_i} \tag{18}$$

$$FPR_i = \frac{FP_i}{FP_i + TN_i} \tag{19}$$

$$AUC = \sum_{i=0}^{n} \frac{(FPR_{i+1} - FPR_i) \times (TPR_i + TPR_{i+1})}{2} \tag{20}$$

where TP, FP, TN, FN denote the counts of true positives, false positives, true negatives, and false negatives, respectively. TP_i, FP_i, TN_i, FN_i represent the numbers of true positives, false positives, true negatives, and false negatives, respectively, at the i th classification threshold. TPR_i and FPR_i represent the true positive rate and false positive rate at the i th classification threshold.

3. Results and discussion

3.1. Performance of different manual feature extractions and pre-trained models

To evaluate the utility of different features for protein sequence characterization, we employed three manual feature extractions (BE, AAindex and BLOSUM62), one secondary structure feature(SecStr) extracted using DSSP (Hekkelman et al., 2025), and four pre-trained model (ESM 2–150 m, ESM 2–650 m, ESM C-300 m, ESM C-600 m) to provide sequence embeddings. And we also concatenated the features of ESM C-600 m with manually extracted features and secondary structure feature, aiming to explore whether these traditional features can complement the pretrained model. Secondly, the CNNCaps model employs 5-fold cross-validation (CV). The training set is split into five folds. Per iteration, four folds are used for training and the fifth serves as the validation set. Final model performance is assessed by aggregating outcomes across all five-fold cross-validation applied to the independent test set. The experimental results of the independent test data were shown in Tables S1 and S2, while the bar charts of each indication were depicted in Figs. 2 and 3.

As can be seen in Table S1 and Fig. 2, the performance of the pre-trained models is significantly better than of the manual feature extractions and each pre-trained models achieved satisfactory predictive result. It is evident that relying solely on manual feature extraction allows the model to capture only local characteristics, which limits its overall performance.

As can be seen in Table S2 and Fig. 3, the combination of ESM C+BE achieved the highest values in ACC, AUC, MCC, and Pre. The combination of ESM C+BE+BLOSUM62 achieved the highest values on Sen and F1. The combination of ESM C+AAindex achieved the highest values on Spe. Through comprehensive comparison, the BE feature exhibits the best complementary effect with pre-trained models.

Through comparison, the pre-trained model ESM C-600 m achieved the best performance in term of ACC, AUC, Sen, MCC and F1, eclipsing all competing pre-trained models and combined features. Specifically, Spe and Pre are metrics used to measure the accuracy of negative class recognition and the conservativeness of positive class predictions, respectively. The ESM 2–150 m model and ESM C+BE+BLOSUM62 both achieve the higher values for these two metrics, demonstrating their strong ability to identify non-DNA binding proteins and high conservativeness when recognizing DNA binding proteins. However, they both perform worse than ESM C-600 m in other metrics. In contrast, ESM C-600 m offers a more balanced and robust overall performance, particularly excelling in sensitivity compared to them, making it more suitable for tasks requiring high sensitivity for DNA binding protein predictions. Therefore, we choose ESM C-600 m to characterize protein sequences.

3.2. Analysis of architectural components and parameter sensitivity

To study the impact of model architectural and hyperparameters on model performance, we conducted ablation experiment, replaced the

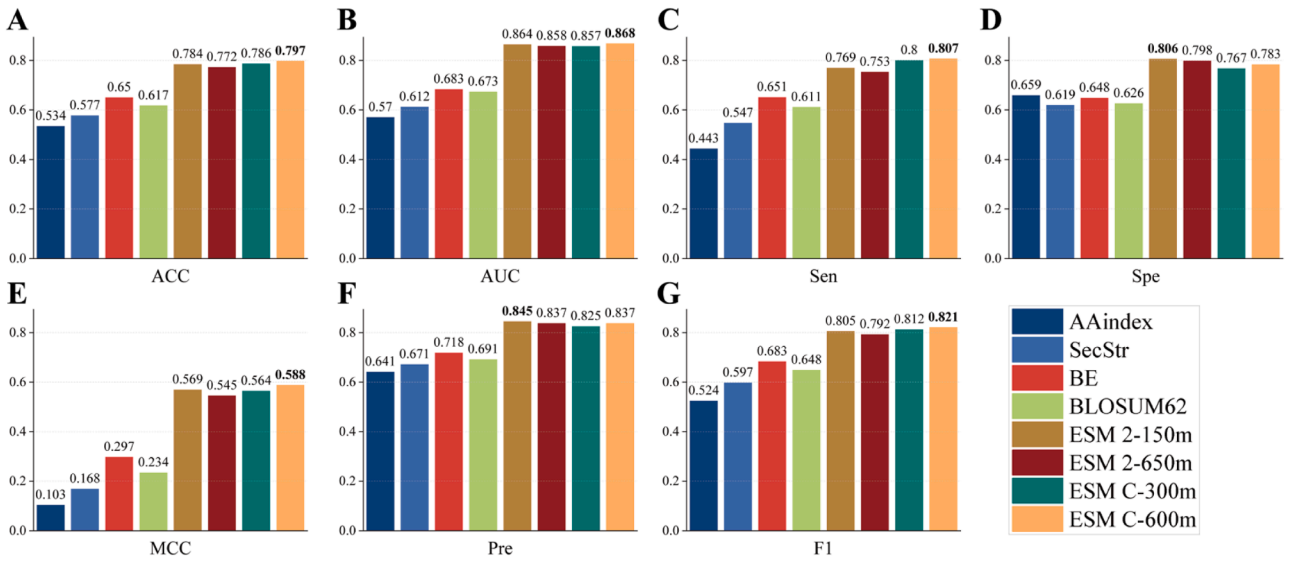

Fig. 2. Model performance using different encoding methods. (A) Bar chart of ACC on different features of CNNCaps-DBP and the highest is ESM C-600 m. (B) Bar chart of AUC on different features and the highest is ESM C-600 m. (C) Bar chart of Sen on different features and the highest is ESM C-600 m. (D) Bar chart of Spe on different features and the highest is ESM 2–150 m. (E) Bar chart of MCC on different features and the highest is ESM C-600 m. (F) Bar chart of Pre on different features and the highest is ESM 2–150 m. (G) Bar chart of F1 on different features and the highest is ESM C-600 m.

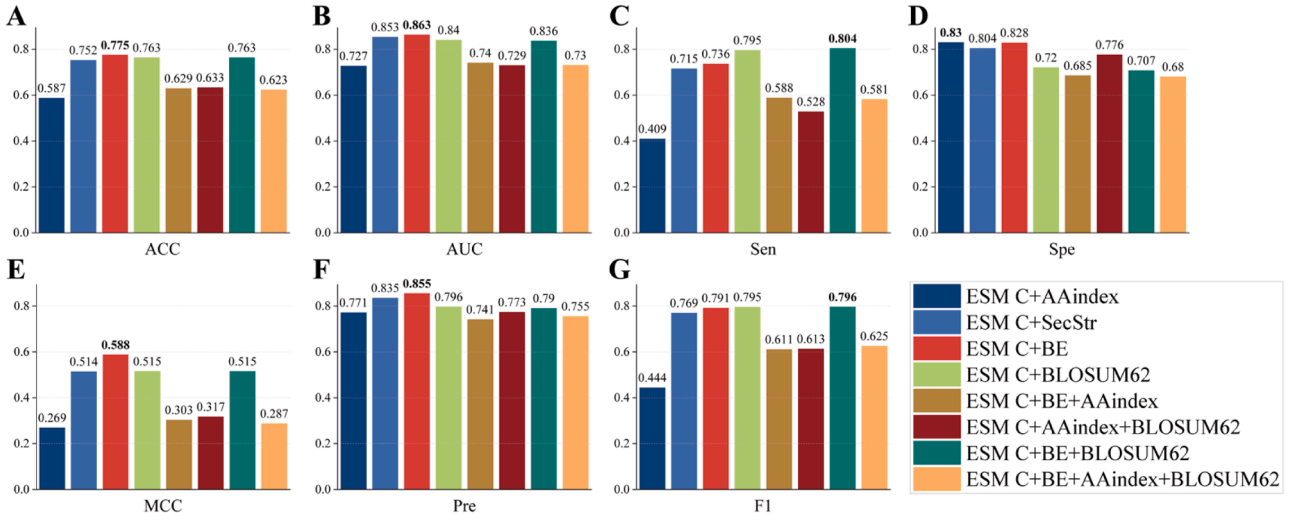

Fig. 3. Model performance using different multiple encodings, the parameter count of ESM C is 600 m. (A) Bar chart of ACC on different features of CNNCaps-DBP and the highest is ESM C+BE. (B) Bar chart of AUC on different features and the highest is ESM C+BE. (C) Bar chart of Sen on different features and the highest is ESM C+BE+BLOSUM62. (D) Bar chart of Spe on different features and the highest is ESM C+AAindex. (E) Bar chart of MCC on different features and the highest is ESM C+BE. (F) Bar chart of Pre on different features and the highest is ESM C+BE. (G) Bar chart of F1 on different features and the highest is ESM C+BE+BLOSUM62.

augmented Conv with ordinary convolution, and modified the batch size and optimizer. The focus of the research is to measure the ACC scores of different combinations while adjusting the model for the purpose of deriving an optimal architectural arrangement.

3.2.1. *Ablation experiment*

To gain a clearer understanding of how each module contributes to the model's overall performance, we devised an ablation experiment. Concretely, starting from the complete model, we progressively eliminated individual modules, aiming to observe how these changes impact model performance.

As shown in Table S3 and Fig. 4A, CNN1 represents the first augmented Conv, CNN2 represents the second augmented Conv, and Caps represents the capsule network. The findings demonstrate that the full-model configuration surpasses all ablation versions across all evaluation metrics except for Spe, exhibiting substantial performance advantages. Specifically, when the augmented Conv module is removed, the model's Sen, MCC and F1 show a significant decline, indicating that this module plays an important role in capturing multi-scale features. When removing the capsule network module, the evaluation metrics except for Spe all drop show an extremely significant decline, which undoubtedly indicating that capsule network module plays a key role in various aspects of the model. Overall, demonstrating strong capabilities in feature extraction and complementarity, the complete CNNCaps-DBP model confirms that maintaining its intact design is vital for peak practical performance.

3.2.2. *Performance of augmented conv and ordinary convolution*

To investigate whether augmented Conv offers better performance than ordinary convolution, we designed a comparative experiment.

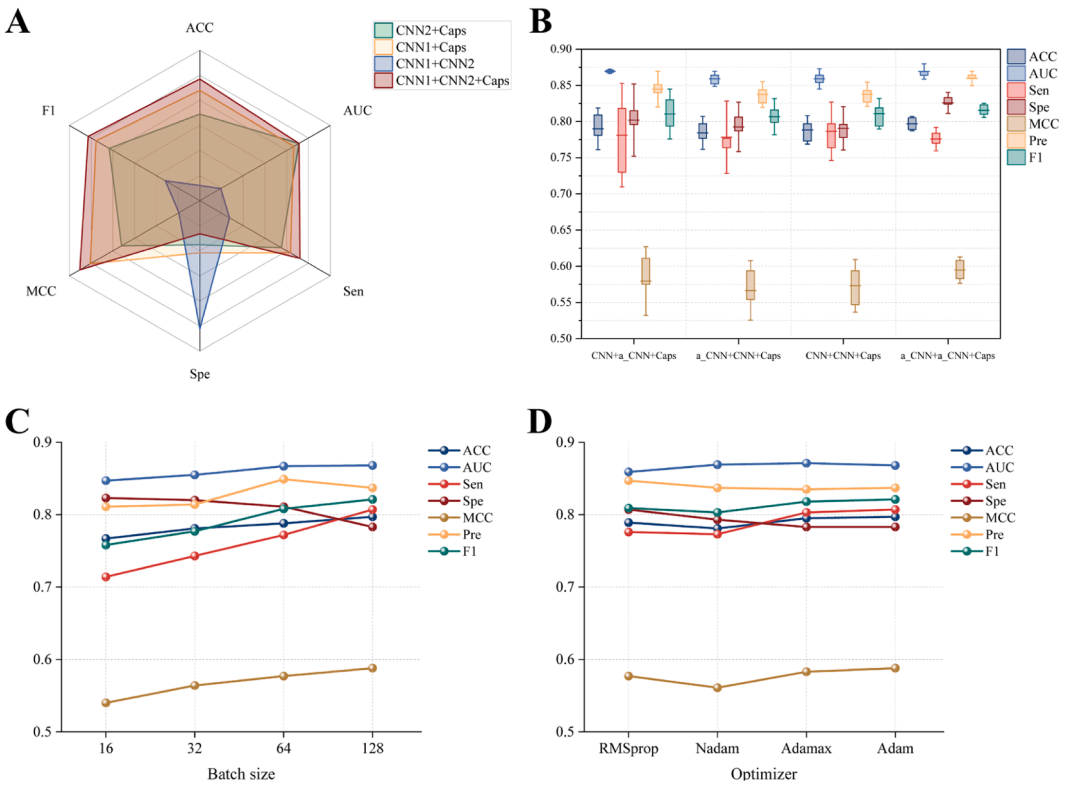

Fig. 4. Impact of architectural modifications and hyperparameter adjustments on model performance. (A) The radar chart of various metrics for CNNCaps-DBP under different ablation conditions, in which the complete model attained peak performance on all metrics except for Spe. (B) The box plot of CNNCaps-DBP in the convolutional comparison experiment, where the confidence intervals of the original model are smaller than those of models with different layer configurations across almost all metrics. (C) The line chart of various metrics of CNNCaps-DBP across different batch size. (D) The line chart of various metrics of CNNCaps-DBP under different optimizers.

Specifically, we sequentially replace the two augmented Conv module with ordinary convolution module.

As shown in Table S4 and Fig. 4B, where a_CNN represents the augmented Conv and CNN represents the ordinary convolution, the original model achieved the highest values on ACC, Sen, MCC and F1, demonstrating significant advantages. In addition to this, in Fig. 4B, the confidence intervals of various metrics for the original model in five-fold cross-validation are almost all smaller than those of models with other layer configurations. This indicates that the augmented Conv module equips the model for diverse data, enhancing its stability and generalization.

3.2.3. *Analysis of parameter sensitivity*

Initially, the effects of different batch size on the performance of the model were assessed, as illustrated in Table S5 and Fig. 4C. The results indicate that an increase in batch size enhances metrics such as ACC, Sen, AUC, MCC and F1. In particular, when the batch size is set to 128, the best ACC of 0.797 was achieve. Even though changing batch size doesn't really affect the ACC much, we've seen an important boost in Sen, MCC and F1 as the batch size increases which shows that this change has impacted the model's holistic efficacy.

Subsequently, we investigated how the choice of optimizer influenced the performance of the model, as shown in Table S6 and Fig. 4D The results indicate that the change of optimizer enhances the ACC, Sen and F1. And when choosing Adam as the optimizer, the best ACC of 0.797 was achieve. However, the experiment results show that although changes in the optimizer lead to certain improvements in the model's ACC, Sen and F1, they do not yield decisive consequences for the comprehensive performance of the model.

3.3. *UMAP visualization characterizes model effectiveness*

As shown in Fig. 5, To evaluate the model's capacity for learning and generalization across data distributions, feature spaces from each module of the CNNCaps-DBP model were visualized using Uniform Manifold Approximation and Projection (UMAP) (McInnes et al., 2018; Li et al., 2025). This dimensionality reduction technique projects high-dimensional features onto lower-dimensional manifolds, enabling clear visualization of distribution patterns. The resulting four subplots specifically highlight the distribution characteristics of positive samples within the UMAP space. Within the four subplots, positive samples' UMAP-space distribution features are visualized in red, whereas negative samples are shown in green.

Following the processing sequence within the model, we visualized the distribution of positive and negative samples step by step. Fig. 5A and B illustrate the distribution patterns of ESM C features after the front and back augmented Conv processing (CONV 1 and CONV 2). Fig. 5C and D show the distribution characteristics of the augmented Conv output features processed by CapsNet and MLP respectively. Sequential observation reveals that each additional module progressively refines the encoded features, thereby enhancing the separability of positive and negative samples in contrast to the preceding stage. This further confirms the significance of each module's design and its contribution to enhancing the overall model performance. This not only demonstrates the augmented Conv's feature extraction capabilities and the CapsNet's maximal retention of spatial hierarchical features, but also indicates the comprehensiveness, accuracy and effectiveness of the sequential features extracted by the ESM C model. Specifically, after passing through the CapsNet, the features show a distinct separation evident in sample classification. This highlights the CapsNet's impressive ability to differentiate between them, and it also indicates the importance of

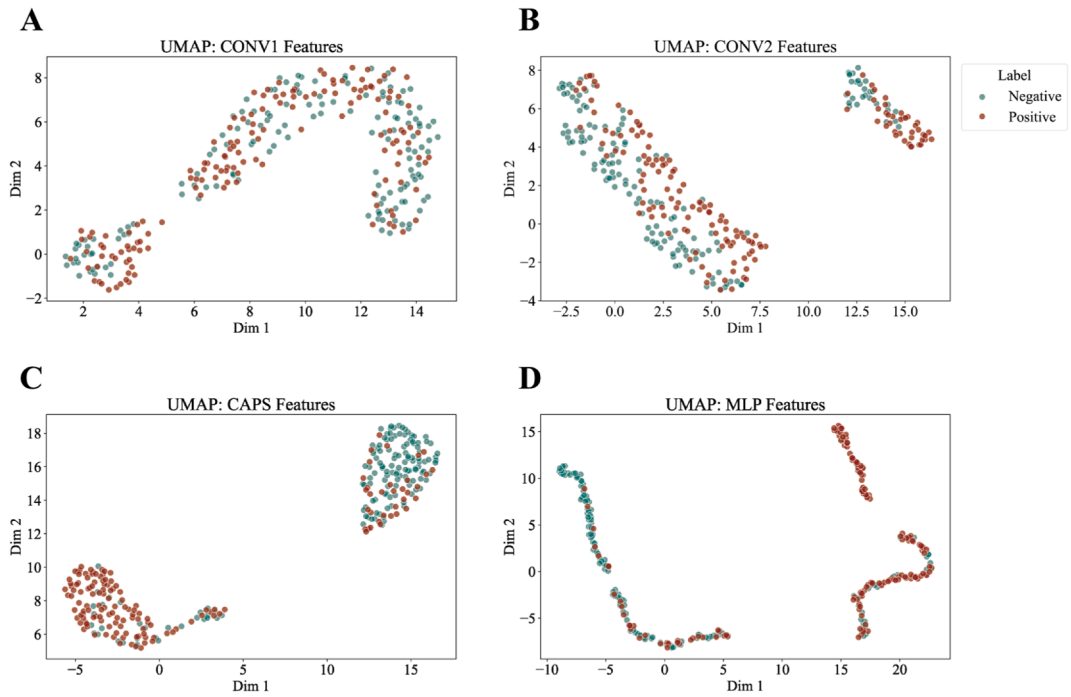

Fig. 5. UMAP-based visualization of feature space across model modules (A) The data output from the first augmented Conv after UMAP dimensionality reduction. (B) The data output from the second augmented Conv after UMAP dimensionality reduction. (C) The data output from the CapsNet after UMAP dimensionality reduction. (D) The data output from the MLP after UMAP dimensionality.

spatial hierarchical features for predicting DBPs.

3.4. Comparison with previous methods

The comparison of methods in predicting DBPs is comprehensively displayed in Table 2 and Fig. 6. The methods evaluated include DPP-PseAAC (Rahman et al., 2018), iDNA-Prot|dis (Liu et al., 2014), iDNA-Prot (Lin et al., 2011), PseDNA-Pro (Liu et al., 2015), IKP-DBPPred (Qu et al., 2017), PSFM-DBT (Zhang & Liu, 2017), CoSEF-DBP (H. Zhang et al., 2025), EmbedCaps-DBP (Naim et al., 2023), TargetDBP (Hu et al., 2019), Deep-WET (Mahmud et al., 2024), and the proposed method. In heatmap Fig. 6, the darker the blue shade, the higher the model's metrics. On the flip side, the lighter the blue shade, the lower the model's metrics. In our method, apart from Spe and Pre, the other indicators are in the darkest state, which means our method performs best across these metrics. The detailed indicators are shown in Table 2. Specifically, CoSEF-DBP achieved the highest scores in Spe and Pre, demonstrating its excellent ability to identify non-DBPs. However, the low ACC and Sen indicate that the model has a very limited ability to accurately identify DBPs, suggesting that it does not possess substantial bioinformatics significance. Comprehensive evaluation shows that, among previous methods, Deep-WET demonstrates the best performance, achieving the highest values in ACC, F1, and MCC. Compared with Deep-WET, our method achieves the highest values across all metrics: an increase of 0.016 points in accuracy (0.7968), 0.0263 points in sensitivity (0.8068), 0.002 points of specificity (0.7833), 0.029 points of MCC (0.588), 0.0165 of Pre (0.8370), and F1 (0.820) improves by 0.020. In particular, on Sen and MCC, our model surpasses Deep-WET by >0.025 in each metric, indicating that our model achieves higher performance in accurately identifying DBPs and overall predictive performance. Undoubtedly, these findings underscore the exceptional capability of our approach in accurately predicting DBPs, identifying true positives, and reducing false positive rates. This enhancement across all metrics effectively demonstrates the stability and dependability of the proposed DBP prediction method, signifying a notable advancement in the field.

Table 2
Performance comparison with previous methods.

Predictor	ACC	Sen	Spe	Pre	F1	MCC
IKP-DBPPred	0.5811	0.527	0.6351	0.5909	0.557	0.163
DPP-PseAAC	0.6115	0.5541	0.6689	0.6260	0.588	0.225
iDNA-Prot	0.6216	0.6351	0.6081	0.6184	0.627	0.243
iDNA-Prot|dis	0.6824	0.723	0.6419	0.6688	0.695	0.366
PseDNA-Pro	0.6723	0.7838	0.5608	0.6409	0.705	0.354
PSFM-DBT	0.6858	0.7162	0.6554	0.6752	0.695	0.372
CoSEF-DBP	0.7453	0.6919	**0.8185**	**0.8530**	0.751	0.521
EmbedCaps-DBP	0.7609	0.7500	0.7759	0.8225	0.783	0.522
TargetDBP	0.7669	0.7635	0.7703	0.7684	0.766	0.534
Deep-WET	0.7808	0.7805	0.7813	0.8205	0.800	0.559
CNNCaps-DBP	**0.7968**	**0.8068**	0.7833	0.8370	**0.820**	**0.588**

To more rigorously verify the superiority of our model, we conducted paired *t*-tests with other models on the independent test dataset. Two reproducible SOTA models were selected for comparison, and their corresponding t-values and p-values were calculated against our model, as shown in Table S7. Specifically, when compared with EmbedCaps-DBP, the p-value was 0.0714, which is slightly higher than 0.05 but shows a trend toward significance, indicating that our model tends to outperform the baseline on the independent test set. In contrast, when compared with CoSEF-DBP, the p-value was 0.0046, significantly lower than 0.05, demonstrating a highly significant difference between the two models. This result confirms that the performance improvement achieved by our model is statistically reliable and not attributable to random fluctuations.

To demonstrate the practical feasibility and reproducibility of our model, we compare and analyze the training time, inference time, and parameter count of CNNCaps-DBP with those of other models which similarly use pre-trained models to encode proteins. We use the same computational resources when running these models. As shown in Table S8, compared to EmbedCaps-DBP and CoSEF-DBP, although our model has more parameters, its training time is shorter and the inference time is only slightly different. Notably, compared to CoSEF-DBP, our training time is only half of theirs. This indicates that our model consumes a certain amount of GPU memory during training; however, the

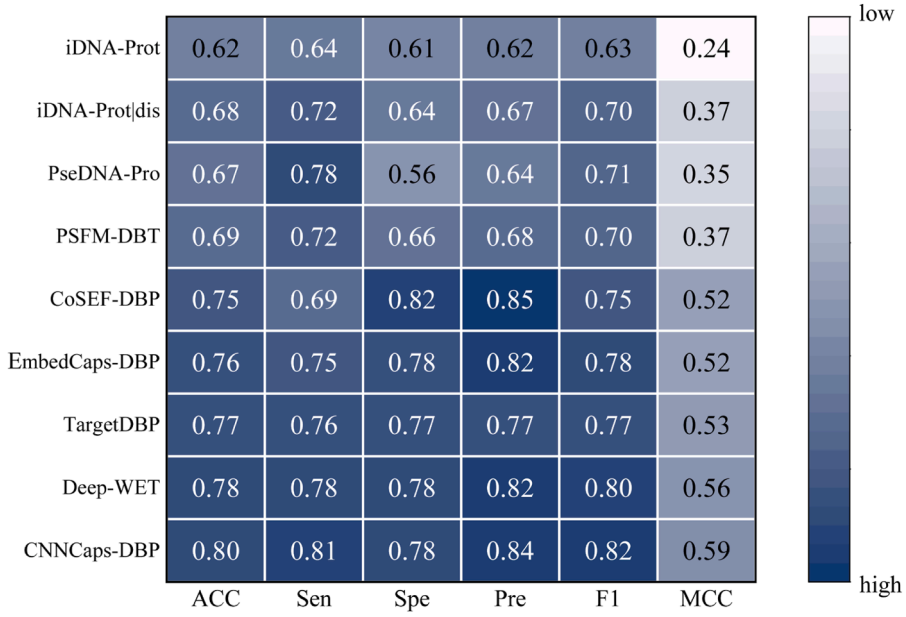

Fig. 6. Heat map of the comparison between CNNCaps-DBP and previous models, where darker blue represents higher metrics and lighter blue indicates lower metrics. CNNCaps-DBP exhibits the deepest blue in ACC, Sen, F1 and MCC, showcasing its outstanding performance.

training time is short, allowing results to be obtained quickly with brief memory usage. These findings clearly demonstrate the practical feasibility and reproducibility of the proposed model in real-world applications.

3.5. Case study: revealing the model's focus on key information through attention weights

We visualized the attention mechanism in our augmented convolutional model to demonstrate its ability to identify DBPs by exploiting the biological information encoded in protein sequences. Specifically, we firstly download the 3D structure of the DBP complexes from the PDB bank, then we import that 3D structure into PyMOL to show the binding sites between the protein and DNA. After that, we retrieved the attention weights from the second augmented Conv module. These attention weights are associated with each amino acid in the protein sequence, reflecting the significance of each amino acid during model training. We plotted the attention weights of the amino acids surrounding the DNA binding sites for the C chain of 4R28 (Horton et al., 2014), the A chain of 4WZS (Butryn et al., 2015) as shown in Fig. 7, and the A chain of 4WZW (Qiu et al., 2014) in Fig. S1. In the Fig. 7, A and B show the 3D structure diagrams and attention heatmaps in which the chroma of the colors indicates the attention values of individual amino acids of complexes 4R28 and 4WZS. Fig. S1 shows the 3D structure diagrams and attention heatmaps in which the chroma of the colors demonstrates the attention values of individual amino acids of complexes 4WZW. It is worth noting that the amino acid sites with high color intensity on the attention heatmap match the DNA binding sites shown in the 3D structure, such as position of 88 of C chain in 4R28, position of 16 and 17 of A chain in 4WZS, and position of 362 of A chain in 4WZW. These observations attest to the model's ability to leverage biological features like DNA binding sites for effective DBP identification, providing indirect support for our hypothesis.

3.6. Generalization experiment

We trained our model on PDB1075 and evaluated its generalization performance on the independent test set PDB186. The models that are compared with ours include: PseDNA-Pro (Liu et al., 2015), DP-Binder (Ali et al., 2019), HMMBinder (Zaman et al., 2017), Local-DPP (Kumar and Raghava, 2007), BiCaps-DBP (Mursalim et al., 2023) DBP-CapsNet (Mursalim et al., 2025). Specifically, since Mursalim et al. (Mursalim et al., 2025) didn't name their model, we call their model DBP-CapsNet to make it easier to describe.

As shown in Table 3, our method reached top accuracy (0.863), sensitivity (0.959) and an impressive Matthew's Correlation Coefficient (0.64). Compared to DBP-CapsNet, having the second-highest accuracy (0.833) and sensitivity (0.930), our method provides a 0.03point improvement in accuracy and a 0.029point improvement in sensitivity, showing its robustness in correctly predicting proteins and high accuracy in identifying DBPs. And compared to DP-Binder and Local-DPP having the second-highest Matthew Correlation Coefficient (0.62), our method achieves the 0.02 improvement in Matthew Correlation Coefficient, demonstrates the stability and reliability of our model. Although less specificity than HMMBinder or DP-Binder, this discrepancy reflects a trade-off between specificity and sensitivity, aimed at prioritizing true positive identification. The performance metrics highlight the suitability of our method for tasks where high-confidence positive predictions are critical.

Similarly, we applied the same paired t-test on the generalization test dataset (PDB186) to evaluate the stability of our model. As shown in Table S7, on the PDB186 dataset, the p-values when compared with BiCaps-DBP ($p = 0.0298$) and DBP-CapsNet ($p = 0.0446$) are both below 0.05, indicating that the performance differences between our model and these baselines are statistically significant. This result undoubtedly provides strong evidence that the superior performance demonstrated by our model in the generalization experiments is not the result of random fluctuations.

4. Conclusions

Characterizing DBPs is crucial for understanding fundamental protein-DNA binding dynamics and deciphering their associated functional consequences. In this research, we developed a whole innovative deep-learning predictor named CNNCaps-DBP, aiming to enhance DBPs prediction accuracy. In CNNCaps-DBP, we use the ESM C pre-trained model to extract features from protein sequences, and then input those features into a deep learning predictor composed of two layers of attention augmented convolution and a capsule network, ultimately obtaining the prediction results. The analysis of the model on the

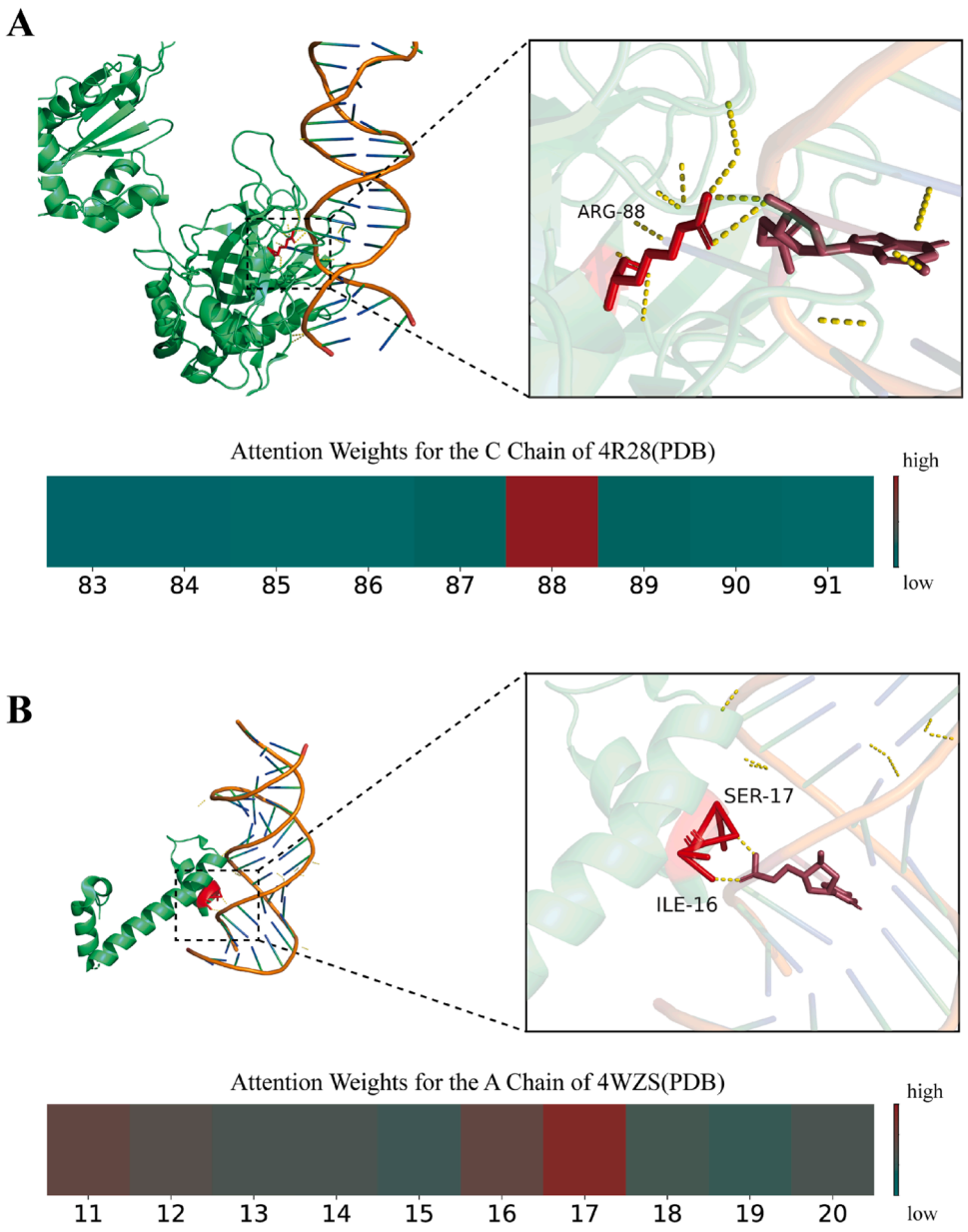

Fig. 7. The attention mechanism reveals the model's proficiency in pinpointing to identify DNA binding sites. (A) 3D structural diagram and the attention mechanism heatmap of the C chain of complexes 4R28, where the attention mechanism weight at position 88 is significantly higher than that of nearby positions. (B) 3D structural diagram and the attention mechanism heatmap of the A chain of complexes 4WZS, where the attention mechanism weight at position 16 and 17 are higher than that of nearby positions.

Table 3
Performance comparison with previous methods in DBP186.

Predictor	ACC	Sen	Spe	MCC
PseDNA-Pro	0.720	0.795	0.645	0.45
DP-Binder	0.812	0.821	**0.802**	0.62
HMMBinder	0.690	0.615	0.763	0.39
BiCaps-DBP	0.735	0.848	0.577	0.44
Local-DPP	0.790	0.925	0.656	0.62
DBP-CapsNet	0.833	0.930	0.611	0.58
CNNCaps-DBP	**0.863**	**0.959**	0.606	**0.64**

independent test dataset indicates that, compared to previous studies, CNNCaps-DBP has achieved breakthroughs in all performance metrics, which undoubtedly highlights the accuracy, effectiveness, and stability of CNNCaps-DBP predictions. The improvement in CNNCaps-DBP performance is mainly attributed to ESM C, augmented Conv and CapsNet. ESM C extracts various features of protein sequences, while the augmented Conv captures local features of the sequences and also focuses on global features through the attention mechanism. The CapsNet preserves the spatial hierarchy of the sequence to the maximum extent when reducing dimensions. Specifically, compared to previous research, ESM C addresses the issue of insufficient comprehensiveness in protein sequence features and the complexity of the extraction process; enhanced convolution resolves the problem of traditional deep learning modules not being able to effectively extract useful information; and capsule networks maximize the retention of spatial information and hierarchical relationships within protein features, thereby preventing feature loss. CNNCaps-DBP is expected to become a powerful tool for accurately and extensively identifying potential DBPs from sequence information.

However, this study has several limitations: (1) although ESM C can extract rich sequence features, there are still shortcomings, such as the

features of protein secondary and tertiary structures, and functional domains. (2) the capsule network exhibits high performance, but have not been directly validated through experiments such as visualization. (3) lacking wet experiments, the biological validity and interpretability of the model predictions need to be improved.

In response to the aforementioned limitations, our future work will focus on these: (1) developing a deep learning model based on graph convolution to capture more comprehensive sequence features of proteins and identify DBPs with unknown structures. (2) trying to explore how capsule network improves the predictive capability of the mode by retaining the hierarchical structure of spatial sequences through visual experiments. (3) Increasing wet testing in experiments to enhance the direct biological applicability of research results.

Data availability

The data and code employed in this study are openly accessible at https://github.com/YZYAlex/CNNCaps-DBP.

CRediT authorship contribution statement

Ziyuan Yan: Writing – review & editing, Writing – original draft, Visualization, Investigation. **Aoyun Geng:** Writing – review & editing. **Yazi Li:** Writing – review & editing. **Jiajing Wang:** Writing – review & editing. **Junlin Xu:** Writing – review & editing. **Yajie Meng:** Writing – review & editing. **Leyi Wei:** Writing – review & editing. **Quan Zou:** Writing – review & editing. **Zilong Zhang:** Writing – review & editing. **Feifei Cui:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

The work is supported by the National Natural Science Foundation of China (No. 62450002), the Hainan Provincial Natural Science Foundation of China (324MS009) and the Science and Technology special fund of Hainan Province (ZDYF2024GXJS018).

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.neunet.2025.108261.

References

Abdelkader, G. A., & Kim, J.-D. (2024). Advances in protein-ligand binding affinity prediction via deep learning: A comprehensive study of datasets, data preprocessing techniques, and model architectures. *Current Drug Targets, 25*(15), 1041–1065.

Akbar, S., Raza, A., Awan, H. H., Zou, Q., & Alghamdi, W. (2025). Saeed A: pNPs-CapsNet: Predicting neuropeptides using protein language models and FastText encoding-based weighted multi-view feature integration with deep capsule neural network. *ACS Omega, 10*(12), 12403–12416.

Alendar, A., & Berns, A. (2021). Sentinels of chromatin: Chromodomain helicase DNA-binding proteins in development and disease. *Genes & Development, 35*(21–22), 1403–1430.

Ali, F., Ahmed, S., Swati, Z. N. K., & Akbar, S. (2019). DP-BINDER: Machine learning model for prediction of DNA-binding proteins by fusing evolutionary and physicochemical information. *Journal of Computer-Aided Molecular Design, 33*(7), 645–658.

Arora, S., Gupta, S., Verma, S., & Malik, I. (2023). Prediction of DNA interacting residues. In *2023 International conference on computational intelligence, communication technology and networking (CICTN)* (pp. 54–57). IEEE.

Attali, I., Botchan, M. R., & Berger, J. M. (2021). Structural mechanisms for replicating DNA in eukaryotes. *Annual Review of Biochemistry, 90*(1), 77–106.

Bello, I., Zoph, B., Vaswani, A., Shlens, J., & Le, Q. V. (2019). Attention augmented convolutional networks. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 3286–3295).

Butryn, A., Schuller, J. M., Stoehr, G., Runge-Wollmann, P., Förster, F., Auble, D. T., & Hopfner, K.-P. (2015). Structural basis for recognition and remodeling of the TBP: DNA: NC2 complex by Mot1. *Elife, 4*, Article e07432.

Castello, A., Horos, R., Strein, C., Fischer, B., Eichelbaum, K., Steinmetz, L. M., Krijgsveld, J., & Hentze, M. W. (2013). System-wide identification of RNA-binding proteins by interactome capture. *Nature Protocols, 8*(3), 491–500.

Chen, Y., Wang, Z., Wang, J., Chu, Y., Zhang, Q., Li, Z. A., & Zeng, X. (2025). Self-supervised learning in drug discovery. *Science China Information Sciences, 68*(7), Article 170103.

Cordoves-Delgado, G., & Garcia-Jacas, C. R. (2024). Predicting antimicrobial peptides using ESMFold-predicted structures and ESM-2-based amino acid features with graph deep learning. *Journal of Chemical Information and Modeling, 64*(10), 4310–4321.

Dai, Y., Wang, C., Chiu, L.-Y., Abbasi, K., Tolbert, B. S., Sauvé, G., Yen, Y., & Liu, C.-C. (2018). Application of bioconjugation chemistry on biosensor fabrication for detection of TAR-DNA binding protein 43. *Biosensors and Bioelectronics, 117*, 60–67.

Donâncio H., Barrier A., South L.F., Forbes F.: Dynamic learning rate for deep reinforcement learning: A bandit approach. *arXiv preprint arXiv:241012598* 2024.

Geng, A., Luo, Z., Li, A., Zhang, Z., Zou, Q., Wei, L., & Cui, F. (2025). ACP-CLB: An anticancer peptide prediction model based on multichannel discriminative processing and integration of large pretrained protein language models. *Journal of Chemical Information and Modeling, 65*(5), 2336–2349.

Grinblat, G. L., Uzal, L. C., Larese, M. G., & Granitto, P. M. (2016). Deep learning for plant identification using vein morphological patterns. *Computers and Electronics in Agriculture, 127*, 418–424.

Gupta, N. K., Wilkinson, E. A., Karuppannan, S. K., Bailey, L., Vilan, A., Zhang, Z., Qi, D.-C., Tadich, A., Tuite, E. M., & Pike, A. R. (2021). Role of order in the mechanism of charge transport across single-stranded and double-stranded DNA monolayers in tunnel junctions. *Journal of the American Chemical Society, 143*(48), 20309–20319.

Hekkelman, M. L., Salmoral, D.Á., Perrakis, A., & Joosten, R. P. (2025). DSSP 4: FAIR annotation of protein secondary structure. *Protein Science, 34*(8), Article e70208.

Horton, J. R., Wang, H., Mabuchi, M. Y., Zhang, X., Roberts, R. J., Zheng, Y., Wilson, G. G., & Cheng, X. (2014). Modification-dependent restriction endonuclease, MspJI, flips 5-methylcytosine out of the DNA helix. *Nucleic Acids Research, 42*(19), 12092–12101.

Hu, J., Zhou, X.-G., Zhu, Y.-H., Yu, D.-J., & Zhang, G.-J. (2019). TargetDBP: Accurate DNA-binding protein prediction via sequence-based multi-view feature learning. *IEEE/ACM transactions on Computational Biology and Bioinformatics, 17*(4), 1419–1429.

Huang, Z., Chen, S., & Yu, L. (2023). Predicting new drug indications based on double variational autoencoders. *Computers in Biology and Medicine, 164*, Article 107261.

Huang, Z., Guo, X., Qin, J., Gao, L., Ju, F., Zhao, C., & Yu, L. (2024). Accurate RNA velocity estimation based on multibatch network reveals complex lineage in batch scRNA-seq data. *BMC Biology, 22*(1), 290.

Huang, Z., Xiao, Z., Ao, C., Guan, L., & Yu, L. (2025). Computational approaches for predicting drug-disease associations: A comprehensive review. *Frontiers of Computer Science, 19*(5), 1–15.

Jaiswal, A., AbdAlmageed, W., Wu, Y., & Natarajan, P. (2018). Capsulegan: Generative adversarial capsule network. In *Proceedings of the European conference on computer vision (ECCV) workshops*, 0-0.

Jin, J., Yu, Y., Wang, R., Zeng, X., Pang, C., Jiang, Y., Li, Z., Dai, Y., Su, R., & Zou, Q. (2022). iDNA-ABF: Multi-scale deep biological language learning model for the interpretable prediction of DNA methylations. *Genome Biology, 23*(1), 219.

Kumar, M., & Raghava, G. P. S. (2007). Identification of DNA-binding proteins using support vector machines and evolutionary profiles. *BMC Bioinformatics, 8*(1), 463.

Kurata, H., Harun-Or-Roshid, M., Tsukiyama, S., & Maeda, K. (2024). PredIL13: Stacking a variety of machine and deep learning methods with ESM-2 language model for identifying IL13-inducing peptides. *PloS One, 19*(8), Article e0309078.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.

Li, T., Ren, X., Luo, X., Wang, Z., Li, Z., Luo, X., Shen, J., Li, Y., Yuan, D., & Nussinov, R. (2024). A foundation model identifies broad-spectrum antimicrobial peptides against drug-resistant bacterial infection. *Nature Communications, 15*(1), 7538.

Li, Y., Geng, A., Zhou, Z., Cui, F., Xu, J., Meng, Y., Wei, L., Zou, Q., Zhang, Q., & Zhang, Z. (2025). AVP-HNCL: Innovative contrastive learning with a queue-based negative sampling strategy for dual-phase antiviral peptide prediction. *Journal of Chemical Information and Modeling*.

Lin, H. (2024). Artificial intelligence with great potential in medical informatics: A brief review. *Medinformatics, 1*(1), 2–9.

Lin, W.-Z., Fang, J.-A., Xiao, X., & Chou, K.-C. (2011). iDNA-Prot: Identification of DNA binding proteins using random forest with grey model. *PloS One, 6*(9), Article e24756.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., & Shmueli, Y. (2023). Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science (New York, N.Y.), 379*(6637), 1123–1130.

Liu, B., Gao, X., & Zhang, H. (2019). BioSeq-Analysis2. 0: An updated platform for analyzing DNA, RNA and protein sequences at sequence level and residue level based on machine learning approaches. *Nucleic Acids Research, 47*(20). e127-e127.

Liu, B., Xu, J., Fan, S., Xu, R., Zhou, J., & Wang, X. (2015). PseDNA-pro: DNA-binding protein identification by combining Chou's PseAAC and physicochemical distance transformation. *Molecular Informatics, 34*(1), 8–17.

Liu, B., Xu, J., Lan, X., Xu, R., Zhou, J., Wang, X., & Chou, K.-C. (2014). iDNA-Prot dis: Identifying DNA-binding proteins by incorporating amino acid distance-pairs and reduced alphabet profile into the general pseudo amino acid composition. *PloS One, 9*(9), Article e106691.

Liu, J.-M., & Yan, X.-P. (2012). Competitive aptamer bioassay for selective detection of adenosine triphosphate based on metal-paired molecular conformational switch and fluorescent gold nanoclusters. *Biosensors and Bioelectronics, 36*(1), 135–141.

Luo, Z., Geng, A., Wei, L., Zou, Q., Cui, F., & Zhang, Z. (2025). CPL-Diff: A diffusion model for De Novo design of functional peptide sequences with fixed length. *Advanced Science, 12*(20), Article 2412926.

Lv, J., Li, K., Wang, Y., Xu, J., Meng, Y., Cui, F., Wei, L., Zhang, Q., & Zhang, Z. (2025). ACP-EPC: An interpretable deep learning framework for anticancer peptide prediction utilizing pre-trained protein language model and multi-view feature extracting strategy. *Molecular Diversity*, 1–16.

Lye, Y. S., & Chen, Y. R. (2022). TAR DNA-binding protein 43 oligomers in physiology and pathology. *IUBMB life, 74*(8), 794–811.

Mahmud, S. H., Goh, K. O. M., Hosen, M. F., Nandi, D., & Shoombuatong, W. (2024). Deep-WET: A deep learning-based approach for predicting DNA-binding proteins using word embedding techniques with weighted features. *Scientific Reports, 14*(1), 2961.

McInnes L., Healy J., Melville J.: Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:180203426* 2018.

Mursalim M.K., Mengko T.L., Hertadi R., Purwarianti A., Susanty M.J.C.i.B., Medicine: BiCaps-DBP: Predicting DNA-binding proteins from protein sequences using Bi-LSTM and a 1D-capsule network. 2023, 163:107241.

Mursalim, M. K. N., Purnajaya, A. R., & Saragih, R. E. (2025). Classifying DNA-binding proteins using GRU, Bi-LSTM, and CapsNet. *Next Research, 2*(1), Article 100177.

Naim, M. K., Mengko, T. R., Hertadi, R., Purwarianti, A., & Susanty, M. (2023). Embedcaps-dbp: Predicting dna-binding proteins using protein sequence embedding and capsule network. *IEEE Access, 11*, 121256–121268.

Qiu, C., McCann, K. L., Wine, R. N., Baserga, S. J., & Hall, T. M. T. (2014). A divergent Pumilio repeat protein family for pre-rRNA processing and mRNA localization. *Proceedings of the National Academy of Sciences, 111*(52), 18554–18559.

Qu, K., Han, K., Wu, S., Wang, G., & Wei, L. (2017). Identification of DNA-binding proteins using mixed feature representation methods. *Molecules (Basel, Switzerland), 22*(10), 1602.

Rahman, M. S., Shatabda, S., Saha, S., Kaykobad, M., & Rahman, M. S. (2018). DPP-PseAAC: A DNA-binding protein prediction model using Chou's general PseAAC. *Journal of Theoretical Biology, 452*, 22–34.

Ren, B., Robert, F., Wyrick, J. J., Aparicio, O., Jennings, E. G., Simon, I., Zeitlinger, J., Schreiber, J., Hannett, N., & Kanin, E. (2000). Genome-wide location and function of DNA binding proteins. *Science (New York, N.Y.), 290*(5500), 2306–2309.

Sanaboyana, V. R., & Elcock, A. H. (2024). Improving signal and transit peptide predictions using AlphaFold2-predicted protein structures. *Journal of Molecular Biology, 436*(2), Article 168393.

Shao, J., Chen, J., & Liu, B. (2024). ProFun-SOM: Protein function prediction for specific ontology based on multiple sequence alignment reconstruction. *IEEE Transactions on Neural Networks and Learning Systems*.

Sun, A., Li, H., Dong, G., Zhao, Y., & Zhang, D. (2024). Dbpboost: A method of classification of dna-binding proteins based on improved differential evolution algorithm and feature extraction. *Methods (San Diego, Calif.), 223*, 56–64.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Wang, C., Hu, L., Guo, M., Liu, X., & Zou, Q. (2015). imDC: An ensemble learning method for imbalanced classification with miRNA data. *Genetics and Molecular Research, 14*(1), 123–133.

Wang, Y., Zhai, Y., Ding, Y., & Zou, Q. (2024). SBSM-Pro: Support bio-sequence machine for proteins. *Science China Information Sciences, 67*(11), Article 212106.

Wei, L., He, W., Malik, A., Su, R., Cui, L., & Manavalan, B. (2021). Computational prediction and interpretation of cell-specific replication origin sites from multiple eukaryotes by exploiting stacking framework. *Briefings in Bioinformatics, 22*(4), bbaa275.

Wei, R., Zhang, L., Zheng, H., & Xiao, M. (2024). A systematic review of the application of machine learning in CpG Island (CGI) detection and methylation prediction. *Current Bioinformatics, 19*(3), 235–249.

Xie, X., Changchun, W., Fuying, D., Kejun, D., Dan, Y., Jian, H., Hao, L., & Hao, L. (2025). scRiskCell: A single-cell framework for quantifying pancreatic islet risk cells and unravelling their dynamic transcriptional and molecular adaptation in the progression of type 2 diabetes. *iMeta*.

Xu, L., Jiang, S., Wu, J., & Zou, Q. (2021). An in silico approach to identification, categorization and prediction of nucleic acid binding proteins. *Briefings in Bioinformatics, 22*(3), bbaa171.

Yan, C., Zhang, Z., Xu, J., Meng, Y., Yan, S., Wei, L., Zou, Q., Zhang, Q., & Cui, F. (2025). CasPro-ESM2: Accurate identification of Cas proteins integrating pre-trained protein language model and multi-scale convolutional neural network. *International Journal of Biological Macromolecules, 308*, Article 142309.

Yao, L., Xie, P., Guan, J., Chung, C.-R., Zhang, W., Deng, J., Huang, Y., Chiang, Y.-C., & Lee, T.-Y. (2024). ACP-CapsPred: An explainable computational framework for identification and functional prediction of anticancer peptides based on capsule network. *Briefings in Bioinformatics, 25*(5).

Zaman, R., Chowdhury, S. Y., Rashid, M. A., Sharma, A., Dehzangi, A., & Shatabda, S. (2017). HMMBinder: DNA-binding protein prediction using HMM profile based features. *BioMed Research International, 2017*(1), Article 4590609.

Zeng, S., Wang, D., Jiang, L., & Xu, D. (2024b). Parameter-efficient fine-tuning on large protein language models improves signal peptide prediction. *Genome Research, 34*(9), 1445–1454.

Zeng, W., Yu, X., Shang, J., Zhao, P., Liu, W., Hu, J., & Peng, S. (2024a). LBi-DBP, an accurate DNA-binding protein prediction method based lightweight interpretable BiLSTM network. *Expert Systems with Applications, 249*, Article 123525.

Zhang, D., Qi, R., Lan, X., & Liu, B. (2025a). A novel multislice framework for precision 3D spatial domain reconstruction and disease pathology analysis. *Genome Research, 35*(8), 1794–1808.

Zhang, H., Yang, X., Chen, P., Yang, C., Chen, B., Jiang, B., & Shan, G. (2025c). CoSEF-DBP: Convolution scope expanding fusion network for identifying DNA-binding proteins through bilingual representations. *Expert Systems with Applications, 263*, Article 125763.

Zhang, H.-Q., Arif, M., Thafar, M. A., Albaradei, S., Cai, P., Zhang, Y., Tang, H., & Lin, H. (2025b). PMPred-AE: A computational model for the detection and interpretation of pathological myopia based on artificial intelligence. *Frontiers in Medicine, 12*, Article 1529335.

Zhang, J., & Liu, B. (2017). PSFM-DBT: Identifying DNA-binding proteins by combing position specific frequency matrix and distance-bigram transformation. *International Journal of Molecular Sciences, 18*(9), 1856.

Zhang, X., Wang, Y., Wei, Q., He, S., Salhi, A., & Yu, B. (2024). DRBPPred-GAT: Accurate prediction of DNA-binding proteins and RNA-binding proteins based on graph multi-head attention network. *Knowledge-Based Systems, 285*, Article 111354.

Zhang, Y., Bao, W., Cao, Y., Cong, H., Chen, B., & Chen, Y. (2022). A survey on protein–DNA-binding sites in computational biology. *Briefings in Functional Genomics, 21*(5), 357–375.

Zhang, Y., Wang, Z., Zeng, Y., Zhou, J., & Zou, Q. (2021). High-resolution transcription factor binding sites prediction improved performance and interpretability by deep learning method. *Briefings in Bioinformatics, 22*(6), bbab273.

Zhou, Z., Xiao, C., Yin, J., She, J., Duan, H., Liu, C., Fu, X., Cui, F., Qi, Q., & Zhang, Z. (2024). PSAC-6mA: 6mA site identifier using self-attention capsule network based on sequence-positioning. *Computers in Biology and Medicine, 171*, Article 108129.