Predicting protein–peptide binding residues via interpretable deep learning

Article *in* Bioinformatics · May 2022

DOI: 10.1093/bioinformatics/btac352

5 authors, including:

Ruheng Wang
Shandong University
2 PUBLICATIONS **0** CITATIONS
SEE PROFILE

Junru Jin
Shandong University
5 PUBLICATIONS **7** CITATIONS
SEE PROFILE

10.1016/j.jtbi.2020.110278

Quan Zou
Xiamen University
102 PUBLICATIONS **4,856** CITATIONS
SEE PROFILE

Leyi Wei
Shandong University
100 PUBLICATIONS **4,792** CITATIONS
SEE PROFILE

Bioinformatics, 2022, 1–10
https://doi.org/10.1093/bioinformatics/btac352
Advance Access Publication Date: 23 May 2022
Original Paper

Sequence analysis

Predicting protein–peptide binding residues via interpretable deep learning

Ruheng Wang[1,2], Junru Jin[1,2], Quan Zou [3], Kenta Nakai [4,*] and Leyi Wei [1,2,*]

[1]School of Software, Shandong University, Jinan 250101, China, [2]Joint SDU-NTU Centre for Artificial Intelligence Research (C-FAIR), Shandong University, Jinan 250101, China, [3]Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 610054, China, and [4]Human Genome Center, Institute of Medical Science, The University of Tokyo, Tokyo 108-8639, Japan

*To whom correspondence should be addressed.

Associate Editor: Lenore Cowen

Received on December 15, 2021; revised on April 13, 2022; editorial decision on May 17, 2022; accepted on May 18, 2022

Abstract

Summary: Identifying the protein–peptide binding residues is fundamentally important to understand the mechanisms of protein functions and explore drug discovery. Although several computational methods have been developed, most of them highly rely on third-party tools or complex data preprocessing for feature design, easily resulting in low computational efficacy and suffering from low predictive performance. To address the limitations, we propose PepBCL, a novel BERT (Bidirectional Encoder Representation from Transformers) -based contrastive learning framework to predict the protein–peptide binding residues based on protein sequences only. PepBCL is an end-to-end predictive model that is independent of feature engineering. Specifically, we introduce a well pre-trained protein language model that can automatically extract and learn high-latent representations of protein sequences relevant for protein structures and functions. Further, we design a novel contrastive learning module to optimize the feature representations of binding residues underlying the imbalanced dataset. We demonstrate that our proposed method significantly outperforms the state-of-the-art methods under benchmarking comparison, and achieves more robust performance. Moreover, we found that we further improve the performance via the integration of traditional features and our learnt features. Interestingly, the interpretable analysis of our model highlights the flexibility and adaptability of deep learning-based protein language model to capture both conserved and non-conserved sequential characteristics of peptide-binding residues. Finally, to facilitate the use of our method, we establish an online predictive platform as the implementation of the proposed PepBCL, which is now available at http://server.wei-group.net/PepBCL/.

Availability and implementation: https://github.com/Ruheng-W/PepBCL.

Contact: weileyi@sdu.edu.cn or knakai@ims.u-tokyo.ac.jp

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Protein–peptide interaction is one of the most important protein interactions, playing a vital role in many cellular processes, such as DNA repair and replication, gene expression and metabolism (Pawson and Nash, 2003; Rubinstein and Niv, 2009). Studies have demonstrated that protein interactions are involved in some abnormal cellular behaviors that induce many kinds of diseases (Lee *et al.*, 2015) and about 40% of these interactions are mediated by relatively small peptides (Neduva *et al.*, 2005). Therefore, identifying the protein–peptide binding residues is necessary both in the understanding of protein functions and drug discovery. Recently, the increasing number of protein–peptide complexes caused by advance in structural biology has brought a lot of physicochemical understanding to study them experimentally. However, traditional experimental methods normally need to determine the structures of protein–peptide complexes before identifying the binding residues, and thus are expensive and time-consuming. Moreover, some technical challenges remain, impacting the accurate identification of protein–peptide binding residues, such as small peptide sizes (Vlieghe *et al.*, 2010), weak binding affinity (Dyson and Wright, 2005), peptide flexibility (Bertolazzi *et al.*, 2014), etc. Therefore, it is highly urgent to develop computational methods for the prediction of protein–peptide binding residues.

Computational methods for predicting protein–peptide binding residues generally can be divided into two classes: structure- and sequence-based. The structure-based approaches use the information

of protein–peptide complex structures to make predictions while the sequence-based methods use the information from sequence perspective. There are some well-known structural-based predictive methods, including PepSite (Petsalaki *et al.*, 2009), Peptimap (Lavi *et al.*, 2013), SPRINT-Str (Taherzadeh *et al.*, 2018), PepNN-Struct (Abdin *et al.*, 2020) and so on. For example, Petsalaki *et al.* (2009) developed PepSite, a computational model that firstly finds the possible binding regions through Position Specific Scoring Matrix (PSSM) derived from known protein–peptide complex structures and uses distance constraints to determine the binding residues. Lavi *et al.* (2013) designed a computational method called Peptimap by using small molecule probes to map and cluster the predicted binding residues. Later on, Taherzadeh *et al.* (2018) proposed SPRINT-Str, a Random Forest (RF)-based predictor trained with various structural features like Accessible Surface Area (ASA) (Kabsch and Sander, 1983), Secondary Structure (SS) and Half Sphere Exposure (HSE). More recently, Abdin *et al.* (2020) proposed a graph learning method namely PepNN-Struct, in which they introduce a graph attention module to encode the protein structural context, integrate peptide sequence contextual information by a multi-head attention module, and determine the binding residues with the peptide and protein embeddings.

The sequence-based methods mainly include SPRINT-Seq (Taherzadeh *et al.*, 2016), PepBind (Zhao *et al.*, 2018), Visual (Wardah *et al.*, 2020) and PepNN-Seq (Abdin *et al.*, 2020). For instance, Taherzadeh *et al.* (2016) presented a Support Vector Machine-based method named SPRINT-Seq, which uses sequence-based features including one-hot vector encoded amino acid type information, evolutionary information, structural information and physicochemical properties to predict the residues that bind to peptides. Beyond that, Zhao *et al.* (2018) designed a consensus-based method namely PepBind where the intrinsic disorder is introduced into the feature design for the first time, since studies demonstrate that the protein-peptide binding is closely associated with the intrinsic disorder (Weatheritt and Gibson, 2012). There exist many methods (Sharma *et al.*, 2019, 2021) that can transform non-image data to image for Convolutional Neural Network (CNN). By using the similar methods, Wardah *et al.* (2020) proposed Visual, a CNN-based method that encodes protein sequences into image-like representations for predicting the peptide-binding residues in proteins. Moreover, Abdin *et al.* (2020) proposed PepNN-Seq in the same study as PepNN-Struct, and the difference between these two models is that PepNN-Seq uses a pre-trained contextualized language model for protein sequence embedding, without using any structural information.

Although computational methods have been developed and made some success in the past few years, they have the following problems that limit the application for large-scale high-throughput prediction. First, as most of protein–peptide complex structures are unknown, existing structure-based prediction methods highly rely on the structural information predicted by third-party computational tools, which easily result in low prediction efficiency and bring some noise information that impacts the predictive performance. As for those sequence-based methods, they also face the same problem; for example, the evolutionary information they often use are derived from position specific scoring matrices (PSSM), which are generated by running the sequence alignment between the query proteins and large-scale protein databases. Second, existing methods heavily rely on hand-crafted feature design to build models, which requires high-professional knowledge for researchers; moreover, hand-crafted feature design is lack of adaptability to some extent. Third, existing methods cannot well address the data imbalance problem underlying the protein–peptide binding residue prediction, easily leading to the poor overall performance. Current methods usually construct a balanced dataset by using under-sampling approaches. However, under-sampling the dataset cannot make full use of the majority samples (Gao and Siu, 2020). Moreover, some methods attempt to address the data imbalance problem by giving the minority samples higher weight to make the model pay more attention to them; however, the weight setting might be closely related to specific datasets, and randomly giving the minority class a high weight cannot be considered as a generic strategy to deal with such a problem.

To solve the above problems, we here develop a novel BERT-based contrastive learning framework named PepBCL to predict protein–peptide binding residues. The novelties of our method can be briefly summarized as follows:

- Unlike existing methods using the information predicted by third-party tools, PepBCL is a completely sequence-based prediction method that uses protein sequences only for model training and making predictions, thus accelerating the prediction process and improving computational efficiency.
- By introducing a well pre-trained protein language model, we can extract and learn high-latent feature representations automatically, which are relevant for protein structures and functions. On the other hand, the BERT model can adaptively learn important features with respect to different datasets.
- We propose a novel contrastive learning-based module to address the data imbalance problem. It can adaptively learn more discriminative and high-quality representations of the peptide binding residues while taking full advantage of the majority samples in the imbalanced dataset.
- We provide the interpretability for our model prediction via visualization of the attention scores in binding regions within protein sequences, and our model can capture the important sequential patterns contributing to peptide-binding prediction in proteins.

2 Materials and methods

2.1 Datasets

In this study, we chose two commonly used benchmark datasets to fairly evaluate and compare our proposed method with existing methods. For convenience of discussion, we denoted the two datasets as **Dataset 1** and **Dataset 2**, respectively. It is worth noting that both two datasets were pre-processed by similar procedure to train robust predictive models; for example, the sequence identity was reduced to 30% using 'blastclust' in BLAST package (Altschul *et al.*, 1997) to avoid the bias of performance evaluation. The summary of the datasets is shown in Table 1. The details of the datasets regarding model training and evaluation are described as follows.

Description of Dataset 1. The dataset was originally proposed by the study of the structure-based method SPRINT-Str (Taherzadeh *et al.*, 2018). It contains 1279 protein–peptide complexes with a total of 16 749 binding residues (positive) and 290 943 non-binding residues (negative). From this dataset, they randomly selected 10% complexes as the independent testing set (denoted as TE125), and

Table 1. Summary of datasets

Datasets	Dataset 1		Dataset 2	
	TR1154 (training set)	TE125 (testing set)	TR640 (training set)	TE639 (testing set)
Number of proteins	1154	125	640	639
Number of residues	276 822	30 870	157 362	150 330
Number of binding residues	15 030	1719	8259	8490
Number of non-binding residues	261 792	29 151	149 103	141 840

Note: The TE125 is only trained on TR1154 and the TE639 is only trained on TR640.

the remaining as training set (denoted as TR1154). The TE125 set contains 125 proteins with 1719 binding residues and 29 151 non-binding residues, while the TR1154 set comprises 1154 proteins with 15 030 binding residues and 261 792 non-binding residues.

Description of Dataset 2. The dataset is derived from Zhao *et al.*'s (2018) work, which comprises the 1279 protein–peptide complexes with 16 749 peptide-binding residues and 290 943 non-binding residues. For model training, they randomly chose 640 out of 1279 complexes with 8259 binding residues and 149 103 non-binding residues (denoted as TR640). The remaining complexes with 8490 binding residues and 141 840 non-binding residues were used as the independent testing set, which is denoted as TE639.

Notably, most existing work use the above training sets for model training and independent testing sets for model evaluation and comparison. For fair comparison, we also followed the same procedure to split Dataset 1 and Dataset 2 for model training and testing, respectively.

2.2 Architecture of the proposed method

Figure 1 illustrates the overview of our model architecture. As can be seen, it includes four modules: (A) sequence embedding module, (B) BERT-based encoder module, (C) output module, and (D) contrastive learning module. In the sequence embedding module, the query protein sequence is submitted to our model, in which every amino acid in the sequence is encoded to a pre-trained embedding vector. As a result, the sequence is converted to an embedding matrix. In the BERT-based encoder module, the resulting embedding matrix is further encoded by a deep pre-trained BERT, generating a high dimensional representation vector. By doing so, the problem of long-distance dependence in protein sequences for identifying peptide-binding residues can be well alleviated with the multi-head attention mechanism of BERT. Afterwards, the representation vector is fed into a fully connected neural network (FNN) layer, which is one such that every input neuron is connected to every neuron in the next layer, resulting in denser representations. Next, in the contrastive learning module, we calculate and optimize the contrastive loss between any two training samples in the training set, and thus obtain more discriminative representations of the binding (non-binding) residues from the BERT-based encoder module. Finally, the output module can generate a residue-level peptide-binding probability and determine whether the residue in the input sequence is peptide-binding or not. More details of the four modules are described as follows.

2.3 Sequence embedding module

The raw protein sequence is first translated into a digital sequence according to a defined vocabulary dictionary, in which every amino acid in a sequence can be seen as a word in a sentence and mapped to a digit value. For example, the amino acid *S* (serine) corresponds

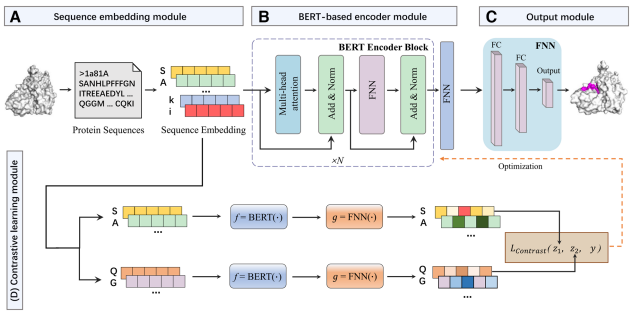

Fig. 1. Flowchart of the proposed PepBCL. Our method consists of four modules. (**A**) The sequence embedding module generates feature embedding matrices from raw protein sequences. (**B**) The BERT-based encoder module can extract the high-quality representations of residues in proteins. (**C**) The feature representations are fed into the output module to predict the residues binding or not. (**D**) Here, we design and implement a contrastive learning module. It can help the BERT-based encoder module obtain more distinguishable representations by minimizing our proposed contrastive loss

to the digit 11, and *L* (leucine) to 6. To be noted, the rare amino acids *U*, *Z*, *O*, *B* will be uniformly replaced by *X* that corresponds to 26 in the dictionary. Here, we do not pad the protein sequences to the same length considering the not big dataset and especially the performance degradation due to excessive paddings. In this way, the raw protein sequence is encoded to a digit-value vector. Then, the encoded vector is embedded by a look-up table, which is an embedding layer pre-trained the same time with the BERT-based encoder module on vast amounts of protein sequences to generate an improved initial embedding.

2.4 BERT-based encoder module

2.4.1 Pre-training of the BERT model

BERT is a bidirectional language representation model based on transformer proposed by Devlin *et al.* (2018). Due to its powerful performance in language understanding against many kinds of large corpus, BERT has been widely used in lots of NLP tasks. To better play the role of BERT, generally it will first be trained on a large background-related corpus with two pre-training tasks namely the masked language model and the next sentence prediction. Here, we use a pre-trained BERT model ProtBert-Big Fantastic Database (BFD) (Elnaggar *et al.*, 2021), which is pre-trained based on the general-domain BERT using a large corpus of unlabeled protein data in BFD (Steinegger *et al.*, 2019), a dataset consisting of 2.1 billion protein sequences. Notably, because there is no direct semantic logic between protein sequences, this domain pre-training stage only uses the masked language model technique compared with the original BERT pre-training.

2.4.2 Encoding process of the module

The basic component of the BERT model is the encoder block which consists of a multi-head attention mechanism, a FNN, and the residual connection technique. The multi-head attention mechanism is composed of many independent self-attention modules to learn multi-view context representation of protein sequences. The self-attention mechanism is described as follows:

$$\begin{cases} Q = XW^Q \\ K = XW^K \\ V = XW^V \end{cases}, \tag{1}$$

$$\text{Self\_Attention}(Q,\ K,\ V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V, \tag{2}$$

where $X \in \mathbb{R}^{L\times d_m}$ is the output of the sequence embedding module and transformed to the query matrix $Q \in \mathbb{R}^{L\times d_k}$, key matrix $K \in \mathbb{R}^{L\times d_k}$, and value matrix $V \in \mathbb{R}^{L\times d_k}$ through the linear layers $W^Q, W^k, W^v \in \mathbb{R}^{d_m\times d_k}$ respectively. L is the length of the input protein sequence, d_m is the initial embedding dimension and the d_k is the dimension of matrix Q, K and V.

The multi-head attention is based on the self-attention module and can be expressed as follows:

$$\begin{cases} \text{Head}_i = \text{Self\_Attention}(XW_i^Q,\ XW_i^K,\ XW_i^V),\ i = 1,\ \ldots,\ h \\ \text{MultiHead\_Attention}(Q,K,V) = [\text{head}_1,\ \text{head}_2,\ \cdots,\ \text{head}_h]\ W^O, \\ X_{\text{MultiHead}} = LN(\text{MultiHead\_Attention}(Q,K,V) + X) \end{cases} \tag{3}$$

where $W_i^Q,\ W_i^K,\ W_i^V \in \mathbb{R}^{d_m\times d_k}$ are the query, key and value linear transformation layers of the ith head while h is the number of head. $W^O \in \mathbb{R}^{hd_k\times d_m}$ is a linear conversion layer that can map the output dimension of the multi-head attention to the initial embedding dimension of the embedding module. Afterward, the residual connection technique and the layer-normalization (LN) are applied, and $X_{\text{MultiHead}}$ is the final output of the multi-head attention module.

The FNN is added to extract a better representation by the activation function, and its mathematical description is as follows:

$$\begin{cases} \mathrm{FNN}(X_{\mathrm{MultiHead}}) = gelu\left(X_{\mathrm{MultiHead}} W^{(1)}\right) W^{(2)} \\ X_{\mathrm{FNN}} = LN(\mathrm{FNN}(X_{\mathrm{MultiHead}}) + X_{\mathrm{MultiHead}}) \end{cases}, \quad (4)$$

where $X_{\mathrm{MultiHead}}$ is the output of the multi-head attention, $W^{(1)} \in \mathbb{R}^{d_m \times d_f}$ and $W^{(2)} \in \mathbb{R}^{d_f \times d_m}$ are two linear layers and shared across all positions. d_m is the initial embedding dimension and d_f is the hidden dimension of the FNN. *gelu* (Gaussian Error Linear Units) is a non-linear activation function. The output of the FNN is also added by a residual and going through a layer normalization.

Since the BERT model has a stack of the encoder blocks, the final encoding process of BERT can be expressed as:

$$X^{(i)} = \mathrm{FNN}\left(\mathrm{MultiHead}\left(X^{(i-1)}\right)\right),\ i = 1, \ldots, n \quad (5)$$

where $X^{(i)}$ is the output of the ith encoder block, while n represents the amount of the encoder block. In particular, $X^{(0)}$ is the initial input embedding matrix, and here for convenience, we view that the multi-head attention and FNN include the residual connection technique and LN.

After the encoding of the BERT model, we will get $X^{(n)}$, the output of the last encoder block, with a still high dimensionality. So, to avoid the redundancy of the dimension, a FNN is used as follows to better extract the representation of the amino acid itself in an input sequence while reducing the dimensionality.

$$X_{\mathrm{Encode}} = elu\left(X^{(n)} W^{(3)}\right) W^{(4)}, \quad (6)$$

where $W^{(3)} \in \mathbb{R}^{d_m \times d_1}$ and $W^{(4)} \in \mathbb{R}^{d_1 \times d_2}$ are linear layers of the FNN, and *elu* (Exponential Linear Units) is a popular non-linear activation function. $d_1,\ d_2$ are the hidden dimensions of the first and second layers of the FNN, respectively. In this way, we get a well-extracted low-dimensional representation of every amino acid in the input sequence.

2.5 Contrastive learning module

Available access to lots of negative samples due to the development of the data augmentation techniques has aroused a wide use of contrastive learning in computer vision, and many contrastive learning frameworks have been proposed such as MoCo (He *et al.*, 2020), SimCLR (Chen *et al.*, 2020) and so on.

These contrastive learning frameworks are mostly based on unsupervised learning and lack of constraints on negative samples because of the characteristics of their tasks. Here, we propose a novel contrastive learning module based on supervised learning to make the representations of the same class inputs mapped to nearby points in the representation space and the different class inputs far. Specifically, given that we do not pad protein sequences to the same length, we will first collect a batch-size of representation matrices from the encoder module. In this way, we can obtain sufficient residue-level data for contrastive learning. Subsequently, to make samples of the same class having similar representations while different class samples having dissimilar representations, we construct a contrastive loss $\mathcal{L}_{\mathrm{contrast}}$ as one of the loss functions of our framework against the imbalanced dataset. For a pair of residue representations $z_1,\ z_2$ in a batch, the loss is defined as follows:

$$\begin{cases} D(z_1,\ z_2) = 1 - cosine\ <\ z_1,\ z_2\ > \\ \mathcal{L}_{\mathrm{contrast}}(z_1,\ z_2,\ y) = \dfrac{1}{2}(1-y)D(z_1,\ z_2)^2 + \dfrac{1}{2}y\{D_{\max} - D(z_1,\ z_2)\}^3 \end{cases} \quad (7)$$

where the distance of the pair of the residue representations $z_1,\ z_2$ can be measured by $D(z_1,\ z_2)$. y equals 1 if this pair of residues belong to different classes that means one residue is binding while the other not and equals 0 if this pair of residues are the same class. $D_{\max}$ is the max value of $D(z_1,\ z_2)$, and equals two here. Notably, we give the different-class pair a higher power that is three to make the model pay more attention to the minority class indirectly.

Algorithm 1: Contrastive loss in one batch

Input: ***X***: the input protein sequences, ***Label***: the labels of the input protein sequences, ***Embed***: Sequence embedding module, ***Encode***: BERT-based encoder module, ***M***: the batch size, ***N***: the number of residues in a batch, ***Z***: the representations of residues in a batch, ***Y***: the labels of residues in a batch;

for $\boldsymbol{i} = 1, \ldots, \boldsymbol{M}$ ***do***
  $\boldsymbol{X_{Embed,i}} = \boldsymbol{Embed}(\boldsymbol{X_i})$
  $\boldsymbol{X_{Encode,i}} = \boldsymbol{Encode}(\boldsymbol{X_{Embed,i}})$
  $\boldsymbol{Z} = \boldsymbol{concate}(\boldsymbol{Z},\ \boldsymbol{X_{Encode,i}})$
  $\boldsymbol{Y} = \boldsymbol{concate}(\boldsymbol{Y},\ \boldsymbol{Label_i})$
end
for $\boldsymbol{j} = 1, \ldots, \boldsymbol{N}//2$ ***do***
  $\boldsymbol{z_j} = \boldsymbol{Z}[\boldsymbol{j}]$
  $\boldsymbol{z_{N//2+j}} = \boldsymbol{Z}[\boldsymbol{N}//2 + \boldsymbol{j}]$
  $\boldsymbol{y} = \boldsymbol{Y}[\boldsymbol{j}]\ \hat{}\ \boldsymbol{Y}[\boldsymbol{N}//2 + \boldsymbol{j}]$
  $\boldsymbol{L_1} = \boldsymbol{L_1} + \mathcal{L}_{\boldsymbol{contrast}}(\boldsymbol{z_j},\ \boldsymbol{z_{N//2+j}},\ \boldsymbol{y})$
end

For computation convenience, we calculate the contrastive loss between the first half and second half of one batch residues data. More details of the contrastive learning module are in Algorithm 1. Then, we can minimize the loss L_1 to obtain a more distinguishable and well-extracted representation of every residue in protein sequences from the BERT-based encoder module.

2.6 Output module

The residue representation vector z, generated by the previous modules from the raw protein sequence x, is fed into a FNN to convert the feature vector to a residue-level category output y_p, that is,

$$\begin{cases} x_{\mathrm{Encode}} = \mathrm{BERT\_based\_Encode}(\mathrm{Sequence\_Embed}(x)) \\ y_p = \mathrm{FNN}(z),\ z\epsilon\{x_{\mathrm{Encode},i} | i = 1, \ldots, n\} \end{cases}, \quad (8)$$

where Sequence_Embed denotes the Sequence embedding module, and BERT_based_Encode denotes the BERT-based encoder module. x_{Encode} is the encoded sequence-level representation composed of many residue feature vectors and $x_{\mathrm{Encode},i}$ is the ith residue representation in the sequence while n is the number of the residues.

The cross-entropy loss function ℓ_{CE} is used here to train the output module to improve the prediction performance, that is,

$$\begin{cases} p_k = \dfrac{\exp(y_{p,k})}{\sum_j \exp(y_{p,j})},\ k = 0, 1 \\ \mathcal{L}_{\mathrm{CE}}(p_1,\ y) = -y \log p_1 - (1-y)\log(1-p_1), \\ L_2 = \sum_{i=1}^{N} \ell_{CE}(p_{1,i},\ y_i) \end{cases} \quad (9)$$

where $k = 0$ *or* 1 denoting the non-binding-residue class or the binding-residue class, and p_k is the probability considering the residue as category k. N is the number of the residues in a batch, y_i is the label of the residue i and L_2 denotes the cross-entropy loss of a batch.

To avoid the back propagation of the loss L_2 disturbing the residue representation learning and the gradient vanishing caused by the deep model BERT, the optimization of the representation learning part and the prediction part are separated. Specifically, we freeze the parameters in the BERT-based encoder module while training the output module. In summary, the loss function of our model is described as:

$$\begin{cases} L_1 = \sum_i \mathcal{L}_{\text{contrast}}\left(Z_i, Z_{N//2+i}, y\right) \\ L_2 = \sum_j L_{CE}\left(Z_j, y_j\right) \\ L = L_1 + L_2 \end{cases} \quad (10)$$

2.7 Evaluation metrics

In this study, the positive and negative samples in the independent datasets are highly imbalanced. To better evaluate the overall performance of our proposed method, we choose to use four metrics commonly used in imbalanced classification tasks, including Recall, Specificity, Precision and Matthews correlation coefficient (MCC), respectively. The calculation formulas are as follows:

$$\begin{cases} \text{Recall} = \dfrac{\text{TP}}{\text{TP}+\text{FN}} \\ \text{Specificity} = \dfrac{\text{TN}}{\text{TN}+\text{FP}} \\ \text{Precision} = \dfrac{\text{TP}}{\text{TP}+\text{FP}} \\ \text{MCC} = \dfrac{\text{TP}\times\text{TN}-\text{FP}\times\text{FN}}{\sqrt{(\text{TP}+\text{FN})(\text{TP}+\text{FP})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}} \end{cases}, \quad (11)$$

where TP (true positive) and TN (true negative) denote the numbers of correctly predicted binding residues and the non-binding residues; FP (false positive) and FN (false negative) denote the numbers of binding residues and non-binding residues that are incorrectly predicted as. Recall measures the proportion of binding residues correctly predicted by the model, and Specificity calculates the proportion of non-binding residues correctly predicted by the model. Precision measures the prediction accuracy of residues that are predicted as binding. MCC is a comprehensive metric considering the prediction ability of both binding residues and non-binding residues and widely used for imbalanced datasets. In addition, AUC that is the area under Receiver Operating Characteristic (ROC) curve (Kumar and Indrayan, 2011) is also calculated to measure the overall performance of our model. Moreover, we used the DeLong test (DeLong *et al.*, 1988) to compute the statistical significance in performance between two different predictive methods.

3 Results

3.1 Comparison with existing methods

To evaluate the predictive ability of our proposed PepBCL, we compared it with eight existing methods, including PepSite, Peptimap, SPRINT-Str, PepNN-Struct, SPRINT-Seq, PepBind, Visual, and PepNN-Seq. It is worth pointing out that four out of the eight methods (i.e. SPRINT-Seq, PepBind, Visual, and PepNN-Seq) are sequence-based predictors while the others are structure-based. We conducted the comparative experiments on two independent testing sets: TE125 and TE639, respectively. The comparative results are shown in Tables 2 and 3, respectively. Note that the source codes for most of the compared methods are not available. As a result, the results for the compared methods on the two datasets are taken directly from their studies.

As seen in Table 2, our proposed PepBCL achieves significantly higher performance than the sequence-based methods (i.e. SPRINT-Seq, PepBind, Visual, and PepNN-Seq) on TE125. Specifically, PepBCL achieves a Precision of 0.540, an AUC of 0.815, and an MCC of 0.385, yielding a relative improvement over the PepBind by 7.1%, 2.2%, and 1.3%. In Figure 2A, we can also see that the ROC curve of our PepBCL is always above the ROC curves of other existing methods on TE125, and achieved the highest AUC. Meanwhile, we can observe that PepBCL also outperforms all the structure-based methods (i.e. Pepsite, Peptimap, SPRINT-Str, and PepNN-Struct) in terms of MCC and all the structure-based methods except PepNN-Struct in terms of AUC. Similar results are also observed in Table 3; that is, our PepBCL performs better than the sequence-based methods such as PepBind and PepNN-Seq by 3.7% and 1.2% in terms of AUC. However, our model performs slightly worse than PepBind in terms of MCC. But seen in Figure 2B, the ROC curve of

Table 2. Comparison of the proposed PepBCL and other methods on TE125 test set

Methods	Recall	Specificity	Precision	AUC	MCC
Pepsite*	0.180	0.970	–	0.610	0.200
Peptimap*	0.320	0.950	–	0.630	0.270
SPRINT-Seq	0.210	0.960	–	0.680	0.200
SPRINT-Str*	0.240	0.980	–	0.780	0.290
PepBind	0.344	–	0.469	0.793	0.372
Visual	0.670	0.680	–	0.730	0.170
PepNN-Seq	–	–	–	0.805	0.278
PepNN-Struct*	–	–	–	**0.841**	0.321
PepBCL (ours)	0.315	0.984	**0.540**	0.815	**0.385**

Note: The metrics of other methods were obtained from the corresponding publications and the methods with * are structure-based. The bold font indicates the best performance for each metric.

Table 3. Performance of the proposed PepBCL and State-of-the-Art methods on TE639 test set

Methods	Recall	Specificity	Precision	AUC	MCC
PepBind	0.317	—	0.450	0.767	**0.348**
PepNN-Seq	—	—	—	0.792	0.251
PepNN-Struct*	—	—	—	**0.838**	0.301
PepBCL(ours)	0.252	0.983	**0.470**	0.804	0.312

Note: The metrics of other methods were obtained from the corresponding publications and the methods with * are structure-based. The bold font indicates the best performance for each metric.

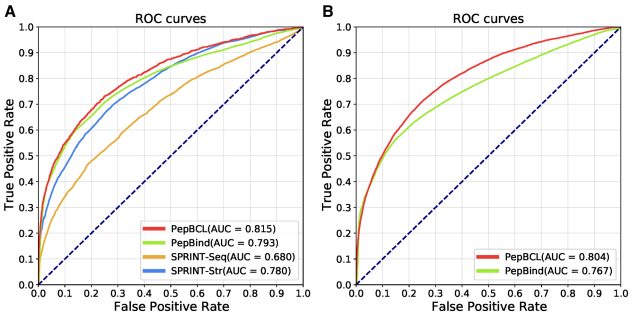

Fig. 2. ROC curves of PepBCL and existing methods. (**A**) ROC curves of the proposed PepBCL and other existing methods on TE125. (**B**) ROC curves of the proposed PepBCL and state-of-the-art method PepBind on TE639

our PepBCL is above the curve of PepBind all the time on TE639, and achieved higher AUC. Compared with the state-of-the-art methods like PepNN-Seq and PepNN-Struct in MCC, our proposed method PepBCL achieved a remarkable improvement of 6.1% as compared to PepNN-Seq and 1.1% higher than PepNN-Struct. The above results demonstrate that our PepBCL achieves better and more robust predictive performance than all existing sequence-based predictors and most structure-based predictors.

Notably, the computational methods such as Pepsite, Peptimap, SPRINT-Seq, SPRINT-Str, PepBind, and Visual are all based on protein properties obtained by many predictive tools to train their models. Compared with these methods, our PepBCL can extract more discriminative features only from the protein primary sequences to predict the peptide binding residues. From Table 2, we can observe that the deep learning approaches (i.e. PepNN-Seq, PepNN-Struct, and our proposed PepBCL) that can automatically learn and extract features perform better than the methods based on tool-based properties, indicating that the models can mine sufficient information by using deep learning techniques to distinguish peptide-binding

residues from non-peptide-binding residues. However, PepNN-struct is a structure-based method and it along with PepNN-Seq uses the peptide sequences in their prediction while ours does not, demonstrating that using the protein primary sequences only, rather than the information from protein structure and peptide is enough to accurately identify the peptide binding residues. In conclusion, the comparative results shown in Tables 2 and 3 demonstrate the overall predictive performance of our proposed model is the best among the compared methods. In addition, it should be pointed out that almost all the existing methods are trained on the benchmark datasets, which are extremely imbalanced, thus leading to the low recall values of the methods. In other words, the methods would miss lots of true protein–peptide binding sites in real application scenarios. Therefore, to solve this problem, we re-trained our model under different probability thresholds and investigated the performance change under different thresholds in terms of Recall, MCC, and Specificity. From Supplementary Figures S2 and S3, we found that the Recall can significantly improve when the threshold is decreased. Similarly, we also observed that the threshold impacts the other metrics. The details and comparison results are shown in Supplementary Tables S1, S2 and Supplementary Figures S2 and S3. Moreover, for the wide use of our proposed method, we provide the threshold setting in our online server, which would provide more options for users according to their need in real application scenarios.

3.2 PepBCL achieves better predictive performance for most protein sequences

Although our model is optimized to improve the overall performance on the datasets, it would be interesting to see if our method can achieve good prediction performance for each protein sequence. Therefore, we further investigated and compared the predictive performance between our proposed PepBCL and existing methods in sequence-level prediction. We compared our sequence-level predictions with the predictions of a typical structure-based method SPRINT-Str on TE125. The results are shown in Figure 3. As seen, our predictions have AUCs > 0.6 in more than 87% of the protein sequences and about 60% of the protein sequences achieve AUC > 0.8. In addition, comparing with SPRINT-Str, our model can predict more protein sequences with higher AUCs in different intervals of AUC metric.

Notably, there are some sequences for which our predictions achieve perfect performance with AUCs significantly higher than SPRINT-Str, including 4buoB, 2vn5C, 2pvcA, and 3mp1A. Additionally, for several protein sequences such as 4m38A, 3hqiA, 2pkuA, and 4gvcA, the AUCs of them nearly lead to 1. Moreover, as noted in the table in Figure 3, there are majority of protein sequences with higher AUC for PepBCL than SPRINT-Str in all range of the AUC metric.

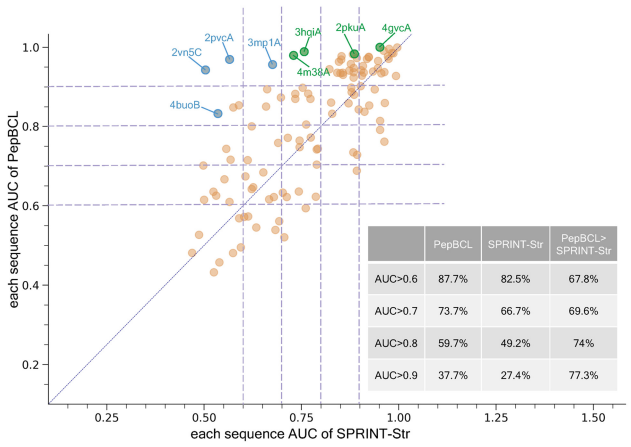

Fig. 3. The AUC score comparison of PepBCL and SPRINT-Str for each protein sequence in TE125. The scatter plot shows the performance comparison between our PepBCL and SPRINT-Str. There are some protein sequences (PDB ID: 4buoB, 2vn5C, 2pvcA, and 3mp1A) that achieve much higher AUCs for PepBCL than SPRINT-Str and several protein sequences (PDB ID: 4m38A, 3hqiA, 2pkuA, and 4gvcA) achieve the perfect performance for PepBCL with AUCs close to 1. The number of protein sequences where the AUCs of PepBCL are better than SPRINT-Str accounts for a large proportion, as noted in the table for all intervals of AUC

3.3 The contrastive learning module learns more discriminative representations in the imbalanced dataset

To examine the effectiveness of the contrastive learning in our model (PepBCL), we compared the PepBCL with the model not trained with contrastive learning module, and evaluated the two models on TE125 and TE639, respectively. The results are shown in Tables 4 and 5. As seen, the Specificity and Precision of the PepBCL are higher than the model without contrast module, indicating that the contrastive learning module can help the model correctly detect more non-binding residues and reduce the number of false positives. Moreover, PepBCL performs better than the model without using the contrastive learning module in terms of AUC and MCC, yielding a relative improvement of 0.9% (P-value $= 0.040$) in AUC and 6.3% in MCC on TE125, and 1% (P-value $<$ 1e-4) in AUC and 2% in MCC on TE639, respectively, demonstrating that the contrastive learning effectively improves the prediction ability of the model. The possible reason for the performance improvement by introducing the contrastive learning is that it can help learn more discriminative representations in the imbalanced datasets. Here, we used t-Distributed Stochastic Neighbor Embedding (t-SNE) to visualize the feature space distribution of the two models as depicted in Figure 4, in which each point represents a residue of a protein sequence and different colors are used to distinguish the positive and negative classes. Figure 4A and B shows the visualization results of the original model and the model without contrastive learning module on TE125, respectively; while Figure 4C and D illustrates the results of the two models on TE639, respectively. As seen, the residue samples of two classes are distributed more clearly in the feature space of the model with contrastive learning module as compared to the model without contrastive learning module, demonstrating that using contrastive learning can capture more discriminative information from different class samples. In summary, we verified that the

Table 4. Performance of the proposed PepBCL with and without the contrastive learning module

Datasets	Methods	Recall	Specificity	Precision	MCC
TE125	PepBCL	0.315	0.984	**0.540**	**0.385**
	PepBCL (no contrast module)	0.485	0.927	0.282	0.322
TE639	PepBCL	0.252	0.983	**0.470**	**0.312**
	PepBCL (no contrast module)	0.397	0.941	0.288	0.292

Note: The bold font indicates the best performance for each metric.

Table 5. Comparison of AUCs of the proposed PepBCL with and without the contrastive learning module

Datasets	Methods	AUC	Difference of AUC	P-value
TE125	PepBCL	**0.815**	0.009	0.040
	PepBCL (no contrast module)	0.806		
TE639	PepBCL	**0.804**	0.010	<1e-4
	PepBCL (no contrast module)	0.794		

Note: The bold font indicates the best performance for each metric.

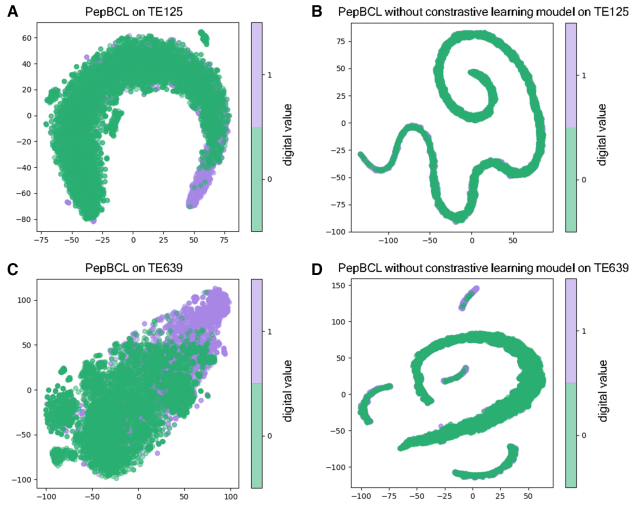

Fig. 4. t-SNE visualization of the feature space distribution of our complete PepBCL and the model lacking of the contrast module. (**A,B**) represent t-SNE visualization results of our PepBCL with and without the contrastive learning module on TE125; (**C,D**) represent t-SNE visualization results of our PepBCL with and without the contrastive learning module on TE639

contrastive learning module is conducive to improving the prediction performance of the model and provide a better solution to the imbalance problem.

3.4 Case study

To intuitively show the prediction performance of our model, we first randomly selected two protein sequences (pdbID: 4l3oA and 1fchA) from TE125, and made predictions using our PepBCL and PepBind. We visualize the actual binding residues from biological experiments and the predicted binding residues of different methods, respectively, in Figure 5. Figure 5A–C illustrates the results of the protein 4l3oA, while Figure 5D–F displays the results of the protein 1fchA. The protein sequences are visualized as 3D structures, in which the predicted binding residues are shown in *magenta* while the non-binding residues are shown in *white*. It can be seen in Figure 5B that the predicted binding residues by our method are more similar to the actual binding residues obtained from experimental methods as compared with PepBind. For example, as shown in Figure 5C, PepBind predicts all residues of the protein 4l3oA as non-binding residues. Similar results can be observed in Figure 5D–F. We can see from Figure 5E and F that our model predicts the binding regions more correctly and determines more complete binding residues in local continuous regions than PepBind. Moreover, the predicted binding residues by our method have a certain degree of similarity as compared with the experimental result. This may be because the features extracted from protein sequences in our model better retain the continuous characteristics of the sequence, and the binding regions of proteins are often continuous and concentrated.

3.5 Discriminative ability of PepBCL over other ligand-binding residues

As mentioned in previous studies, different ligands (peptides, DNA, RNA, and carbohydrate) normally bind to the unique residues on the same protein (Miao and Westhof, 2015; Zhang and Kurgan, 2018). Given this assumption, we explored the effect of our model in detection of other ligand-binding protein residues to see whether the proposed PepBCL is specific for the prediction of peptide-binding residues. For comparison, we randomly collected 30 DNA-, 30 RNA-, and 30 carbohydrate-binding proteins from previous studies (Gattani *et al.*, 2019; Yan *et al.*, 2016), which are denoted as DNA30, RNA30, and CBH30 for convenience of discussion, respectively. We then tested the predictive performance of our PepBCL on the three datasets. The results are illustrated in Figure 6. As shown in Figure 6A, we can observe that the recall of our model

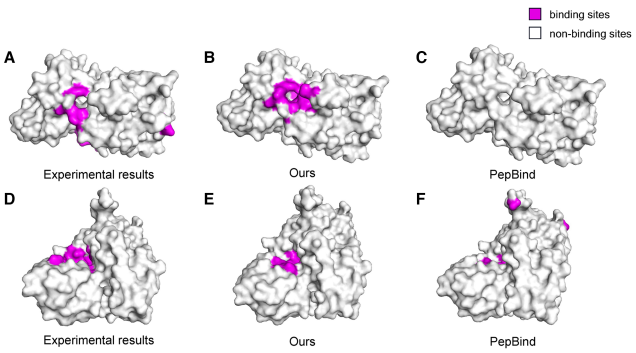

Fig. 5. Visualization of predicted binding residues about two proteins (pdbID: 4l3oA and 1fchA) for our method and existing methods. (**A–C**) The actual binding residues, the predicted binding residues of our PepBCL and the predicted binding residues of PepBind, respectively, about protein 4l3oA; (**D–F**) represent the actual binding residues, the predicted binding residues of our PepBCL and the predicted binding residues of PepBind, respectively about protein 1fchA

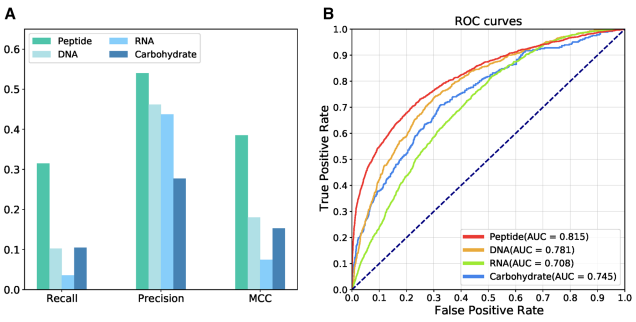

Fig. 6. The performance of our method PepBCL on proteins binding to different ligands (peptides, DNA, RNA, and carbohydrate). (**A**) Recall, Precision, and MCC of our method on different ligand-binding datasets. (**B**) ROC curves of our method on the four different ligand-binding datasets

on the peptide dataset is above 0.3, which is about 3, 8, and 3 times as compared with that on the DNA, RNA, and carbohydrate datasets, respectively. This demonstrates that the proposed PepBCL is more specific to the identification of protein–peptide binding residues. At the same time, we can see that the Precision on the peptide-binding dataset is higher than other three datasets, suggesting the model can well capture the discriminative information for the peptide binding residues. Notably, the Precision of our model on the datasets TE125, DNA30, and RNA30 is much better than on CBH30, which indicates that the predicted binding residues of the peptide, DNA, and RNA have a certain similarity. And these model-predicted DNA and RNA binding residues might be potential peptide binding residues to be verified by further experiments. As shown in Figure 6B, our PepBCL achieves an AUC of 0.815 on TE125, generating a relative improvement over DNA30, RNA30, and CBH30 of 3.4, 10.7, and, 7%, respectively. The highest MCC and AUC on the dataset TE125 as shown in Figure 6A and B also suggests the better performance of PepBCL for the prediction of peptide binding residues. All in all, the above comparison results demonstrate the proposed PepBCL is more discriminative for peptide binding residues prediction compared with other ligand binding proteins.

3.6 BERT-based features outperform the traditional hand-crafted features

In this section, we discuss whether the features learnt from our BERT model are more discriminative than traditional hand-crafted features for the prediction of peptide-binding residues. For comparison, we selected three typical hand-crafted features, which encode sequential, evolutionary, and structural information, respectively. For sequential features, we selected one-hot encoding, in which each residue in sequences is represented by a 20-dimensional one-hot

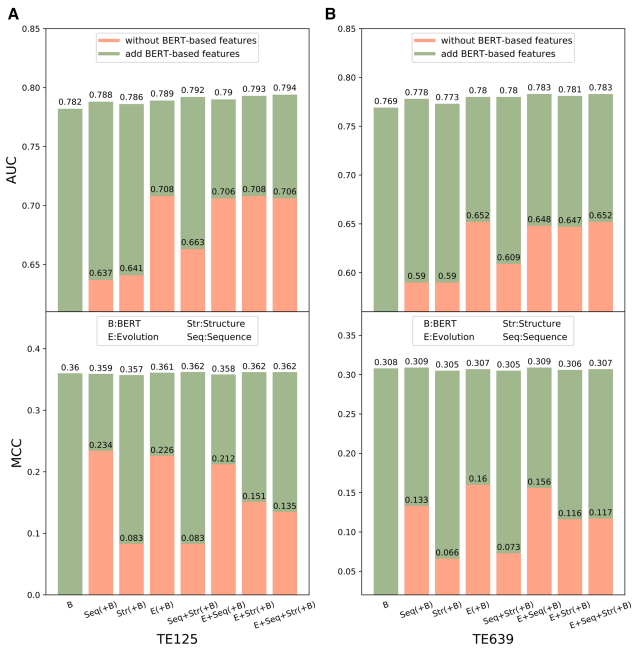

Fig. 7. The performance of different features and their combinations. (**A**) The AUC and MCC values of features were trained on TR1154 and tested on TE125. (**B**) The AUC and MCC values of features were trained on TR640 and tested on TE639

vector with '1' in the position of the corresponding amino acid type and '0' in others. The evolutionary features are derived from the PSSM by running PSI-BLAST. The structural features are generated by SPOT-1D-Single (Singh *et al.*, 2021), which consists of ASA, HSE, Local backbone angles, and SS. Our BERT-based features are the output of the BERT-based encoder module in our PepBCL. As RF-based trained models performed commonly better than other machine-learning algorithms (Manavalan *et al.*, 2019a, b), we used RF here as the underlying classifier for training and predicting the protein–peptide binding residues.

We conducted the comparative experiments on Dataset 1 and Dataset 2, respectively. Specifically, we trained the RF classifiers for all possible combinations of different feature groups on TR1154 and TR640, and investigated the contribution of the three groups of hand-crafted features and BERT-based features on TE125 and TE639, respectively. The results are shown in Figure 7. As seen, the RF classifiers built based on one single feature group obtained the AUC from 0.637 to 0.782 on TE125 and from 0.59 to 0.769 on TE639, and MCC from 0.083 to 0.36 on TE125 and from 0.066 to 0.308 on TE639, in which the BERT-based feature is more powerful than other three hand-crafted features. We can also see that the different combinations of the three hand-crafted feature groups cannot get an obvious improvement of the prediction performance. However, by integrating our BERT-based features with traditional features, the predictive performance obtained a great improvement. In addition, although the RF classifier built upon all the three hand-crafted feature groups got a bad performance compared with other feature combinations, it achieved the highest AUC and MCC by combining the BERT-based feature. The results demonstrate that our designed BERT-based encoder module can extract and learn high-latent representations of protein sequences, complementary to the traditional hand-crafted features for the further performance improvement.

3.7 Interpretable analysis for our model

The previous results have demonstrated that our model has a better performance in predicting the protein–peptide binding residues. Here, to solve the black-box problem in deep learning and reveal the role of the attention mechanism in our model, we visualized the attention scores of binding regions in the protein sequence and further analyzed the sequence patterns derived from our model that are significant for predicting binding residues.

Figure 8 shows the learned attention maps of three binding regions in protein 2wh6A and the sequence patterns obtained by analyzing the binding regions from the dataset. In the attention maps of three binding regions, we can see that the attention scores are larger in the area that the model focuses on, and thus the corresponding colors are darker. About the sequence patterns, specifically, we collected and analyzed the preference of residues around the frequently occurring amino acids in binding residues. Then we visualized the above preference of residues and considered as the sequence patterns in Motif analysis. So, in Figure 8 (see Motif analysis), the amino acid letter with bigger size represents it occurs in this position more frequently. Moreover, we compared the contexts around the residue with largest score with the corresponding sequence patterns in Motif analysis. We can see in region one of Attention maps that there are all six amino acids around the amino acid 'R' occurring in the same position in the corresponding sequence pattern in Motif analysis. In region three, there are four of six amino acids around the 'T' occurring in the same position in the corresponding sequence pattern in Motif analysis. Therefore, the sequence contexts of the residues with largest attention scores in two binding regions (number one and three) identified by attention mechanism are similar to corresponding sequence patterns obtained by analyzing the binding regions of the dataset, indicating the capability of our model to learn the underlying association within contexts of the binding residues. While as for the region two, it is worth mentioning that there are only two amino acids around the 'V' occurring in the same position in the corresponding sequence pattern in Motif analysis. So, we can see that the sequence context around the residue with largest attention score in number two binding region has less similarity with corresponding sequence pattern, demonstrating that our model can find new specific binding sequence patterns different from the sequence patterns obtained by simply analyzing the dataset, due to the attention mechanism.

4 Discussion

In this study, we have developed a BERT-based contrastive learning framework named PepBCL for predicting protein–peptide binding residues using protein sequences only. Benchmarking experiments

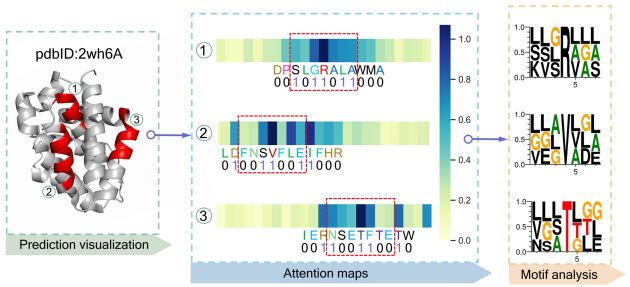

Fig. 8. Visualizations of attention scores for binding regions in PepBCL. Attention maps of three binding regions in protein 2wh6A and the sequence patterns of the residues with largest attention scores in the binding regions obtained by analyzing the dataset

show that the proposed PepBCL outperforms the existing sequence-based methods in terms of all the metrics and structure-based methods in terms of most metrics, further demonstrating that protein sequences themselves contain sufficient information for the prediction of peptide binding residues.

In particular, we use a popular pre-training model BERT as our encoder to automatically learn and generate better representations of the protein sequences avoiding the low efficiency caused by the hand-crafted feature engineering in traditional computational methods. The BERT model is pre-trained on a large corpus of database containing hundreds of millions of protein sequences and BERT-extracted features can capture constraints relevant for structural and functional information of proteins, opening the door in a novel perspective on many downstream tasks about protein sequences. By comparing the BERT-based features with the three groups of hand-crafted features from sequential, evolutionary and structural information, we can observe that evolutionary features obtained from PSSM is more important for predicting peptide-binding residues than other hand-crafted features. A possible reason is that the amino acid type and the structural properties may have a large variability in the protein–peptide interactions, while the evolutionary features are more conserved. Similar results are also observed when combining the BERT-based features; that is, when combined with the BERT-based features, the evolutionary features obtained from PSSM perform better than the other two hand-crafted features. The reason may be the PSSM provides more new information, while the information brought by one-hot encoding and structural features is more included in the BERT-based latent representations. In addition, the incorporation of BERT-based features and hand-crafted features result to a great improvement for the prediction performance; this may be due to the better complementary high-latent features extracted from the BERT-based encoder module in our PepBCL. Furthermore, we conducted comprehensive studies (Supplementary Table S3 and Supplementary Fig. S1) to demonstrate the current model architecture is optimal for our prediction task.

Moreover, the feature representations for different class examples extracted by traditional machine learning methods are less distinguishable, especially when encountering imbalanced datasets. So, we designed and implemented a novel contrastive learning-based module to extract better representations against the imbalanced dataset. By minimizing our constructed contrastive loss function, we can make samples of the same class having more similar representations while different class samples having more dissimilar representations, thus helping the model correctly find more binding residues. The experimental results demonstrate that the contrastive learning module can adaptively extract the discriminative and high-quality features of different class samples and significantly improve the prediction performance.

Many application areas, such as drug discovery and design, heavily require reliable methods to detect the binding residues in protein–ligand interactions. However, there are only about 5.4% of residues involved in the protein–peptide interactions and 94.6% of the residues were not involved. By visualizing the prediction results of two randomly selected peptide-binding proteins (pdbID: 4l3oA and 1fchA), we found that our model can predict the binding regions more correctly and determine more complete binding residues in local continuous regions. This may be because the features extracted from protein sequences in our model better retain the continuous characteristics of the sequence, and the binding regions of proteins are often continuous and concentrated. Overall, the visualization of two proteins (pdbID: 4l3oA and 1fchA) prediction results intuitively illustrates that our model has a better prediction performance than existing methods.

Since different types of ligands usually bind to different residues in proteins, we tested our model on different ligand-binding proteins (peptides, DNA, RNA, and carbohydrate) and the comparison results indicate that our final model trained on the peptide-binding dataset can capture more discriminative information about the protein–peptide interactions and is more specific for the prediction of protein–peptide binding residues as compared with other protein-ligand binding residues.

To conclude, we expect that PepBCL will be a great aid for expanding our knowledge of protein functions and ligand–protein interactions, and for performing numerous sequence-based analyses.

Funding

This work was supported by the National Natural Science Foundation of China [62071278].

Conflict of Interest: All the authors have no conflict of interest.

Data availability

The authors declare that the data supporting the findings of this study are available by accessing the website http://server.wei-group.net/PepBCL/. Besides, the benchmarking datasets and related codes are also available for downloading at https://github.com/Ruheng-W/PepBCL.

References

Abdin,O. *et al.* (2020) Sequence and structure based deep learning models for the identification of peptide binding sites. *Advances in Neural Information Processing Systems*, **33**.

Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Bertolazzi,P. *et al.* (2014) Predicting protein–ligand and protein–peptide interfaces. *Eur. Phys. J.*, **129**, 1–10.

Chen,T. *et al.* (2020) A simple framework for contrastive learning of visual representations. In: *International Conference on Machine Learning*. Vienna, Austria, PMLR. pp. 1597–1607.

DeLong,E.R. *et al.* (1988) Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, **44**, 837–845.

Devlin,J. *et al.* (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:04805.

Dyson,H.J. and Wright,P.E. (2005) Intrinsically unstructured proteins and their functions. *Nat. Rev. Mol. Cell Biol.*, **6**, 197–208.

Elnaggar,A. *et al.* (2021) ProtTrans: towards cracking the language of life's code through self-supervised deep learning and high performance computing. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1–1.

Gao,L. and Siu,S.W. (2020) Study of data imbalanced problem in protein–peptide binding prediction. In: *Proceedings of the 2020 12th International Conference on Bioinformatics and Biomedical Technology, Xi'an, China*. pp. 61–66.

Gattani,S. *et al.* (2019) StackCBPred: a stacking based prediction of protein–-carbohydrate binding sites from sequence. *Carbohydrate Res.*, **486**, 107857.

He,K. *et al.* (2020) Momentum contrast for unsupervised visual representation learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA*. pp. 9729–9738.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolym. Original Res. Biomol.*, **22**, 2577–2637.

Kumar,R. and Indrayan,A. (2011) Receiver operating characteristic (ROC) curve for medical researchers. *Indian Pediatrics*, **48**, 277–287.

Lavi,A. *et al.* (2013) Detection of peptide-binding sites on protein surfaces: the first step toward the modeling and targeting of peptide-mediated interactions. *Proteins Struct. Funct. Bioinf.*, **81**, 2096–2105.

Lee,H. *et al.* (2015) GalaxyPepDock: a protein–peptide docking tool based on interaction similarity and energy optimization. *Nucleic Acids Res.*, **43**, W431–W435.

Manavalan,B. *et al.* (2019a) mAHTPred: a sequence-based Meta-predictor for improving the prediction of anti-hypertensive peptides using effective feature representation. *Bioinformatics*, **35**, 2757–2765.

Manavalan,B. *et al.* (2019b) Meta-4mCpred: a sequence-based meta-predictor for accurate DNA 4mC site prediction using effective feature representation. *Mol. Therapy Nucleic Acids*, **16**, 733–744.

Miao,Z. and Westhof,E. (2015) A large-scale assessment of nucleic acids binding site prediction programs. *PLoS Comput. Biol.*, **11**, e1004639.

Neduva,V. *et al.* (2005) Systematic discovery of new recognition peptides mediating protein interaction networks. *PLoS Biol.*, **3**, e405.

Pawson,T. and Nash,P. (2003) Assembly of cell regulatory systems through protein interaction domains. *Science*, **300**, 445–452.

Petsalaki,E. *et al.* (2009) Accurate prediction of peptide binding sites on protein surfaces. *PLoS Comput. Biol.*, **5**, e1000335.

Rubinstein,M. and Niv,M.Y. (2009) Peptidic modulators of protein-protein interactions: progress and challenges in computational design. *Biopolym. Original Res. Biomol.*, **91**, 505–513.

Sharma,A. *et al.* (2019) DeepInsight: a methodology to transform a non-image data to an image for convolution neural network architecture. *Sci. Rep.*, **9**, 11399.

Sharma,A. *et al.* (2021) DeepFeature: feature selection in nonimage data using convolutional neural network. *Brief. Bioinformatics*, **22**, bbab297.

Singh,J. *et al.* (2021) SPOT-1D-Single: improving the single-sequence-based prediction of protein secondary structure, backbone angles, solvent accessibility and half-sphere exposures using a large training set and ensembled deep learning. *Bioinformatics*, **37**, 3464–3472.

Steinegger,M. *et al.* (2019) Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nat. Methods*, **16**, 603–606.

Taherzadeh,G. *et al.* (2016) Sequence-based prediction of protein–peptide binding sites using support vector machine. *J. Comput. Chem.*, **37**, 1223–1229.

Taherzadeh,G. *et al.* (2018) Structure-based prediction of protein–peptide binding regions using Random Forest. *Bioinformatics*, **34**, 477–484.

Vlieghe,P. *et al.* (2010) Synthetic therapeutic peptides: science and market. *Drug Discov. Today*, **15**, 40–56.

Wardah,W. *et al.* (2020) Predicting protein–peptide binding sites with a deep convolutional neural network. *J. Theor. Biol.*, **496**, 110278.

Weatheritt,R.J. and Gibson,T.J. (2012) Linear motifs: lost in (pre) translation. *Trends Biochem. Sci.*, **37**, 333–341.

Yan,J. *et al.* (2016) A comprehensive comparative review of sequence-based predictors of DNA-and RNA-binding residues. *Brief. Bioinf.*, **17**, 88–105.

Zhang,J. and Kurgan,L. (2018) Review and comparative assessment of sequence-based predictors of protein-binding residues. *Brief. Bioinf.*, **19**, 821–837.

Zhao,Z. *et al.* (2018) Improving sequence-based prediction of protein–peptide binding residues by introducing intrinsic disorder and a consensus method. *J. Chem. Inf. Model.*, **58**, 1459–1468.