Current Drug Targets, 2026, *27*, 132-144

RESEARCH ARTICLE

Improving B-cell Linear Epitope Prediction *via* Multiple Feature Fusion and an Integrated Machine Learning Algorithm

Bing Rao[1,#], Yuxuan Tang[2,#], Jun Hu[2,#,*], Hanin Alahmadi[3], Yaseer Alqahtani[4], Muhammad Arif[5,*] and Tanvir Alam[5,*]

[1]School of Information and Electrical Engineering, Hangzhou City University, Hangzhou, 310015, China; [2]Zhejiang University of Technology, College of Information Engineering, Hangzhou, 310023, China; [3]Department of Computer Science, College of Computer Science and Engineering, Taibah University, Madinah, 344, Saudi Arabia; [4]Independent Research in Health Science, Ministry of Health, Madinah, Saudi Arabia; [5]College of Science and Engineering, Hamad Bin Khalifa University, Doha 34110, Qatar

ARTICLE HISTORY

Received: June 08, 2025
Revised: July 31, 2025
Accepted: August 05, 2025

DOI:
10.2174/0113894501413502251009001940

Abstract: ***Introduction*:** The identification of linear B-Cell epitopes (BCEs) is significantly important for the discovery of drugs, such as antibody production, peptide-based vaccines, and other therapeutics.

***Materials and Methods*:** Unlike traditional laboratory-based methods, computational techniques can save cost and time in predicting large-scale BCEs. For this purpose, numerous *in-silico* methods have been designed to enhance the overall efficacy of BCE prediction. However, research gaps exist for further improvement in the context of using novel feature representations and learning models for BCE prediction. Therefore, in the present study, we aimed to design a novel sequence-based predictor named CoBCEs for screening and discriminating accurate BCEs. The proposed CoBCEs model incorporates the notion of graph-based signature, texture-based, and protein language model (pLM)-based features to sufficiently explore the local and global evolutionary information from protein sequences alone. Then, we fed the fused features, *i.e.*, ProtVec sequence embeddings, Distance-Enhanced Graph (DE-Graph), and term frequency-inverse document frequency (TF-IDF), to an ensemble machine learning classifier.

***Results*:** Experimental results of cross-validation and independent tests on several datasets demonstrate that CoBCEs attained superior performance in terms of accuracy, 77.3%, and Matthews correlation coefficient (MCC) of 61.8%, compared with other existing BCE predictors.

***Discussion*:** Detailed data analyses show that the major advantage of CoBCEs lies in the combined utilization of graph-based and pLM-based features, which extract more discriminative information from sequences. In the future, we aim to develop a publicly available web server using biological language models for large-scale BCE peptide prediction.

***Conclusion*:** We believe our proposed approach will offer valuable insights for drug discovery and disease treatment.

Keywords: B-cell epitopes, protein language model based features, feature selection, machine learning, drug discovery.

1. INTRODUCTION

An antigen is a substance that triggers an immune response and can bind to specific antibodies in the related environment [1], especially in the context of diseases [2]. The sites on an antigen that can be specifically bound by complementary antibodies are called epitopes or antigenic determinants. When the complementary antibody is a BCR, the epitopes are called B-cell epitopes (BCEs). When the complementary antibody is a TCR, the epitopes are called T-cell epitopes (TCEs). Since epitopes help us understand and investigate the mechanisms of cell and tissue activity and assist in diagnosing and treating diseases, the accurate recognition of epitopes is crucial for prevention, diagnosis, and treatment. Particularly, in epitope recognition studies, the identification of BCEs is crucial for peptide-based vaccine design, immunodiagnostic testing, synthetic antibody production, and related applications. Early experiment-

*Address correspondence to these authors at the Zhejiang University of Technology, College of Information Engineering, Hangzhou, 310023, China; E-mail: junh_cs@126.com (J. Hu); College of Science and Engineering, Hamad Bin Khalifa University, Doha 34110, Qatar;
E-mails: mfarif@hbku.edu.qa (M. Arif); talam@hbku.edu.qa (T. Alam)
[#]*These authors contributed equally to this work.*

based methods for BCE identification, such as cryo-EM, mass spectrometry, and X-ray crystallography, have made great progress, but they are costly and time-consuming. Therefore, numerous quick and effective computational techniques are available to predict BCEs, which drive the development of new vaccines and drugs. BCE identification can generally be classified into two major groups: linear (continuous) BCE identification and conformational (discontinuous) BCE identification. Linear BCEs consist of linear extensions of residues in an antigenic protein sequence, while conformational BCEs consist of amino acids that are distantly located in the antigenic primary protein but are brought into proximity through folding in three-dimensional space. While most naturally occurring BCEs are conformational, predictions of linear BCEs have drawn much attention because they are used to synthesize peptide vaccines, among other applications [3, 4]. Improving the accuracy of BCE prediction will directly accelerate epitope vaccine design by shortening candidate screening cycles and significantly reducing time costs in preclinical development.

During the study of linear BCE prediction, the advancement of machine learning techniques has progressively emerged as a popular area of study. For example, BepiPred [5] was developed by combining properties of the Hidden Markov Model (HMM) and amino acid propensity scales for predicting linear epitopes. Similarly, a multi-kernel support vector machine (SVM) classifier was implemented as a learning model for developing the BCPred method [6]. The ABCPred [7] employed a recurrent neural network to predict continuous BCEs. In the meantime, an ensemble-based model called iBCE-EL was proposed using probability features of physicochemical and compositional descriptors [4]. Furthermore, the BepiPred-2.0 predictor [8] for targeting epitopes was proposed based on a random forest classifier. The LBtope [9] method employed diverse features, including binary profile, physicochemical profile, dipeptide composition (DPC) feature descriptor, and AAP (amino acid pair) profile, and used SVM and Weka classifiers to predict BCEs. The EpitopeVec [1] method utilized a blend of residue characteristics, adjusted antigenicity scales, and pLM-based pre-trained models to extract the local and global features from the peptide sequence. It employed Support Vector Machine (SVM) as the model for predicting linear BCEs. The epitope1D method [10] utilized graphical properties of amino acid residues and Organism Ontology information as input features and used an explainable machine learning classifier for BCE prediction. In addition, some methods for predicting linear BCEs based on deep learning have also emerged. For example, the EpiDope method [11] used peptide sequences as inputs to train deep neural networks for the identification of linear B-cell epitopes. Although numerous computational tools have been developed to improve the prediction of linear BCEs directly from raw sequences, there are several limitations in these existing methods, such as limited generalization and insufficient feature fusion, leaving significant potential for enhancing the accuracy and performance of linear BCE prediction.

In the present research, a novel ensemble machine learning predictor called CoBCEs was developed for discriminating and characterizing linear and non-linear BCEs with high accuracy from peptide sequences. In CoBCEs, the feature fusion strategy of residual attributes, pLM-based pre-trained models, *i.e*., ProtVec representation, improved antigenicity scales, multi-text representations, and an improved Graph-based signature representation based on the epitope1D method [10] is used as the features of peptides. To avoid data redundancy and noise problems due to the large number of feature representations, we adopted Random Forest (RF) as a feature selection algorithm to choose the enriched attributes based on their ranking. Meanwhile, we used the AdaBoost (Adaptive Boosting) ensemble algorithm, whose base classifier is RF, as our predictive model, which has better generalization performance than ordinary machine learning algorithms. The empirical prediction outcomes of the cross-validation on several training data sets show that the average accuracy of CoBCEs is 77.3%, and the average MCC value is 61.8%, which outperforms the existing advanced linear BCE prediction tools. To verify the robustness of the proposed CoBCEs method, we compared the independent test results of CoBCEs with other advanced linear BCE prediction methods on several testing data sets, which also demonstrate the superior performance of CoBCEs compared to most other methods. The contribution of this study is summarized as follows:

- **(a)** We developed an ensemble machine learning model called CoBCEs for predicting BCEs from peptide sequences.
- **(b)** We employed embedding feature descriptor (ProtVec), texture-based (TF-IDF), and graph-based signature (DE-Graph) that capture the local and global information from the given peptides.
- **(c)** We further enhanced the proposed model's performance by selecting the optimal feature set using the Random Forest (RF) algorithm.

2. METHODS

2.1. Benchmark Data Sets

The collection or construction of a valid benchmark dataset plays an integral role in developing a computational predictor [12-14]. Most of the existing methods for linear BCE prediction, such as ABCPred, BCPred, AAP, LBtope, iBCE-EL, and EpitopeVec, are trained on datasets compiled from databases of experimentally verified epitopes, such as Bcipep [15] or IEDB [3, 16]. From the above methods, we compile a list of benchmarking datasets with eight datasets: BCPreds, ABCPred, Chen, Blind387, LBTope_fixed_nr, ibce-training, ibce-ind-full, and Viral. CoBCEs performs a fair and comprehensive comparison with existing advanced approaches on these datasets. In order to ensure fair comparison, the same set of training and testing datasets was adopted. The cross-validation is performed on BCPreds, ibce-training, and Viral, and the model obtained by cross-validation on BCPreds is tested on ABCPred, Chen,

Table 1. The benchmarking datasets in this study.

Dataset	Original Method	Epitopes	Non-epitopes
BCPreds	BCPreds	701	701
ABCPred	ABCPred	700	700
Chen	AAP	872	872
Blind387	ABCPred	187	200
LBTope_fixed_nr	LBTope	7824	7853
ibce-training	iBCE-EL	4440	5485
ibce-ind-full	iBCE-EL	1110	1408
Viral	EpitopeVec	4432	8460

Blind387, LBTope_fixed_nr, and ibce-ind-full. The statistical summaries of the benchmarking datasets are listed in Table **1**.

2.2. Feature Encoding

Feature encoding is an important preprocessing step to formulate a biological protein sequence into numerical feature vectors [17-19]. We extracted the prominent features by using compositional descriptors, for example, the composition of AAs, DPC-based encoding method, pLM-based ProtVec features, multi-text representations (TF-IDF), and improved Graph-based protein sequence signature representations. In order to reduce data complexity and noise, the importance values of these features were calculated by the Random Forest algorithm, and then the representative features whose importance values are greater than a set threshold are retained. The above nine features are defined as follows:

AAC considers the normalized occurrence frequency of each AA in a given protein sequence p. The mathematical expression is denoted as equation (1):

$$AAC[P] = [f_1, f_2, f_3, \dots, f_{20}] \quad (1)$$

where $f_i = \frac{P_i}{N}\ [i = 1, 2, \dots, 20]$ is the percentage that is made up by the type i of AAs, P_i is the frequency of occurrence of AA type i in the peptide p and N represents the peptide's p length.

DPC considers the pair occurrence frequency of AAs in a given peptide$p$. It can be expressed as equation (2):

$$DPC(p) = [f_1, f_2, f_3, \dots, f_{400}] \quad (2)$$

where $f_i = \frac{R_i}{N}\ [i = 1, 2, 3, \dots, 400]$ is the percentage of composition of dipeptide type i, R_i is the frequency of occurrence of dipeptide type i in the peptide p and N is the length of the peptide p.

AAP antigenicity scale: The AAP antigenicity scale was initially introduced by Chou *et al*., [20]. It can be defined as the logarithmic ratio of the occurrence frequency of AAPs in a positive dataset compared to a negative dataset. The Bcipep dataset serves as the BCE class, whereas the negative class, as non-BCEs, is constructed from the UniProt50 database in Swiss-Prot [21]. We normalized the values of the extracted features in the scale of +1 and -1, to mitigate the dominance of individual propensity values (equation 3):

$$R_{AAP} = \log[\frac{f^+_{AAP}}{f^-_{AAP}}] \quad (3)$$

B Yao *et al*. developed an antigenic epitope predictor called SVMTrip [22], that incorporates the tri-peptide similarity and propensity score [21]. The proposed feature encoding method considers the specific AA triplet antigenicity score. It calculates the logarithmic ratio of the frequency of AA triplets in BCEs and non-BCEs classes. The equation below (4) suggests that the score $R_{AAT} > 0$ indicates that it is more common in the positive class (BCEs) and vice versa. To ensure balanced representation, the resulting values are normalized between +1 and -1 (equation 4):

$$R_{AAT} = \log[\frac{f^+_{AAT}}{f^-_{AAT}}] \quad (4)$$

ProtVec sequence embeddings: ProtVec sequence embeddings are a protein sequence encoder, inspired by natural language processing methods. It treats protein sequences as "sentences" and k-mers (subsequences of length k) as "words" [23], which has been used in the EpitopeVec method [3]. ProtVec embeddings are trained on large protein sequence databases, such as SwissProt, which lack any metadata, to generate a generic representation of protein k-mers. Detailed steps for constructing ProtVec sequence embeddings are provided in reference [3].

Distance-Enhanced Graph signatures: The Distance-Enhanced Graph (DE-Graph) is a variant of Graph-based signatures proposed by epitope1D [10]. The key idea of Graph-based signatures is to model distance patterns between residues (nodes) at different distance cutoffs (each distance inducing the edges of a graph) according to different labels (15 combinations of five biochemical characteristics of AAs: Aromatic, PolarNeutral, Acidic, Basic, and Apolar), which is inspired by the Cutoff Scanning Matrix (CSM) algorithm. Specifically, the generation steps of DE-Graph signatures are as follows: First, the sequence is divided into different dipeptide combinations according to the distance d as shown in Fig. (**1**), where d = 1, 2, 3, ..., n. Sec-

ondly, the obtained dipeptide combination is labeled according to the AA's biochemical attributes. For example, 'I' is Apolar, 'M' is Apolar, so the label of the 'ZM' dipeptide combination is Apolar - Apolar; 'M' is Apolar and 'N' is PolarNeutral, so the label of the 'MN' dipeptide combination is Apolar - PolarNeutral. Thirdly, for 15 labels, the number of each label is calculated according to different distances, that is, the label value. Finally, the distance information is introduced by adding the distance d to the beginning of each one-dimensional vector, and each sequence is transformed into a two-dimensional matrix of size n × 16. In addition, in DE-Graph signatures, instead of cumulating the label value on different distances, we generate independent label values for 15 labels at different distances.

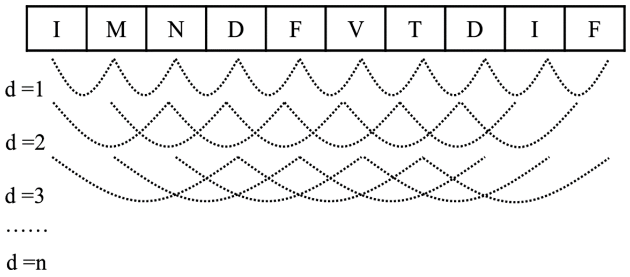

Fig. (1). Dividing the sequence into different dipeptide combinations according to the distance d. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

CTDC: The CTDC is the Composition descriptor in CTD (Composition, Transition, and Distribution) [24, 25] features, which represent the distribution pattern of amino acids in a particular structural or physicochemical property of a protein or peptide sequence. In the CTD feature encoding scheme, numerous physicochemical attributes of AAs are considered. For example, polarity, hydrophobicity, charge, normalized Van der Waals volume, polarizability, solvent accessibility, and secondary structure properties of BCE peptide sequences are employed for generating the feature vector. The CTD feature method works based on the following principle: (i) The given BCE peptide sequences are translated into a feature vector based on the AAs' physicochemical characteristics; (ii) Seven different physicochemical properties of twenty amino acids are divided into three categories: polar, neutral, and hydrophobic, based on the main clusters of the amino acid indices of Tomii and Kanehisa [24]. Considering the hydrophobicity property of the given peptide sample, the composition encoder in the CTD descriptor (CTDC) denotes the overall percentages of neutral, polar, and hydrophobic properties. The CTDC is computed by the following equation (5):

$$C(r) = \frac{N[r]}{N}, r \in \{polar, neutral, hydrophobic\} \quad (5)$$

where $N[r]$ represents the type of amino acid residues r in the extracted sample of length N.

One-hot Encoding: The One-hot encoding is a widely adopted descriptor for formulating the given peptide or protein sample. In this encoding scheme, each AA is denoted as a binary vector of length 20, where only one element is "1" (indicating the presence of a specific amino acid), and all other elements are "0". For example: Alanine (A) is encoded by (10000000000000000000), Cystine (C) is encoded by (01000000000000000000), and Tyrosine (Y) is encoded by (00000000000000000001), respectively. The input sequence is sparsely represented so that the elements in the coding matrix correspond to the positions in the input sequence. For a sequence of length L, each base is converted into a One-hot vector of length 20, for example, the base at the first position is encoded as (1,0,0,...,0), the base at the second position is encoded as (0,1,0,...,0), and the base at the third position is encoded as (0,0,1,...,0), the Lth-position base is encoded as (0,0,0,...,1). Since the input polypeptide sequence length is inconsistent, we set a fixed sequence length of 20, which is the most frequent length. When the length of the sequence is shorter than 20, the encoded sequence is filled with zeros. When the length of the sequence is greater than 20, the encoded sequence is cut from the end. Accordingly, each amino acid is represented by a 43-dimensional binary vector after the encoding, and each peptide sequence is represented by an 860-dimensional binary vector after flattening.

Term frequency-inverse document frequency encoding: The term frequency-inverse document frequency (TF-IDF) encoding consists of two parts: TF and IDF. In the text model, TF-IDF is used to assess the importance of a word in a corpus or document set relative to a document. IDF refers to the frequency of the inverse document, and TF refers to the frequency of the term appearing in the document. The TF and IDF can be calculated by the following equations (6-8):

$$TF[j] = \frac{F_{i\_j}}{N_j} \quad (6)$$

$$DIF[j] = log[\frac{N_{full}}{F_{i\_full}+1} \quad (7)$$

$$TF - DIF = TF * DIF \quad (8)$$

where F_{i_j} Denotes the AA's i frequencies in sequence j. N_j Denotes the sequence j length. N_{full} Represents the total number of AAs in the entire sequence. F_{i_full} Represents the number of sequences in which AA i occurs.

2.3. Ensemble Learning Method

In most of the past methods, RF, SVM, and other ML-based learning models are often used in the task of linear BCE prediction. In the current research, we improved the generalization power of the proposed model by integrating an AdaBoost classifier with RF as the base classifier, which can combine multiple weak classifiers into a strong classifier to improve the overall classification performance and has some resistance to overfitting, to construct the predictive model [26]. The predictive model is evaluated under a 5-fold cross-validation test (CV), and the best classifier is selected for training and prediction [27]. In addition, in the AdaBoost ensemble algorithm, a grid search strategy is employed to select the parameters of the model [28]. The overall workflow of the proposed CoBCE protocol for predicting BCEs and non-BCEs is illustrated in Fig. (**2**).

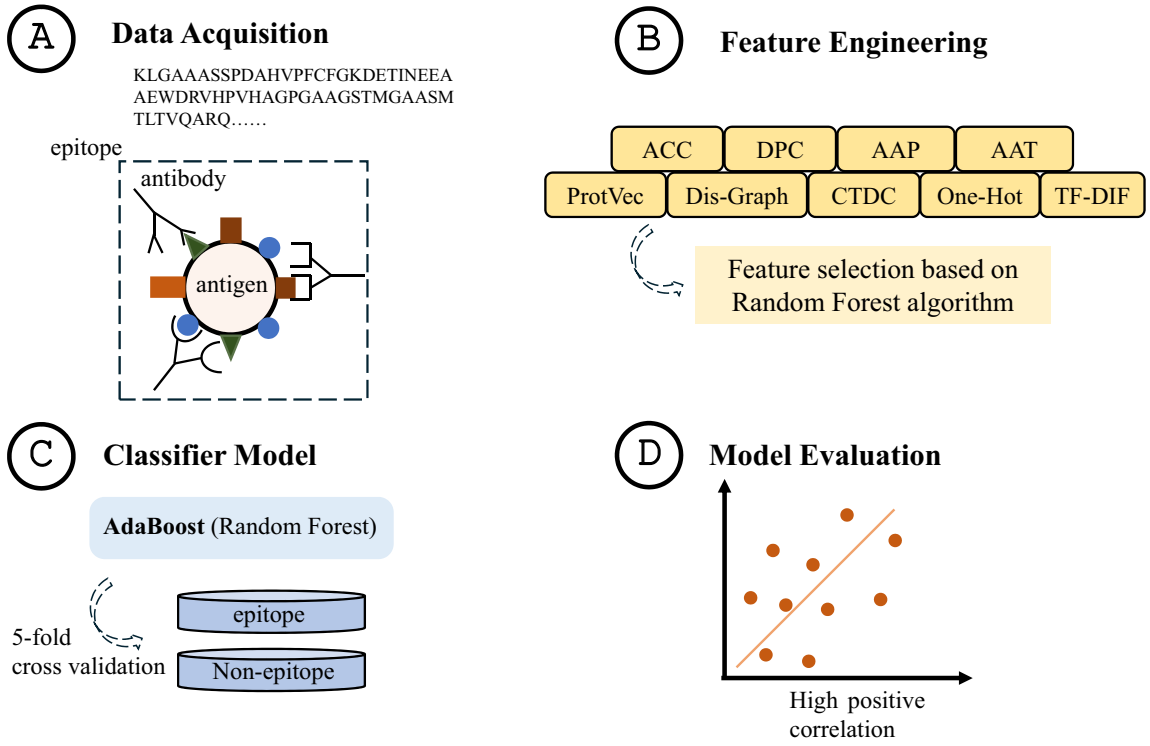

Fig. (2). Overview of the CoBCEs pipeline composed of four main stages. In the first stage (**A**), data acquisition is performed. A list of benchmarking datasets is compiled. In the second stage (**B**), feature engineering is performed. Nine features are used to represent the input data, and the Random Forest algorithm is employed to select features based on the importance values of each feature. In the third stage (**C**), the final encoded sequence is trained and tested using the AdaBoost ensemble algorithm, in which the base classifier is Random Forest. In the fourth stage (**D**), model evaluation is performed. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

2.4. Electrophoretic Gels and Blots

We performed a comprehensive evaluation of the prediction efficacy of our proposed CoBCEs model by several evaluation measures, *i.e.*, accuracy (Acc), Precision (PRE), Recall, F1-score (F1), and Matthews correlation coefficient (MCC) [29]. In this study, we adopt the following equations (9-13):

$$Acc = \frac{TN+TP}{TN+TP+FN+FP} \quad (9)$$

$$PRE = \frac{TP}{TP+FP} \quad (10)$$

$$Recall = \frac{TP}{TP+FN} \quad (11)$$

$$F1 = \frac{2 \cdot TP}{2 \cdot TP+FN+FP} \quad (12)$$

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FN) \cdot (TP+FP) \cdot (TN+FN) \cdot (TN+FP)}} \quad (13)$$

In the aforementioned mathematical expressions [9-13], TN denotes proteins with non-BCE properties and TP represents the proteins having BCE properties. Likewise, FN represents the incorrect protein samples that do not have BCE activity, and FP represents the incorrect protein samples that have BCE properties. The above performance indicators are threshold-dependent. For fair evaluation, we further computed our proposed model's prediction performance using the receiver operating characteristic (ROC) curve, along with the area under the ROC curve (AUC) [30-34].

3. RESULTS

3.1. Determination of Threshold in Feature Selection

In this study, the RF algorithm is used to calculate the importance value of features, and the representative features whose importance values are greater than a set threshold are retained. In this section, the optimal value of the threshold for feature selection for linear BCE prediction will be analyzed. Specifically, we evaluate the MCC variations of CoBCEs obtained by cross-validation on BCPreds, ibce-training, and Viral datasets by gradually changing the value of the threshold from 0.002 to 0.003 with a step size of 0.0001. Table **2** shows that when the threshold is set to 0.0026, the MCC value is highest on the BCPreds dataset; when the threshold is set to 0.002, the MCC value is highest on the ibce-training dataset; when the threshold is set to 0.0022, the MCC value is highest on the Viral dataset. Taking the BCPreds dataset as an example, the MCC value at a threshold of 0.0026 is 0.74306, which is 0.04% higher than that of the second-best threshold (0.74277) and 2.31% higher than that of the worst threshold (0.0029). Although no unified threshold can achieve the highest MCC values on the four datasets, it can be seen that when the threshold is set to 0.0022, the average MCC value is the highest. Since the linear BCE datasets contain a variety of species, in order to increase the generalization of CoBCEs, the threshold with the highest average MCC value is chosen, *i.e.*, 0.0022.

Table 2. The MCC variations of CoBCEs is obtained by cross-validation on BCPreds, ibce-training, and Viral data sets, by gradually varying the value of threshold from 0.002 to 0.003 with the step size of 0.0001.

Threshold	MCC			
	BCPreds	Ibce-training	Viral	Average
0.0020	0.73616	0.51152	0.60049	0.61606
0.0021	0.73875	0.50870	0.59773	0.61506
0.0022	0.73932	0.50591	0.60905	0.61809
0.0023	0.74082	0.50319	0.60398	0.61600
0.0024	0.73597	0.50119	0.60431	0.61382
0.0025	0.74277	0.48985	0.60375	0.61212
0.0026	0.74306	0.49228	0.59451	0.60995
0.0027	0.74054	0.48771	0.59023	0.60616
0.0028	0.74219	0.48870	0.59306	0.60798
0.0029	0.72628	0.47845	0.59295	0.59923
0.003	0.73181	0.47444	0.59852	0.60159

Note: Average means the average MCC values of CoBCEs obtained by cross-validation on BCPreds, Chen, ibce-training, and Viral datasets.

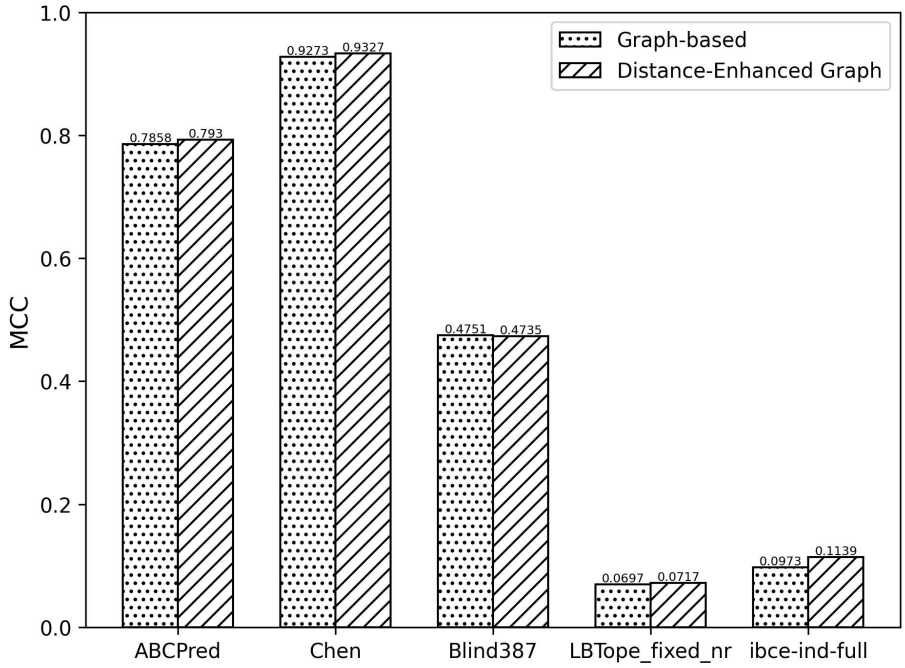

Fig. (3). The performance of the model using Distance-Enhanced Graph signature and the model using Graph-base signature. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

3.2. Comparison of Distance-Enhanced Graph and Gragh-based Signature

To verify that the Distance-Enhanced Graph signature, which introduces distance information into Graph-based signature of the epitope1D method by not cumulating label values, can improve the performance of linear BCE prediction, the performance of the model using the Distance-Enhanced Graph signature and the model using Graph-based signature are compared. Fig. (**3**) shows the comparison results on ABCPred, Chen, Blind387, LBTope_fixed_nr, and ibce-ind-full datasets. It can be observed that the values of MCC for the model using the Distance-Enhanced Graph signature on ABCPred, Chen, LBTope_fixed_nr, and ibce-ind-full are 0.793, 0.9327, 0.0717, and 0.1139, which are 0.92, 0.58, 2.87, and 17.06 percent higher than the values measured for the model using Graph-based signature, respectively, although the model using the Distance-Enhanced Graph signature has a lower MCC value on the Chen dataset. Therefore, in this study, the Distance-Enhanced Graph signature is employed to help improve the predictive performance of linear BCE prediction.

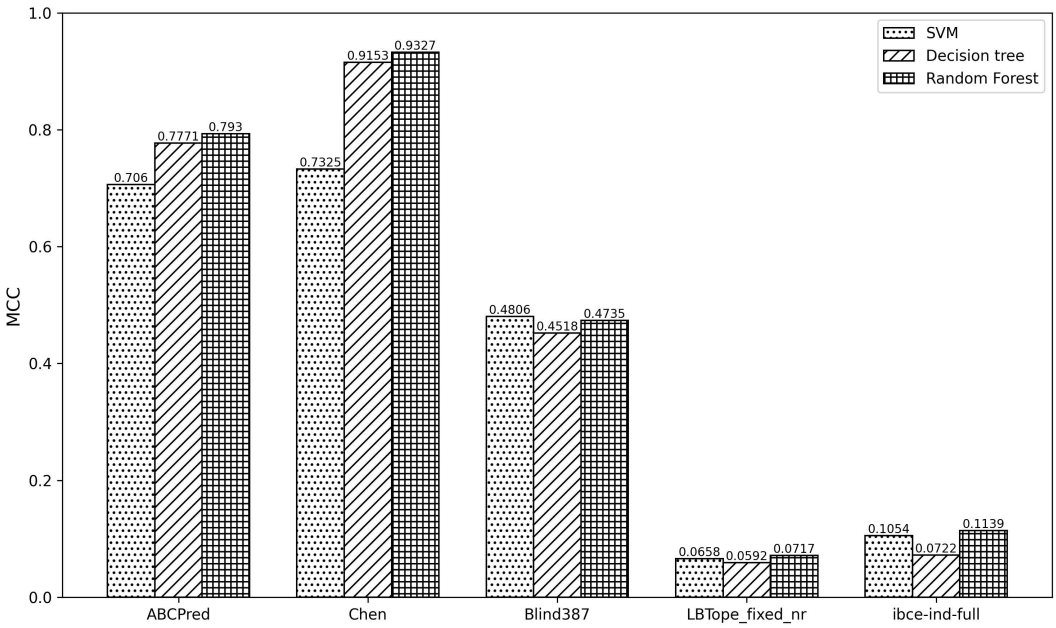

Fig. (4). The performances of AdaBoost ensemble algorithm using Decision tree, SVM, and Random Forest as base classifiers on ABCpred, Chen, Blind387, LBTope_fixed_nr, and ibce-ind-full dataset, respectively. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

3.3. Determination of the Base Classifier in AdaBoost

To verify which base classifier in the AdaBoost ensemble algorithm performs better, the performances of the AdaBoost ensemble algorithm using the three most commonly used machine learning algorithms: Decision Tree, SVM, and Random Forest, as base classifiers are compared. It can be seen from Fig. (**4**) that the AdaBoost ensemble algorithm using SVM as the base classifier on the Blind387 dataset achieves the highest MCC value. However, on the ABCPred, Chen, LBTope_fixed_nr, and ibce-ind-full datasets, the AdaBoost ensemble algorithm using Random Forest as the base classifier achieves the highest MCC values. Therefore, in this study, Random Forest is chosen as the base classifier for the AdaBoost ensemble algorithm.

3.4. CoBCEs Comparison with Existing Methods

In the current subsection, the prediction performance of our proposed CoBCEs model was compared with several advanced BCE-based tools, *i.e.*, BepiPred, ABCPred, BCPreds, iBCE-EL, EpiDope, EpitopeVec, and epitope1D. The results of epitope1D have been obtained from the cited study [10], and the predicted outcomes of other methods are taken from [3]. These predictions are tabulated in Tables **2** and **3**. It is worth mentioning that for the ibce-training and Viral datasets, only the EpitopeVec method and its source method have been cross-validated, so we have not compared all methods for these two datasets.

Table **3** shows the CV performances of CoBCEs and several existing advanced predictors on BCPreds, ibce-training, and Viral datasets. For BCPreds and ibce-training, we can observe that our proposed CoBCEs attains the highest values of Acc, PRE, F1, MCC, and AUC and the second-highest value of Recall compared to other methods. For Viral, compared with EpitopeVec, CoBCEs achieves 3.51, 8.50, 0.29, 9.93, and 2.17 percent improvements in Acc, PRE, Recall, MCC, and AUC, respectively, although CoBCEs has a lower F1.

Table **4** shows the prediction outcomes of existing methods and CoBCEs on blind datasets of ABCPred, Chen, Blind387, LBTope_fixed_nr, and ibce-ind-full datasets of the models obtained by cross-validation on the BCPreds dataset. It can be seen that the highest Acc, PRE, F1, MCC, and AUC values and the third-highest Recall value are obtained by CoBCEs on the ABCPred dataset compared to other methods. For the Chen dataset, the overall performance of CoBCEs is better than that of other methods. For the Blind387 dataset, CoBCEs obtains the second-highest Acc, PRE, MCC, and AUC values compared to other methods. For the LBTope_fixed_nr dataset, the second-highest PRE value and third-highest Acc and MCC values are achieved by CoBCEs. For the ibce-ind-full dataset, CoBCEs obtains the highest MCC value compared with other methods, although it has lower values for the other metrics. In addition, the performance of CoBCEs cross-validated on the ibce-training dataset on the ibce-ind-full dataset is also compared with the EpitopeVec method and its source method. In Table **5**, the comparison results are reported on ibce benchmark datasets (training and testing). From Table **5**, we can observe that the proposed CoBCEs method obtains superior performance in terms of AUC, MCC, Recall, PRE, Acc, and F1 score, although it has a lower F1.

4. DISCUSSION

4.1. Ablation Experiment

In our study, nine features are used to represent protein sequences, and there are multiple combinations of these nine

Table 3. CoBCEs performance comparison with existing methods on BCPreds, ibce-training, and Viral benchmark datasets.

Dataset	Method	Acc	PRE	Recall	F1	MCC	AUC
BCPreds	BepiPred	0.612	0.650	0.490	0.610	0.232	0.665
	ABCPred	-	-	-	-	-	0.643
	BCPreds	0.679	-	0.730	-	0.360	0.758
	iBCE - EL	0.487	0.490	0.970	0.330	-0.009	0.576
	EpiDope	0.507	0.810	0.020	0.350	0.067	0.575
	EpitopeVec	0.813	0.816	0.807	0.811	0.627	0.889
	epitope1D	-	-	-	0.860	0.720	0.930
	CoBCEs	0.866	0.919	0.803	0.856	0.739	0.931
ibce-training	iBCE - EL	0.729	-	0.716	-	0.454	0.782
	EpitopeVec	0.714	0.700	0.640	0.710	0.419	0.789
	CoBCEs	0.756	0.739	0.705	0.722	0.506	0.833
Viral	EpitopeVec	0.797	0.718	0.698	0.850	0.554	0.875
	CoBCEs	0.825	0.779	0.700	0.734	0.609	0.894

Table 4. Performance comparison of CoBCEs with existing predictors on blind datasets of ABCpred, Chen, Blind387, LBTope_fixed_nr, and ibce-ind-full data sets.

Dataset	Method	Acc	PRE	Recall	F1	MCC	AUC
ABCpred	BepiPred	0.577	0.600	0.460	0.57	0.158	0.624
	ABCPred	-	-	-	-	-	-
	BCPreds	0.746	-	0.700	-	0.493	0.801
	iBCE - EL	0.527	0.510	0.960	0.420	0.112	0.588
	EpiDope	0.506	0.760	0.020	0.350	0.059	0.599
	EpitopeVec	0.856	0.840	0.890	0.860	0.714	0.929
	epitope1D	0.823	-	-	0.798	0.667	0.823
	CoBCEs	0.896	0.903	0.889	0.896	0.793	0.948
Chen	BepiPred	0.604	0.640	0.470	0.600	0.217	0.665
	ABCPred	-	-	-	-	-	-
	BCPreds	-	-	-	-	-	-
	iBCE - EL	0.494	0.500	0.960	0.350	-0.036	0.528
	EpiDope	0.507	0.770	0.020	0.350	0.061	0.559
	EpitopeVec	0.883	0.850	0.930	0.880	0.770	0.958
	epitope1D	0.907	-	-	0.909	0.815	0.907
	CoBCEs	0.966	0.940	0.994	0.967	0.933	0.997
Blind387	BepiPred	0.566	0.550	0.530	0.570	0.132	0.627
	ABCPred	0.664	-	0.720	-	-	-
	BCPreds	0.659	-	0.660	-	0.318	0.699
	iBCE - EL	0.434	0.440	0.840	0.320	-0.227	0.501
	EpiDope	0.508	1.000	0.020	0.360	0.091	0.541
	EpitopeVec	0.718	0.750	0.630	0.720	0.445	0.778
	epitope1D	0.781	-	-	0.848	0.543	0.841
	CoBCEs	0.727	0.829	0.545	0.658	0.474	0.789

(Table 4) contd…

Dataset	Method	Acc	PRE	Recall	F1	MCC	AUC
LBTope_ fixed_nr	BepiPred	0.546	0.550	0.490	0.550	0.092	0.566
	ABCPred	-	-	-	-	-	-
	BCPreds	0.526	-	-	-	-	-
	iBCE - EL	0.522	0.510	0.990	0.390	0.135	0.619
	EpiDope	0.503	0.680	0.010	0.350	0.036	0.559
	EpitopeVec	0.530	0.550	0.320	0.510	0.065	0.548
	epitope1D	0.528	-	-	0.333	0.067	0.527
	CoBCEs	0.528	0.576	0.208	0.305	0.072	0.558
ibce-ind-full	BepiPred	0.552	0.500	0.470	0.550	0.104	0.568
	ABCPred	-	-	-	-	-	-
	BCPreds	-	-	-	-	-	-
	iBCE - EL	-	-	-	-	-	-
	EpiDope	0.505	0.700	0.010	0.370	0.049	0.595
	EpitopeVec	0.542	0.520	0.310	0.530	0.095	0.571
	epitope1D	0.594	-	-	0.253	0.092	0.531
	CoBCEs	0.582	0.575	0.197	0.294	0.114	0.561

Table 5. Performance of CoBCEs trained on ibce-training data set on ibce-ind-full data set.

Method	Acc	PRE	Recall	F1	MCC	AUC
iBCE-EL	0.734	-	-	0.730	0.454	0.786
EpitopeVec	0.707	0.068	0.630	0.700	0.402	0.782
CoBCEs	0.741	0.712	0.694	0.703	0.474	0.814

features. In order to verify which combination is most useful for predicting BCEs, an ablation experiment is conducted in this section. Due to the large number of experiments required to perform a complete ablation experiment on the nine features, the following rules are adopted to simplify the experiment: First, the feature importance values of the nine features obtained through the Random Forest algorithm are shown in Fig. (**5**). Then, features whose importance value is greater than 0.05 are selected, namely AAT, DPC, ProtVec, and One-hot. The combination of these four features is named Base. Finally, from the remaining five features, *i.e.*, AAP, AAC, DE-Gragh, TF-IDF, we select the combination of k features that are not repeated and gradually add them to the Base, where k = 1,2,3,4,5, resulting in 32 feature combinations. The performances of CoBCEs based on 32 feature combinations on ABCPred, Chen, LBTope_fixed_nr, and ibce-ind-full datasets are shown in Tables **S1-S5** of the supplementary file.

From Fig. (**5**), we can see that the ProtVec feature achieves the highest importance value among the other eight features. Concretely, the importance values of ProtVec, AAT, DPC, One-hot, DE-Graph, AAC, TF-IDF, CTDC, and AAP are 0.7385, 0.0747, 0.0551, 0.0504, 0.0453, 0.0188, 0.0118, 0.0055, and 0, respectively. The potential reason that ProtVec achieves the highest importance value is that protein language models (pLMs) capture evolutionary and structural information from vast protein sequence data, which can enhance the prediction of BCEs by identifying conserved functional patterns and key residues that might be missed by traditional methods. From Table **S1**, it can be observed that, on the ABCPred dataset, the feature combination of Base and CTDC obtains the highest Acc value, the feature combination of Base, AAC, and DE-Graph achieves the highest PRE and MCC values, and the feature combination of Base, AAP, and AAC achieves the highest Recall, F1, and AUC values. For the Chen dataset, the feature combination of Base, AAC, and CTDC obtains the highest Acc, F1, MCC, and AUC values, and the feature combination of Base, AAP, AAC, CTDC, and TF-IDF achieves the highest Acc, Recall, F1, and AUC values. For the Blind387 dataset, it can be seen that the feature combination of Base, AAC, and CTDC obtains the highest Acc, PRE, and MCC values, the feature combination of Base, AAP, AAC, and DE-Graph achieves the highest Recall and F1 values, and the feature combination of Base and CTDC obtains the highest AUC value. For the LBTope_fixed_nr dataset, the highest Acc, PRE, Recall, F1, and MCC values are obtained by the feature combination of Base, AAC, DE-Graph, and CTDC. For the ibce-ind-full dataset, the highest Recall and F1 values are obtained by the feature combination of Base, AAP, AAC, DE-Graph, and CTDC. From the above results, it can

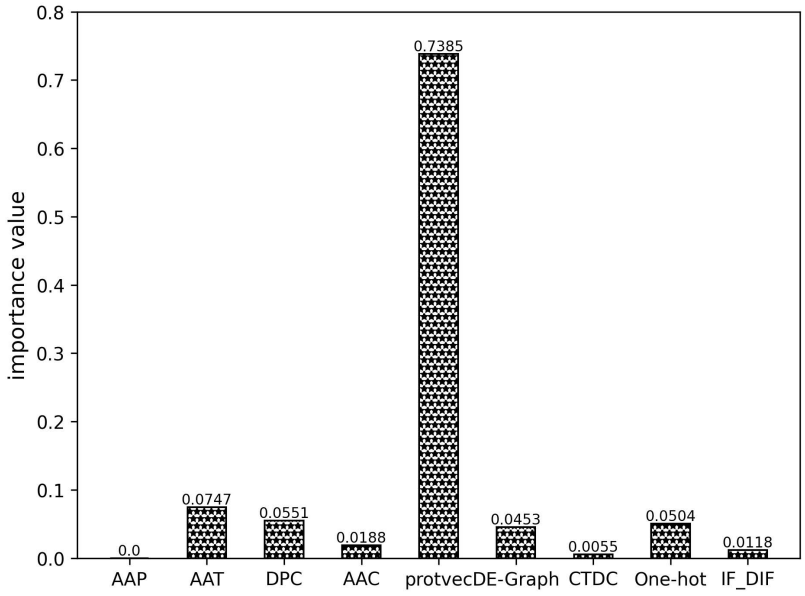

Fig. (5). The importance values of nine features are obtained by Random Forest algorithm. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

be seen that most of the combinations with the highest performance contain AAC, so it can be inferred that AAC plays an indispensable role in the prediction of linear BCEs.

4.2. Impact of Different Species on BCEs Prediction

From Table **5**, **S1-S4**, it is clear that the MCC values tested by all methods on LBTope_fixed_nr and ibce-ind-full datasets are around 10%, while on Chen and ABCPred16 datasets, the MCC values are above 80%. In order to verify whether the species differences between the testing datasets and the training dataset are responsible for the poor predictive performance of these datasets, sequence distribution maps of the BCPreds dataset and four independent testing datasets, ABCPred, Chen, LBTope_fixed_nr, and ibce-ind-full datasets, are generated. From Figs. (**S1-S4**), it can be seen that the average difference values between the positive sequences of ABCPred, Chen, LBTope_fixed_nr, and ibce-ind-full datasets and the positive sequences of the BCPreds dataset are 0.00358, 0.00152, 0.00309, and 0.00309, respectively. Meanwhile, the average difference values between the negative sequences of ABCPred, Chen, LBTope_fixed_nr, and ibce-ind-full datasets and the negative sequences of the BCPreds dataset are 0.00224, 0.00199, 0.00516, 0.00567, respectively. It can be concluded that the sum of the average difference values for the positive and negative sequences of ABCPred, Chen, LBTope_fixed_nr, and ibce-ind-full datasets and the positive and negative sequences of the BCPreds dataset are 0.00582, 0.00351, 0.00825, and 0.00876, respectively, inferring that compared with Chen and ABCPred16 datasets, the LBTope_fixed_nr and ibce-ind-full datasets have more significant differences in species distribution compared to the BCPreds dataset. Therefore, it may lead to a conclusion that the prediction of linear BCEs is highly dependent on the species distribution of the dataset.

In order to improve the generalization of CoBCEs, a mixed dataset, blending three datasets used for cross-validation, BCPreds, iBCE-EL, and Viral, is generated. The test results on ABCPred, Chen, LBTope_fixed_nr, and ibce-ind-full datasets of the model cross-validated on the mixed dataset are compared with those of the model cross-validated on the BCPreds dataset.

It can be seen from Table **3** that, for the LBTope_fixed_nr dataset, compared with the model cross-validated on the BCPreds dataset, the model cross-validated on the mixed dataset achieves 19.1, 11.8, 176.9, 79.3, 329.2, and 28.5 percent improvements in Acc, PRE, Recall, F1, MCC, and AUC values, respectively. For the ibce-ind-full dataset, the Acc, PRE, Recall, F1, MCC, and AUC values of the model cross-validated on the mixed dataset are 14.1, 17.4, 133.5, 86.1, 171.1, and 27.8 percent higher than those of the model cross-validated on the BCPreds dataset. However, for ABCPred and Chen datasets, compared with the model cross-validated on the BCPreds dataset, the performance of the model cross-validated on the mixed dataset decreases. It can be interpreted that for LBTope_fixed_nr and ibce-ind-full datasets with large distribution differences from the BCPreds dataset, increasing the number of species in cross-validated datasets is helpful for the prediction performance, but for ABCPred and Chen datasets with little distribution differences from the BCPreds dataset, increasing the number of species in cross-validated datasets decreases the prediction performance, which may be due to the introduction of noise. The above results further support the conclusion that the prediction of linear BCEs is highly dependent on the species distribution of the dataset.

CONCLUSION

In this study, we propose a robust predictor, “CoBCEs,” for identifying linear B-cell epitopes (BCEs) across multiple

sequence datasets. The proposed CoBCEs method incorporates a heterogeneous feature set, including amino acid residue properties, text-based representations, pLM-based embeddings (protein vectors), and an improved DE-Graph. Then, the RF algorithm was employed as a feature selection technique to select the optimal attributes and mitigate redundancy and overfitting problems. We use the AdaBoost (Adaptive Boosting) ensemble algorithm, whose base classifier is Random Forest, as our predictive model, offering improved generalization performance compared to other machine learning models. Benchmarking outcomes exhibit that the proposed CoBCEs protocol outperforms existing advanced approaches for linear B-cell epitope prediction.

STUDY LIMITATIONS

Last but not least, we believe that due to the diversity of parameters, the selection of parameters in the proposed method achieves the best results. However, in experiments where the species differences between the training set and the testing set are too large, the predictive performance obtained by CoBCEs and existing predictors is low. Thus, to further enhance the predictive efficacy of CoBCEs, future research should focus on three key directions: [1] expanding the size of the dataset to improve the model's generalization performance, [2] developing more effective discriminative features to extract information from linear BCE sequences, such as structure-based features of BCEs, and [3] identifying strategies to address the impact of species differences on predictive performance. Beyond algorithmic improvements, CoBCEs has immediate translational potential-its high-accuracy predictions could accelerate epitope-based vaccine design, facilitate therapeutic antibody development, and support precision immunology applications. The standalone package of CoBCEs is freely available for academic use at: https://github.com/jun-csbio/CoBCEs/.

AUTHORS' CONTRIBUTIONS

The authors confirm their contribution to the paper as follows: study conception and design: BR and YXT, data collection and interpretation: HA and YSA, writing-review & editing: JH, formal analysis,investigation and supervision: MA,TA. All authors reviewed the manuscript and approved the final revision

LIST OF ABBREVIATIONS

AAC	=	Amino Acid Composition
AAP	=	Amino Acid Pair
AAs	=	Amino Acids
Acc	=	Accuracy
Adaboost	=	Adaptive Boosting
AUC	=	Area under the ROC Curve
BCEs	=	B-Cell Epitopes
BR	=	Bing Rao
CTD	=	Composition, Transition and, Distribution
CV	=	Cross-Validation Test
DE-Graph	=	Distance-Enhanced Graph
DPC	=	Dipeptide Composition
F1	=	F1-Score
HA	=	Hanin Alahmadi
HMM	=	Hidden Markov Model
JH	=	Jun Hu
MA	=	Muhammad Arif
MCC	=	Methew Coefficient Correlation
ML	=	Machine Learnig
pLM	=	Protein Language Model
PRE	=	Precision
RF	=	Random Forest
ROC Curve	=	Receiver Operating Characteristic
SVM	=	Support Vector Machine
TA	=	Tanvir Alam
Tces	=	T-Cell Epitopes
TF-IDF	=	Term Frequency-Inverse Document Frequency
YSA	=	Yaseer Saud Alqahtani
YT	=	Yuxuan Tang

ETHICS APPROVAL AND CONSENT TO PARTICIPATE

Not applicable.

HUMAN AND ANIMAL RIGHTS

Not applicable.

CONSENT FOR PUBLICATION

Not applicable.

AVAILABILITY OF DATA AND MATERIALS

The data and code are freely accessible at https://github.com/jun-csbio/CoBCEs/

FUNDING

This work was supported by the National Natural Science Foundation of China (No. 61902352 and 62072243), the Natural Science Foundation of Zhejiang (No. LY21F020025 and LZ20F030002) and the Fundamental Research Funds for the Provincial Universities of Zhejiang (No. RF-A20200012).

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

SUPPLEMENTARY MATERIAL

Supplementary material is available on the publisher's website along with the published article.

REFERENCES

[1] Ramalingam PS, Aranganathan M, Hussain MS, *et al.* Unveiling reverse vaccinology and immunoinformatics toward Saint Louis encephalitis virus: A ray of hope for vaccine development. Front Immunol 2025; 16: 1576557. http://dx.doi.org/10.3389/fimmu.2025.1576557 PMID: 40458397

[2] Ayesha S, Abdul Rahman S, Mohammed Gulzar A, Prajitha B, Sindhu Priya ES, Md Sadique H. Development, characterization, and evaluation of the antidepressant potential of Crocus sativus SLN nasal spray in a drosophila melanogaster model. Comb Chem High Throughput Screen 2025; 28: 1-23.

[3] Bahai A, Asgari E, Mofrad MRK, Kloetgen A, McHardy AC. EpitopeVec: Linear epitope prediction using deep protein sequence embeddings. Bioinformatics 2021; 37(23): 4517-25. http://dx.doi.org/10.1093/bioinformatics/btab467 PMID: 34180989

[4] Manavalan B, Govindaraj RG, Shin TH, Kim MO, Lee G. iBCE-EL: A new ensemble learning framework for improved linear B-cell epitope prediction. Front Immunol 2018; 9: 1695. http://dx.doi.org/10.3389/fimmu.2018.01695 PMID: 30100904

[5] Larsen J, Lund O, Nielsen M. Improved method for predicting linear B-cell epitopes. Immunome Res 2006; 2(1): 2. http://dx.doi.org/10.1186/1745-7580-2-2 PMID: 16635264

[6] EL-Manzalawy Y, Dobbs D, Honavar V. Predicting linear B-cell epitopes using string kernels. J Mol Recognit 2008; 21(4): 243-55. http://dx.doi.org/10.1002/jmr.893 PMID: 18496882

[7] Saha S, Raghava GPS. Prediction of continuous B-cell epitopes in an antigen using recurrent neural network. Proteins 2006; 65(1): 40-8. http://dx.doi.org/10.1002/prot.21078 PMID: 16894596

[8] Jespersen MC, Peters B, Nielsen M, Marcatili P. BepiPred-2.0: Improving sequence-based B-cell epitope prediction using conformational epitopes. Nucleic Acids Res 2017; 45(W1): W24-9. http://dx.doi.org/10.1093/nar/gkx346 PMID: 28472356

[9] Singh H, Ansari HR, Raghava GPS. Improved method for linear B-cell epitope prediction using antigen's primary sequence. PLoS One 2013; 8(5): 62216. http://dx.doi.org/10.1371/journal.pone.0062216 PMID: 23667458

[10] da Silva BM, Ascher DB, Pires DEV. epitope1D: Accurate taxonomy-aware B-cell linear epitope prediction. Brief Bioinform 2023; 24(3): bbad114. http://dx.doi.org/10.1093/bib/bbad114 PMID: 37039696

[11] Collatz M, Mock F, Barth E, Hölzer M, Sachse K, Marz M. EpiDope: A deep neural network for linear B-cell epitope prediction. Bioinformatics 2021; 37(4): 448-55. http://dx.doi.org/10.1093/bioinformatics/btaa773 PMID: 32915967

[12] Ge F, Muhammad A, Yu DJ. DeepnsSNPs: Accurate prediction of non-synonymous single-nucleotide polymorphisms by combining multi-scale convolutional neural network and residue environment information. Chemom Intell Lab Syst 2021; 215: 104326. http://dx.doi.org/10.1016/j.chemolab.2021.104326

[13] Ge F, Li HY, Zhang M, Arif M, Alam T. TCellPredX: A novel approach for accurate prediction of Hepatitis C virus linear T Cell epitopes. ACS Omega 2024; 9(52): 51494-507. http://dx.doi.org/10.1021/acsomega.4c08715 PMID: 39758636

[14] Arif M, Ali F, Ahmad S, Kabir M, Ali Z, Hayat M. Pred-BVP-Unb: Fast prediction of bacteriophage Virion proteins using unbiased multi-perspective properties with recursive feature elimination. Genomics 2020; 112(2): 1565-74. http://dx.doi.org/10.1016/j.ygeno.2019.09.006 PMID: 31526842

[15] Saha S, Bhasin M, Raghava GPS. Bcipep: A database of B-cell epitopes. BMC Genomics 2005; 6(1): 79. http://dx.doi.org/10.1186/1471-2164-6-79 PMID: 15921533

[16] Vita R, Mahajan S, Overton JA, *et al.* The Immune Epitope Database (IEDB): 2018 update. Nucleic Acids Res 2019; 47(D1): D339-43. http://dx.doi.org/10.1093/nar/gky1006 PMID: 30357391

[17] Arif M, Musleh S, Fida H, Alam T. PLMACPred prediction of anticancer peptides based on protein language model and wavelet denoising transformation. Sci Rep 2024; 14(1): 16992. http://dx.doi.org/10.1038/s41598-024-67433-8 PMID: 39043738

[18] Arif M, Kabir M, Ahmed S, *et al.* DeepCPPred: A deep learning framework for the discrimination of cell-penetrating peptides and their uptake efficiencies. IEEE/ACM Trans Comput Biol Bioinformatics 2022; 19(5): 2749-59. http://dx.doi.org/10.1109/TCBB.2021.3102133 PMID: 34347603

[19] Arif M, Ahmad S, Ali F, Fang G, Li M, Yu DJ. TargetCPP: Accurate prediction of cell-penetrating peptides from optimized multi-scale features using gradient boost decision tree. J Comput Aided Mol Des 2020; 34(8): 841-56. http://dx.doi.org/10.1007/s10822-020-00307-z PMID: 32180124

[20] Chou KC, Shen HB. Recent progress in protein subcellular location prediction. Anal Biochem 2007; 370(1): 1-16. http://dx.doi.org/10.1016/j.ab.2007.07.006 PMID: 17698024

[21] Bairoch A, Apweiler R. The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. Nucleic Acids Res 2000; 28(1): 45-8. http://dx.doi.org/10.1093/nar/28.1.45 PMID: 10592178

[22] Yao B, Zhang L, Liang S, Zhang C. SVMTrip: A method to predict antigenic epitopes using support vector machine to integrate tri-peptide similarity and propensity. PLoS One 2012; 7(9): e45152. http://dx.doi.org/10.1371/journal.pone.0045152 PMID: 22984622

[23] Lennox M, Robertson N, Devereux B. Expanding the vocabulary of a protein: Application of subword algorithms to protein sequence modelling. 42nd Annual International Conferences of the IEEE Engineering in Medicine and Biology Society. Montreal, QC, Canada, 20-24 July 2020, pp. 2361-2367. http://dx.doi.org/10.1109/EMBC44109.2020.9176380

[24] Gu X, Chen Z, Wang D. Prediction of G protein-coupled receptors with CTDC extraction and MRMD2. 0 dimension-reduction methods. Front Bioeng Biotechnol 2020; 8: 635. http://dx.doi.org/10.3389/fbioe.2020.00635 PMID: 32671038

[25] Manavalan B, Lee J. FRTpred: A novel approach for accurate prediction of protein folding rate and type. Comput Biol Med 2022; 149: 105911. http://dx.doi.org/10.1016/j.compbiomed.2022.105911 PMID: 36096036

[26] Zhang HQ, Liu SH, Li R, *et al.* MIBPred: Ensemble learning-based metal ion-binding protein classifier. ACS Omega 2024; 9(7): acsomega.3c09587. http://dx.doi.org/10.1021/acsomega.3c09587 PMID: 38405489

[27] Pham NT, Rakkiyapan R, Park J, Malik A, Manavalan B. H2Opred: A robust and efficient hybrid deep learning model for predicting 2′-O-methylation sites in human RNA. Brief Bioinform 2023; 25(1): bbad476. http://dx.doi.org/10.1093/bib/bbad476 PMID: 38180830

[28] Liu Y, Li H, Zeng T, *et al.* Integrated bulk and single-cell transcriptomes reveal pyroptotic signature in prognosis and therapeutic options of hepatocellular carcinoma by combining deep learning. Brief Bioinform 2023; 25(1): bbad487. http://dx.doi.org/10.1093/bib/bbad487 PMID: 38197309

[29] Musleh S, Arif M, Alajez NM, Alam T. Unified mRNA Subcellular Localization Predictor based on machine learning techniques. BMC Genomics 2024; 25(1): 151. http://dx.doi.org/10.1186/s12864-024-10077-9 PMID: 38326777

[30] Lin H. Computational methods and resources in biological and medical data. Curr Med Chem 2022; 29(5): 786-8. http://dx.doi.org/10.2174/0929867329052202141413331 PMID: 35227177

[31] Zulfiqar H, Dao FY, Lv H, *et al.* Identification of potential inhibitors against SARS-Cov-2 using computational drug repurposing study. Curr Bioinform 2021; 16(10): 1320-7. http://dx.doi.org/10.2174/1574893616666210726155903

[32] Ma CY, Luo YM, Zhang TY, *et al.* Predicting coronary heart disease in Chinese diabetics using machine learning. Comput Biol Med 2024; 169: 107952. http://dx.doi.org/10.1016/j.compbiomed.2024.107952 PMID: 38194779

[33] Zhang LQ, Yang H, Liu JJ, *et al.* Recognition of driver genes with potential prognostic implications in lung adenocarcinoma based on H3K79me2. Comput Struct Biotechnol J 2022; 20: 5535-46. http://dx.doi.org/10.1016/j.csbj.2022.10.004 PMID: 36249560

[34] Lv H, Dao FY, Zhang D, *et al.* iDNA-MS: An integrated computational tool for detecting DNA modification sites in multiple genomes. iScience 2020; 23(4): 100991. http://dx.doi.org/10.1016/j.isci.2020.100991 PMID: 32240948