Hyb_SEnc: An Antituberculosis Peptide Predictor Based on a Hybrid Feature Vector and Stacked Ensemble Learning

Xiuhao Fu[1], Hao duan[1], Xiaofeng Zang[1], Chunling Liu[1], Xingfeng Li[1], Qingchen Zhang[1], Zilong Zhang[1, *], Quan Zou[2,3, *], Feifei Cui[1, *]

[1] School of Computer Science and Technology, Hainan University, Haikou, 570228, China

[2] Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 610054, China

[3] Yangtze Delta Region Institute (Quzhou), University of Electronic Science and Technology of China, Quzhou 324000, China

*** Corresponding author:**

Zilong Zhang (zhangzilong@hainanu.edu.cn), Quan Zou (zouquan@nclab.net) and Feifei Cui (feifeicui@hainanu.edu.cn),.

Abstract—Tuberculosis has plagued mankind since ancient times, and the struggle between humans and tuberculosis continues. Mycobacterium tuberculosis is the leading cause of tuberculosis, infecting nearly one-third of the world's population. The rise of peptide drugs has created a new direction in the treatment of tuberculosis. Therefore, for the treatment of tuberculosis, the prediction of anti-tuberculosis peptides is crucial.This paper proposes an anti-tuberculosis peptide prediction method based on hybrid features and stacked ensemble learning. First, a random forest (RF) and extremely randomized tree (ERT) are selected as first-level learning of stacked ensembles. Then, the five best-performing feature encoding methods are selected to obtain the hybrid feature vector, and then the decision tree and recursive feature elimination (DT-RFE) are used to refine the hybrid feature vector. After selection, the optimal feature subset is used as the input of the stacked ensemble model. At the same time, logistic regression (LR) is used as a stacked ensemble secondary learner to build the final stacked ensemble model Hyb_SEnc. The prediction accuracy of Hyb_SEnc achieved 94.68% and 95.74% on the independent test sets of AntiTb_MD and AntiTb_RD, respectively. In addition, we provide a user-friendly Web server (http://www.bioai-lab.com/Hyb_SEnc). The source code is freely available at https://github.com/fxh1001/Hyb_SEnc.

Index Terms—anti-tuberculosis peptide, machine learning, recursive feature elimination, stacked ensemble

1 Introduction

TUBERCULOSIS (TB) has plagued mankind since ancient times. Historical records show that tuberculosis existed in the earliest urbanized societies[1]. According to the "Global Tuberculosis Report-2021," which was recently released by the WHO[2], tuberculosis remains a major threat to global public health. Since the German scientist Robert Koch delivered a famous speech in 1882, announcing the discovery of the pathogen that causes tuberculosis as Mycobacterium tuberculosis[3], humanity has been fighting against tuberculosis and there is a global goal to eliminate tuberculosis by 2035[4]. However, the cumulative reduction rate of TB incidence has not been ideal thus far, and therefore, there is a need to optimize existing treatment methods and discover new methods [2].

For the treatment of tuberculosis, from the effective drug streptomycin, which was discovered as early as 1944, to the "triple therapy" (streptomycin, p-aminosalicylic acid and isoniazid) invented in 1952, the discovery of isoniazid in the 1970s and rifampicin can greatly shorten the treatment time. Finally, in the 1980s, the addition of pyrazinamide to these drugs is able to cure tuberculosis in just six months[5]. However, the treatment of tuberculosis can still experience the problem of drug resistance. Therefore, the effectiveness of these treatments has decreased over time, and in extremely drug-resistant (XDR), completely drug-resistant (TDR) and multidrug-resistant strains (MDR)[6], the high lethality makes the situation even more severe, and the development of new treatments is urgently needed. The rise of peptide drugs in the past two decades has enabled peptide therapeutics to play an important role in various medical fields[7-10]. Additionally, anti-tuberculosis peptides have proven to be more promising anti-tuberculosis drugs[11, 12].

Anti-tuberculosis peptides are actually antibacterial peptides with anti-tuberculosis activity, which have many advantages such as low immunogenicity, selective affinity for bacterial negatively charged cell envelopes and different mechanisms of action[13, 14]. Therefore, the prediction of anti-tuberculosis

peptides is very important for the treatment of tuberculosis. Presently, a variety of machine learning methods have been proposed to predict anti-tuberculosis peptides and all of them have achieved good results. Support vector machine-based models, such as those developed by Usmani et al., use different sequential features[15-17]. Khatun et al. used the amino acid index, binary code, dipeptide composition and tripeptide composition as feature coding methods, adopted the support vector machine and random forest as prediction methods, and finally combined these two prediction models into the iAntiTB final prediction model[18]. Additionally, Manavalan et al. used a variety of encoding methods based on protein sequence characteristics and physical and chemical properties and used the extremely randomized tree (ERT) as a prediction method to build a two-layer model. The first layer used the extreme random tree and each encoding method one by one. Corresponding to multiple baseline models, the second layer stitched the prediction probabilities of multiple baseline models obtained by the first layer into a probability feature matrix as the input of the extreme random tree to obtain the AtbPpred final prediction model [19]. Then, Jain et al. used multiple feature encoding methods and feature selection methods of divergence measures and built the final prediction model based on voting ensemble learning to improve the prediction effect of anti-tuberculosis peptides[20]. Additionally, Akbar et al. used k-spaced amino acid pairs (KSAAP), composite physiochemical properties (CPP), and one-hot encoding (one-hot) as feature encoding methods[21] to create SVM[22], FKNN[23], RF[24], PNN[25], and KNN[26] as five algorithms, and then integrated these five models through the genetic algorithm to obtain the iAtbP-Hyb-Enc final integrated model [27]. These existing predictors have achieved relatively good prediction results, but there is still room to improve the generalization ability.

In this study, predictors are built based on hybrid features and stacked ensemble models. According to existing research, hybrid feature encoding often makes the model show better predictive performance than single feature encoding[28-32], and ensemble learning is widely used to improve the overall predictive performance[27]. The feature encoding methods used in this study are amino acid composition, amphiphilic pseudo amino acid composition, dipeptide composition, adaptive skipping dinucleotide composition, pseudo amino acid composition and quasi-sequence order, and composition/transition/distribution. A variety of machine learning algorithms are then used to evaluate the predictive performance of the feature vectors extracted by each feature encoding method including SVM [22], RF [24], GBDT [33], ERT [19], and XGB [34]. Then, according to the performance effects of these feature encoding methods on various machine learning algorithms, five feature encoding methods with better performance are finally determined, including amphiphilic pseudo amino acid composition, dipeptide composition, adaptive skipping dinucleotide composition, pseudo amino acid composition and quasi-sequence order. Furthermore, two machine learning algorithms, RF and ERT, are selected according to the performance of each machine learning algorithm to construct the final stacking ensemble model. These five encoding methods are used as the final feature extraction method, and these five feature encoding methods are hybridized at the same time to obtain hybrid features and recursive feature elimination is used for feature selection [35]. At the same time, machine learning algorithms, such as RF[35], DT[36], and ERT[37], are used for feature rating. By comparing the influence of the hybrid feature vectors selected by these three algorithms on the prediction effect of the final stacking ensemble model, the selection makes the DT-RFE algorithm with the best model prediction performance used for feature selection to obtain the final optimal feature subset. Finally, the stacking ensemble method is adopted[38] using the RF and ERT machine learning algorithms as the first-level learner, using its LR as the second-level learner, and using the optimal hybrid feature subset as input to build the stacking ensemble model Hyb_SEnc. The overall architecture of this study is shown in Figure 1. To verify the generalization ability of the model obtained in this study, two different independent test sets were used to judge whether the obtained model was overfitting and to address the overfitting problem.

The structure of the rest of the article is as follows. The second part provides a detailed description of the materials and methods used in the study. The third part analyzes the experimental results in detail. Finally, the fourth part summarizes the research.

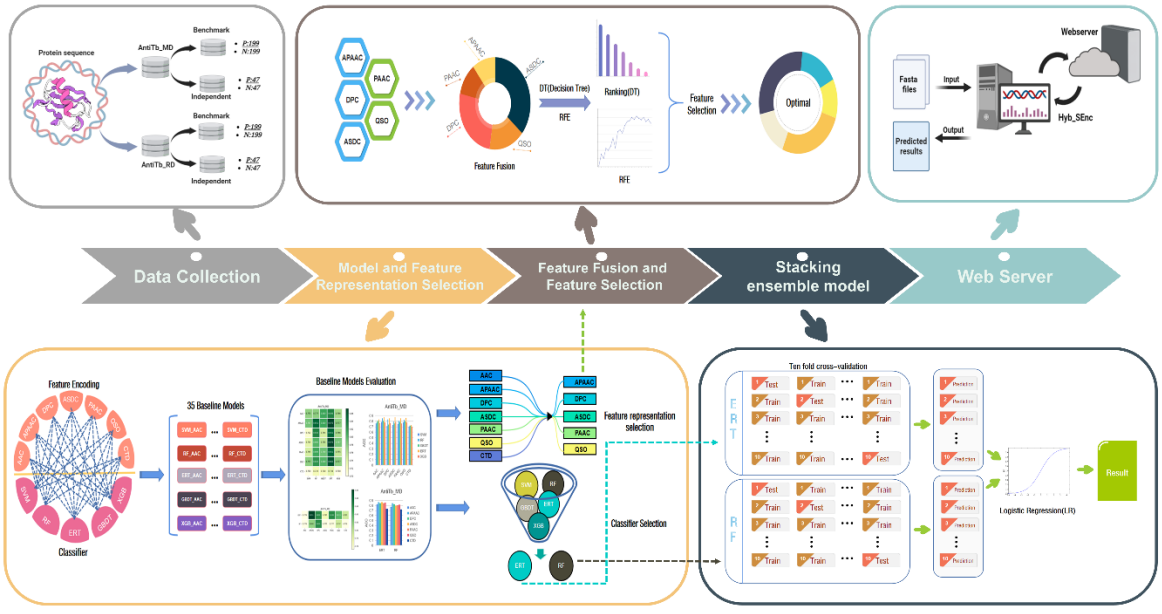


Fig. 1. The construction process of Hyb_SEnc

2 Materials and Methods

2.1 Dataset Collection

The selection of the dataset is very important when training a prediction model[39-41]. For example, choosing an effective dataset will often lead to better training of the model and thus a better prediction effect. In this study, we chose two separate anti-tuberculosis peptide datasets proposed by Usmani et al., which were AntiTb_MD and AntiTb_RD[15]. To obtain a positive dataset, that is, a dataset with anti-tuberculosis peptide activity, the authors extracted 246 unique peptides from the AntiTbPdb database [42] and each had a length between 5 and 61. To obtain negative datasets, that is, datasets without anti-TB peptide activity, they extracted unique peptides from the DBAASP database [43] and the Swiss-Prot database[44] to construct AntiTb_MD and AntiTb_RD, respectively. For AntiTb_MD, a large number of peptides were extracted from the DBAASP database through deredundancy operations, and 246 negative samples with the same number of positive dataset samples were obtained. For AntiTb_RD, 246 random peptides were generated from the Swiss-Prot database to construct negative samples in the dataset. Finally, 80% of the two datasets were taken as the training set after preprocessing. The two training sets obtained were the AntiTb_MD_benchmark and AntiTb_RD_benchmark. Each dataset consisted of a total of 398 samples made up of 199 positive samples and 199 negative samples. Then, the remaining 20% of the data was used to construct two independent test sets, which were Anti_MD_Ind and Anti_RD_Ind. Each independent test machine had a total of 94 samples consisting of 47 positive and 47 negative samples. The samples of each dataset obtained in the end were evenly distributed, which is ideal for experimental data. The sample distribution of each dataset is shown in Table 1.

TABLE I
Sample distribution of the AntiTb_MD and AntiTb_RD datasets

	Dataset	Positive	Negative
AntiTb_MD	Benchmark Dataset	199	199
	Independent Dataset	47	47
AntiTb_RD	Benchmark Dataset	199	199
	Independent Dataset	47	47

2.2 Feature encoding method

In this study, a total of 7 feature encoding methods were used including Amino Acid Composition (AAC), Amphiphilic Pseudo amino Acid Composition (APAAC), Dipeptide Composition (DPC), Adaptive Skip Dinucleotide Composition (ASDC), Pseudo Amino acid composition (PAAC) and quasi-sequence order (QSO). There was also a Composition/Transition/Distribution (CTD) method based on the physical and chemical properties to extract features. These feature encoding methods can be directly implemented on the latest feature extraction platforms to directly obtain the feature vectors extracted by each feature encoding method. These platforms include BioSeq-BLM[45], iFeature[46], and iLearnPlus[47].

2.3 Classifier

In the field of machine learning, the choice of the classifier is crucial. Selection of the appropriate classifier directly affects the prediction effect of the final model. In this study, a total of five machine learning-related algorithms were used including the support vector machine (SVM), random forest (RF), gradient boosting tree (GBDT), extremely randomized tree (ERT) and extreme gradient boosting (XGB). These five machine learning algorithms have been proven to play a satisfactory role in solving binary classification problems. Among the existing predictors developed for anti-TB peptide prediction, SVM, RF and ERT were used to construct the final predictor [15, 18, 19]. At the

same time, for GBDT and XGB machine learning algorithms, satisfactory prediction results were also obtained in other classification problems [33, 34, 48, 49]. Finally, we used scikit-learn (https://scikit-learn.org/) to implement these five machine learning algorithms and used a random grid search[50] to optimize the parameters.

2.4 Feature selection

The feature vector extracted by the feature encoding method, if the dimension is high, often has feature redundancy and contains many unnecessary features, which often results in the model showing unsatisfactory prediction results[51]. Feature selection is necessary to obtain ideal feature vectors. This is especially true for hybrid feature vectors, and to splice feature vectors obtained by different feature encoding methods together to obtain a feature vector with a higher dimension, feature selection is an essential step. Recently, scientists have proposed many methods for evaluating feature importance. There are the more commonly used analyses of variance (ANOVA)[52], maximum-relevance-maximum-distance (MRMD)[53] and so on. There are also some commonly used algorithms for feature selection such as SFS and RFE[54].

In this study, we initially used random forest (RF) [35], decision tree (DT) [36], and extreme random tree (ERT) [37] to rank the features and RFE was used to search and construct the optimal feature subset[36]. By comparing the influence of the optimal feature subset obtained by these three machine learning algorithms and RFE on the prediction effect of the final model on the independent test set, we determined which of the three machine learning algorithms can make the model perform on the independent test set. The best machine learning algorithms are then used to rank the features. Here, we briefly describe the steps of feature selection. The first step is to input the original feature vector with the label value into the machine learning algorithm. We use three different machine learning algorithms, i.e., random forest (RF) for illustration and the remaining two algorithms for feature selection. The general steps are the same. The original feature vector is input into the RF for training, and the feature importance index of each feature is obtained through the built-in function of the RF and then sorted according to the level of the feature importance index to obtain a list of feature importance indices from high to low. The higher the feature importance index of a feature is, the richer the information of the feature, which plays an important role in improving the classification efficiency. In contrast, the lower the feature importance index of a feature, the less information the feature carries and the more likely it is to be useless; that is, it cannot improve or even reduce the classification efficiency. The second part uses the RFE algorithm to search for the optimal feature subset. Through the feature sorting list obtained in the previous round, the RFE algorithm removes the useless features that rank last. Then, the dimension of the feature vector is reduced by one, and iteratively, one feature is eliminated in each round (i.e., the dimension of the feature vector in each round is reduced). One dimension is reduced, and at the same time, the new feature subset obtained in each round is input into the classifier, a stacked ensemble model is built, and tenfold cross-validation[55] is used for evaluation until a certain subset of obtained features makes the performance of the final stacked ensemble model on the independent test set the best. Then, this subset of features is considered the final optimal feature subset.

2.5 Ensemble classification

Integrated systems have received increasing attention over the past few decades due to their effectiveness and versatility in many fields[56]. This attention is justified because various machine learning problems, such as feature selection and incremental learning, have been successfully solved by integrating a system[57]. Thus, ensemble methods are considered state-of-the-art for many machine learning problems[58]. There are also many methods for building integrated systems such as blending and stacking ensemble methods and each has its own advantages[59, 60]. This study adopted a stacked ensemble approach to build an ensemble model. Here, a brief introduction to the ensemble principle and process of the stacking ensemble method is provided. First, the training dataset is input into several first-level learners, that is, the first layer of the stacked ensemble, and then a cross-validation is performed. The prediction results of the validation set in each fold of the cross-validation are spliced together to obtain a one-dimensional feature vector, and the one-dimensional feature vectors from these several first-level learners are then spliced through the above cross-validation to obtain a new feature vector, which is used as the input of the second layer of the stacking ensemble. Since each first-level learner can obtain a one-dimensional vector, that is, it can provide one-dimensional features, the dimension of this newly constructed feature vector is the same as the number of first-level learners. This newly obtained feature vector is then fed into the secondary learner for training, which is the second layer of the stacked ensemble. The final prediction result of the second-level learner is the final prediction result of the stacking ensemble[57]. The entire stacking ensemble algorithm process is shown in Figure 2.

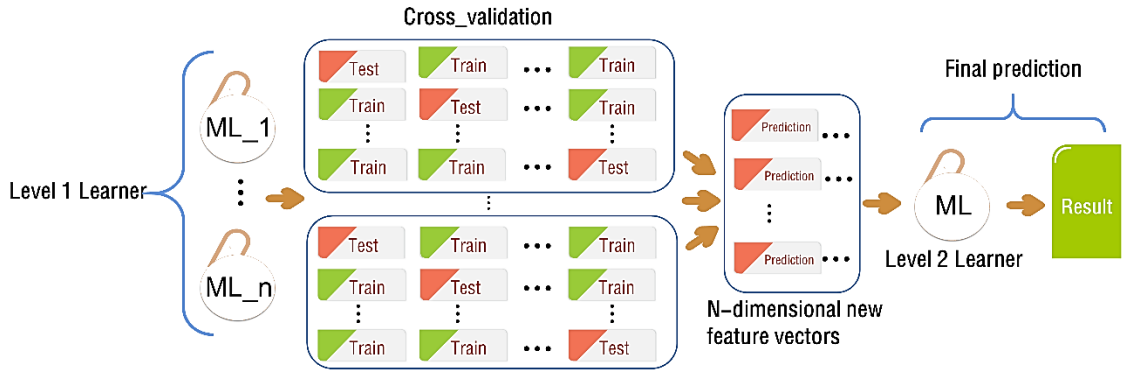


Fig. 2. The entire stacking ensemble algorithm process

2.6 Performance evaluation

A tenfold cross-validation process was used in this study to prevent overfitting and evaluate the proposed machine learning model[61-63]. Tenfold cross-validation divides the original training dataset into ten random and equal parts. Among these parts, 9 datasets are used for training and the remaining one is used as a validation set. Finally, the average performance of the ten subsets is obtained as the result of the tenfold cross-validation. Furthermore, in binary classification tasks, various evaluation indicators are often constructed through values in the confusion matrix[64]. This study used four evaluation indicators that were constructed based on the confusion matrix as follows: accuracy (ACC), sensitivity (SE), specificity (SP) and Matthew's correlation coefficient (MCC). The calculation formulas of these four evaluation indicators are shown in (1)-(4). To understand the pros and cons of the model performance, we use the receiver operating characteristic curve (ROC) and the area under the ROC curve (AUC)[65] to make a comprehensive comparison so that we can better evaluate the performance of the model obtained in this study.

$$ACC = \frac{TP+TN}{TP+FP+TN+FN}. \quad (1)$$

$$SE = \frac{TP}{TP+FN}. \quad (2)$$

$$SP = \frac{TN}{TN+FP}. \quad (3)$$

$$MCC = \frac{(TP*TN)-(FP*FN)}{\sqrt{(TP+FP)*(TN+FN)*(TP+FN)*(TN+FP)}}. \quad (4)$$

TP: true positive; FP: false seropositive; TN: true negative; FN: false-negative

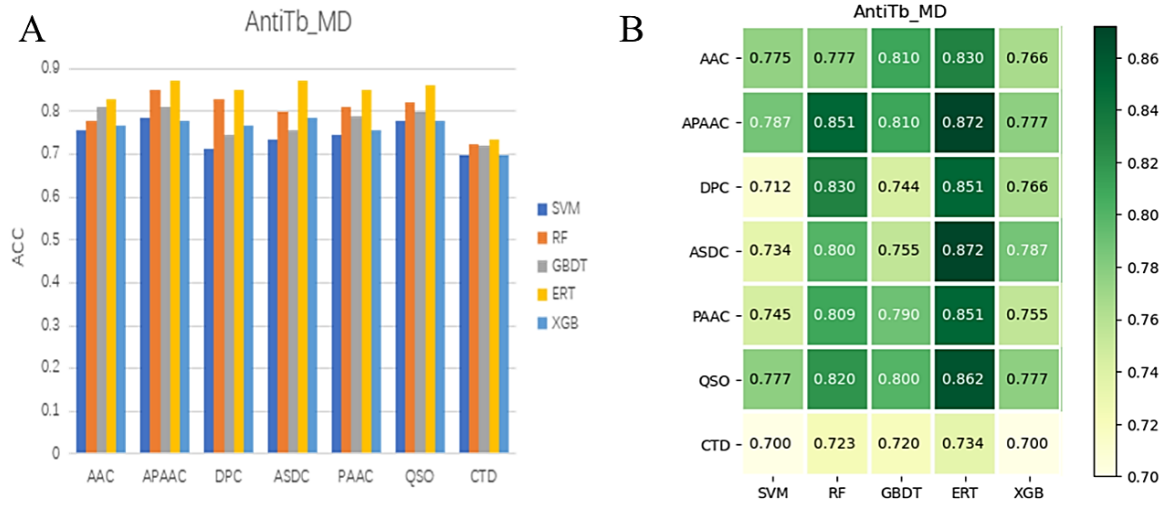


Fig. 3. The performance of five different machine learning algorithms on seven different feature encoding methods on the AntiTb_MD benchmark dataset (A), and the classification heatmap of 35 baseline models with respect to accuracy (B).

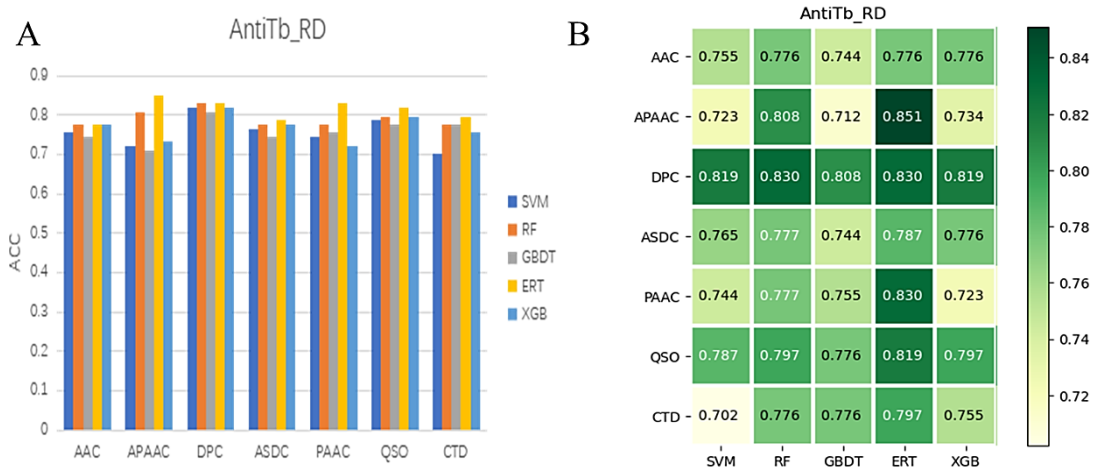


Fig. 4. The performance of five different machine learning algorithms on seven different feature encoding methods on the AntiTb_RD benchmark dataset (A), and a classification heatmap of 35 baseline models with respect to accuracy (B)

3 Experimental Results

3.1 The evaluation of different machine learning models using seven different feature encoding methods

This study used seven different feature encoding methods, and six are classical feature encoding methods based on the following protein sequences: amino acid composition (AAC), amphiphilic pseudo amino acid composition (APAAC), dipeptide composition (DPC), adaptive skip dinucleotide composition (ASDC), pseudo amino acid composition (PAAC) and quasi-sequence order (QSO). There is also a feature coding method, composition/transition/distribution (CTD), which is based on physical and chemical properties. In the initial stage of this study, five machine learning algorithms were used as follows: support vector machine (SVM), random forest (RF), gradient boosting tree (GBDT), extremely randomized tree (ERT) and extreme gradient boosting (XGB) [19, 22, 24, 33, 34]. Usually, the choice of the machine learning algorithm is not determined at the beginning but needs to be compared through experiments to select the best algorithm for the experimental dataset[19, 20]. At the very beginning of this research, we evaluated the five different machine learning models mentioned above by using the seven different feature encoding methods to determine the machine learning model that performs best among the seven feature encoding methods. The performance was determined based on the prediction accuracy (ACC). A feature encoding method corresponds to a machine learning model and so 35 (7*5) baseline models are generated. To prevent the trained model from overfitting and enhance the robustness of the model, we used tenfold cross-validation to train the initial baseline model and random grid search to tune the hyperparameters of each baseline model.

Figure 3A and B show the performance histogram and classification heatmap of the 35 baseline models trained on the AntiTb_MD dataset, respectively. According to Figure 3 A and B, it can be seen that the two machine learning models, ERT and RF, perform better than other machine learning models in the AntiTb_MD dataset in terms of the seven different feature encodings. From a data point of view, the average accuracy rates of SVM, RF, GBDT, ERT and XGB on these seven feature encoding methods are 0.744, 0.801, 0.775, 0.839 and 0.761, respectively. The ERT model performs best overall followed by the RF model. Examining the performance histogram and classification heatmap of the baseline model on the AntiTb_RD dataset, as shown in Figure 4A and B, it can also be seen that ERT and RF perform better in most feature encodings. Similarly, in the AntiTb_RD dataset, the average accuracy rates obtained by SVM, RF, GBDT, ERT and XGB on these seven feature encoding methods are 0.756, 0.791, 0.759, 0.813 and 0.768, respectively. The ERT model performs the best followed by the RF model. According to the above analysis in the AntiTb_MD and AntiTb_RD datasets, these two datasets, the ERT and RF models, perform better than the other models. Therefore, the ERT and RF models were used as the first-level learner for the stacked ensemble.

This article has been accepted for publication in IEEE/ACM Transactions on Computational Biology and Bioinformatics. This is the author's version which has not been fully edited content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2024.3425644

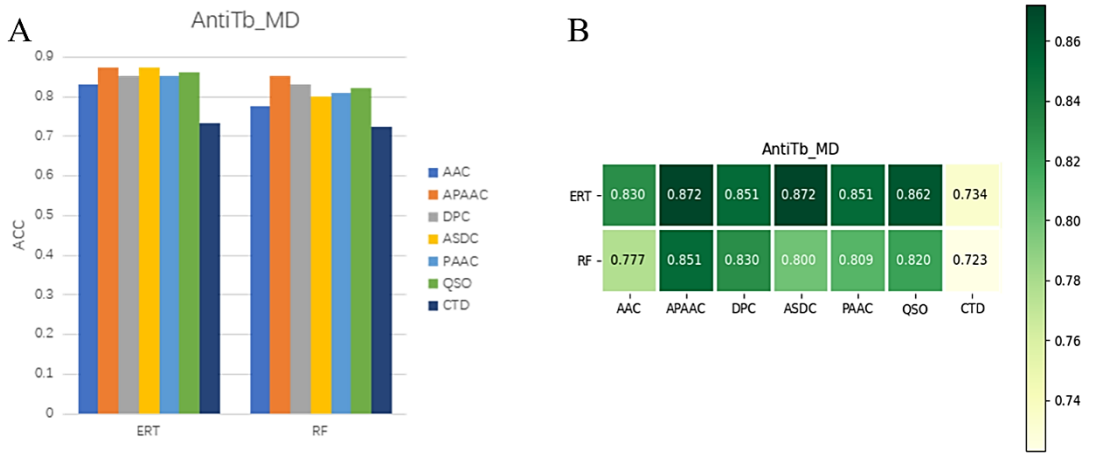


Fig. 5. The performance effects of seven different feature encoding methods on the ERT and RF models on the AntiTb_MD benchmark dataset (A) and the classification heatmap of 14 baseline models with respect to accuracy (B).

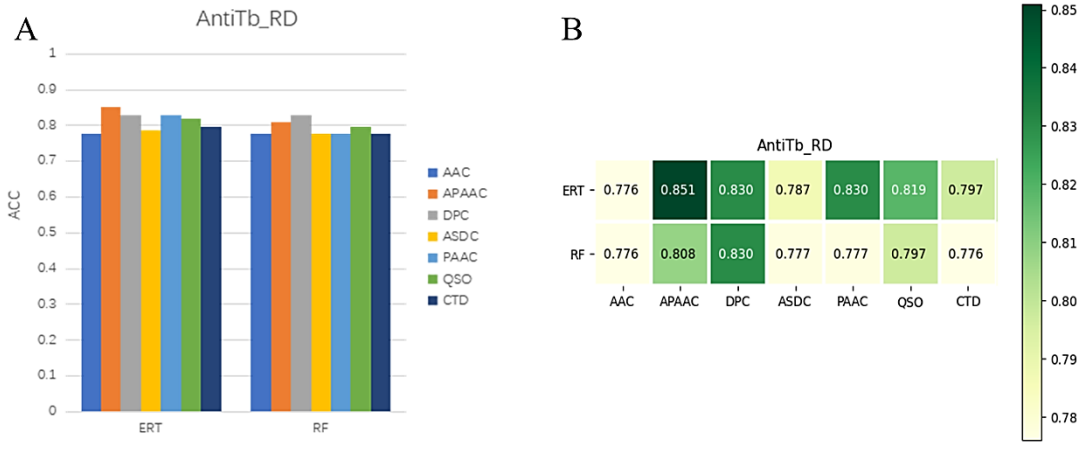


Fig. 6. The performance effects of seven different feature encoding methods on the ERT and RF models in the AntiTb_RD benchmark dataset (A), and a classification heatmap of 14 baseline models with respect to accuracy (B).

3.2 The feature encoding method that performs best on the ERT and RF models

According to the experimental results in the previous step, we choose the ERT and RF models as the first-level learner for the stack ensemble. However, to further improve the performance of the final integrated model, we also needed to select the feature encoding method that performs better on the ERT and RF models. Similarly, in the AntiTb_MD and AntiTb_RD datasets, we conducted experiments as shown in Figure 5A and B and Figure 6A and B. We selected the feature encoding method with the best performance. This performance is based on the prediction accuracy (ACC) of the baseline model after training. According to Figure 5A and B, it can be seen that on the AntiTb_MD dataset, amino acid composition (AAC) and physicochemical properties (CTD) showed worse results than other feature encoding methods on ERT and RF models. We used the mean of the accuracy rate (ACC) of each feature encoding method on the ERT and RF models for comparison, and the mean accuracy rate of AAC, APAAC, DPC, ASDC, PAAC, QSO, and CTD on the ERT and RF models on the AntiTb_MD dataset were 0.804, 0.862, 0.841, 0.836, 0.830, 0.841, and 0.729, respectively. The CTD feature encoding method shows the worst effect on the two models, and the APAAC feature encoding method shows the best effect on the two models. According to Figure 6A and B, on the AntiTb_RD dataset, AAC and CTD show poorer results than other feature encoding methods on the ERT and RF models. Similarly, we use the mean of the accuracy rate (ACC) of each feature encoding method on the ERT and RF models for comparison, the mean accuracy rates of AAC, APAAC, DPC, ASDC, PAAC, QSO, and CTD on the ERT and RF models on the AntiTb_RD dataset were 0.776, 0.830, 0.830, 0.782, 0.804, 0.808, and 0.787, respectively. Then, the AAC feature encoding method showed the worst effect on the AntiTb_RD dataset, and APAAC and DPC showed the same optimal results. According to the above experiments, in the AntiTb_MD and AntiTb_RD datasets, the CTD and AAC feature encoding methods showed the worst results, and so these two feature encoding methods were eliminated. The remaining five feature encoding methods of APAAC, DPC, ASDC, PAAC, and QSO were retained for the next experiment.

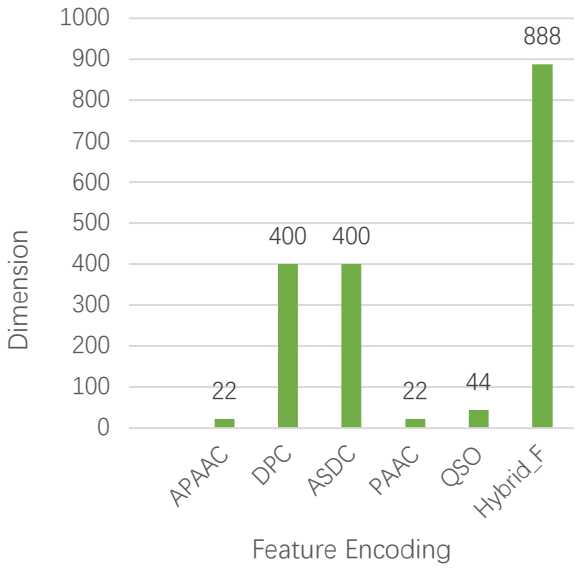


Fig. 7. The dimensions of the feature vectors obtained by seven feature encoding methods and the dimensions of the hybrid feature vectors.

3.3 Construct hybrid features and perform feature selection

After previous experiments, we chose the ERT and RF models to build the final stacked ensemble model and selected five feature encoding methods as follows: APAAC, DPC, ASDC, PAAC, and QSO. At this step, we use these five feature encoding methods to build a hybrid feature vector as Hybrid_F. First, we directly stitch the feature vectors obtained by these five feature encoding methods to obtain the initial hybrid feature vector Hybrid_F. The feature dimensions of APAAC, DPC, ASPC, PAAC, QSO and Hybrid_F are shown in Figure 7.

The feature vector dimension after mixing reaches 888 dimensions and the dimension is high. Considering that there are redundancies, feature selection must be conducted. Before feature selection and to determine whether the hybrid feature vector Hybrid_F will improve the performance of the model, we use five feature encoding methods to extract feature vectors from the AntiTb_MD and AntiTb_RD datasets and obtain the corresponding hybrid feature vector Hybrid_F. Then, Hybrid_F is used to train the ERT and RF models, and it also uses tenfold cross-validation and random grid searching for training and tuning. Tables 2 and 3 represent the performance effects of different feature encoding methods on the Antitb_MD and Antitb_RD datasets, respectively. At this time, the hybrid feature vector does not perform feature selection. Table 2 shows that for the ERT model and on the Antitb_MD dataset, the hybrid feature vector Hybrid_F obtained the highest ACC with a value of 0.883, the highest MCC with a value of 0.766, and the highest SP with a value of 0.872. APAAC and ASDC obtained the highest SE with a value of 0.936. Overall, Hybrid_F performed best on the Antitb_MD dataset. This shows that the hybrid features effectively integrate the important information contained in each feature encoding. However, Hybrid_F has not shown the best effect on each evaluation indicator in the AntiTb_RD dataset. However, its results are not bad. The ACC reached 0.829, which is 0.022 from the highest value of 0.851, and this result is acceptable. Overall, APAAC performed best on the Antitb_RD dataset. It can be seen that APAAC shows excellent performance on both AntiTb_MD and AntiTb_RD datasets, and APAAC makes a huge contribution to the performance of the ERT model. Next, the performance effects on the RF model are examined. According to Table 3 and for the RF model on the Antitb_MD dataset, the hybrid feature vector Hybrid_F shows the best ACC with a value of 0.851, the highest MCC with a value of 0.702, and the highest SE with a value of 0.872. APAAC also shows good performance on the AntiTb_MD dataset. Similarly, for the RF model, the hybrid feature vector Hybrid_F shows excellent effects on the Antitb_MD dataset. However, the RF model uses a hybrid feature vector on the Antitb_RD dataset, as the ERT model does not show a bright result. For the highest ACC of 0.830, the hybrid feature vector, Hybrid_F, obtained an accuracy of 0.809 (ACC) rather than only 0.021, which is also within the acceptable range. Meanwhile, DPC obtained the highest ACC and MCC, APAAC obtained the highest SP. After the above comparison, we can see that for both ERT and RF, APAAC extracted important information on both AntiTb_MD and AntiTb_RD datasets to make a useful contribution to the classification effect of the model, and the hybrid feature vector has the potential to make the model effect better, but it performs slightly worse on the AntiTb_RD dataset. Furthermore, the dimension of the hybrid feature vector is large, and the hybrid feature vector that has no feature selection is redundant. Therefore, the feature selection needs to be performed on the hybrid feature vector Hybrid_F.

Next, we perform feature selection on Hybrid_F and compare it with the feature vectors obtained by other feature encodings. As shown in Figure 7 and in addition to the hybrid feature vector, the maximum dimension of the feature vector extracted by other feature encoding methods is 400. That is, in this experiment, the dimension of the feature selection for hybrid feature Hybrid_F will initially be 400, and the feature vector after feature selection is recorded as Hybrid_400. A preliminary examined looked at whether the effect of the hybrid feature vector will be improved after rough feature selection. A decision tree (DT) is used for feature ranking, and recursive feature elimination (RFE) is used for feature selection. Table 4 and Table 5 show the performance effect of the hybrid feature vector after rough feature selection on the ERT and RF models compared with the hybrid feature without feature selection. Table 4 and Table 5 show that on the ERT model, the hybrid feature vector Hybrid_400 after feature selection improved on the AntiTb_MD dataset compared to Hybrid_F in all evaluation indicators; AAC reached 0.904, MCC reached 0.808, SE reached 0.915, and SP reached 0.894. Each evaluation indicator did not change on the AntiTb_RD dataset, which indicated that there may still be redundancy in the 400-dimensional feature vector after feature selection. On the RF model, Hybrid_400 achieved AAC of 0.872, MCC of 0.745, SE of 0.894, and SP of 0.851 on the AntiTb_MD dataset. On the AntiTb_RD dataset, AAC reached 0.830, MCC reached 0.662, SE reached 0.872, and SP reached 0.787. All evaluation indicators on the two datasets were improved. Therefore, feature selection is necessary, and feature selection can make the model perform better and lay the foundation for the next experiment.

TABLE II
In the ERT model, the performance of APAAC, DPC, ASDC, PAAC, QSO and Hybrid_F on the benchmark datasets of AntiTb_MD and AntiTb_RD. Bold fonts indicate the best results.

	Dataset	Feature Encoding	ACC	MCC	SE	SP
ERT	AntiTb_MD	APAAC	0.872	0.750	**0.936**	0.808
		DPC	0.851	0.702	0.872	0.829
		ASDC	0.872	0.750	**0.936**	0.808
		PAAC	0.851	0.702	0.829	**0.872**
		QSO	0.861	0.724	0.893	0.829
		Hybrid_F	**0.883**	**0.766**	0.893	**0.872**
	AntiTb_RD	APAAC	**0.851**	**0.702**	0.829	**0.872**
		DPC	0.83	0.662	0.872	0.787
		ASDC	0.787	0.579	0.851	0.723
		PAAC	0.83	0.675	0.936	0.723
		QSO	0.819	0.664	**0.957**	0.68
		Hybrid_F	0.829	0.669	0.914	0.745

TABLE III
For the RF model, the performance of APAAC, DPC, ASDC, PAAC, QSO and Hybrid_F on the benchmark datasets of AntiTb_MD and AntiTb_RD. Bold fonts indicate the best results.

	Dataset	Feature Encoding	ACC	MCC	SE	SP
RF	AntiTb_MD	APAAC	**0.851**	**0.702**	0.851	**0.851**
		DPC	0.830	0.662	**0.872**	0.787
		ASDC	0.800	0.602	**0.872**	0.723
		PAAC	0.809	0.618	0.83	0.787
		QSO	0.820	0.642	**0.872**	0.723
		Hybrid_F	**0.851**	**0.702**	**0.872**	0.829
	AntiTb_RD	APAAC	0.808	0.617	0.829	**0.787**
		DPC	**0.830**	**0.669**	0.915	0.745
		ASDC	0.777	0.553	0.787	0.766
		PAAC	0.777	0.560	0.851	0.702
		QSO	0.798	0.629	**0.957**	0.638
		Hybrid_F	0.809	0.620	0.851	0.766

TABLE IV
For the ERT model, the performance of Hybrid_400 with rough feature selection and Hybrid_F without feature selection. Bold fonts indicate the best results.

	Dataset	Feature Encoding	ACC	MCC	SE	SP
ERT	AntiTb_MD	APAAC	0.872	0.750	**0.936**	0.808
		DPC	0.851	0.702	0.872	0.829
		ASDC	0.872	0.750	**0.936**	0.808
		PAAC	0.851	0.702	0.829	0.872
		QSO	0.861	0.724	0.893	0.829
		Hybrid_F	0.883	0.766	0.893	0.872
		Hybrid_400	**0.904**	**0.808**	0.915	**0.894**
	AntiTb_RD	APAAC	**0.851**	**0.702**	0.829	**0.872**
		DPC	0.83	0.662	0.872	0.787
		ASDC	0.787	0.579	0.851	0.723
		PAAC	0.83	0.675	0.936	0.723
		QSO	0.819	0.664	**0.957**	0.68
		Hybrid_F	0.829	0.669	0.914	0.745
		Hybrid_400	0.829	0.669	0.914	0.745

TABLE V
For the RF model, the performance of Hybrid_400 with rough feature selection and Hybrid_F without feature selection. Bold fonts indicate the best results.

	Dataset	Feature Encoding	ACC	MCC	SE	SP
RF	AntiTb_MD	APAAC	0.851	0.702	0.851	**0.851**
		DPC	0.830	0.662	0.872	0.787
		ASDC	0.800	0.602	0.872	0.723
		PAAC	0.809	0.618	0.830	0.787
		QSO	0.820	0.642	0.872	0.723
		Hybrid_F	0.851	0.702	0.872	0.829
		Hybrid_400	**0.872**	**0.745**	**0.894**	**0.851**
	AntiTb_RD	APAAC	0.808	0.617	0.829	**0.787**
		DPC	**0.830**	**0.669**	0.915	0.745
		ASDC	0.777	0.553	0.787	0.766
		PAAC	0.777	0.560	0.851	0.702
		QSO	0.798	0.629	**0.957**	0.638
		Hybrid_F	0.809	0.620	0.851	0.766
		Hybrid_400	**0.830**	0.662	0.872	**0.787**

3.4 Building a stacking ensemble (Stacking) model

After the previous experiments were completed, this study chose two machine learning algorithms, ERT and RF, and concluded that the hybrid feature encoding method has a better effect on improving the performance of the model. To prevent feature redundancy, feature selection is necessary. Therefore, at this stage, the ERT model and RF model are used as the first-level classifier, and LR (logistic regression) is used as the second-level classifier to build a stacking ensemble (Stacking)

model. The hybrid feature matrix Hybrid_F is used as the input of the integrated model, and recursive feature elimination (RFE) is used for feature selection. Initially, the three machine learning methods of RF, DT, and ERT were used for the feature rating and combined with RFE to perform feature selection on the hybrid feature matrix Hybrid_F using tenfold cross-validation and a random grid search for training and parameter adjustment. According to the model performance, the feature rating method is selected that can make the model perform best from the above three machine learning methods. The feature selection in building the final stacked ensemble model is the fine feature selection and not the rough feature selection in Section 3.3. The goal of feature selection at this stage is to select the feature vectors that maximize the performance of the stacked ensemble model. In the initial stage, this study uses three methods of RF-RFE, DT-RFE, and ERT-RFE for feature selection, and the feature vectors obtained after feature selection by each method are used to train the stacked ensemble model Hyb-Senc. The performance effects of the three Hyb_SEnc stacked ensemble models obtained on the independent test sets of the AntiTb_MD dataset and the AntiTb_RD dataset are shown in Table 6.

Table 6 shows that on the independent test dataset of AntiTb_MD, the feature selection results based on DT-RFE show the best overall effect and performed best on the ACC, MCC, SE and SP evaluation indicators by reaching 0.946, 0.893, 0.957 and 0.936, respectively, and achieved the best results on all evaluation indicators. Similarly, on the independent test dataset of AntiTb_RD, the feature selection results based on DT-RFE showed the best performance and performed best on the four indicators of ACC, MCC, SE, and SP in reaching 0.957, 0.916, 0.936 and 0.978, respectively. Thus, using the feature selection method based on DT-RFE for these two datasets can make the final stacking ensemble model show the strongest performance and provides better generalization ability. Therefore, we decided to use the feature selection method of DT-RFE to perform the fine feature selection on the hybrid feature vector. Feature selection was performed on the AntiTb_MD dataset and the AntiTb_RD dataset, and the accuracy of the stacked ensemble model Hyb-SEnc on the respective independent test sets is shown in Figure 8A and B, respectively. According to Figure 8A, on the AntiTb_MD dataset, DT-RFE performs feature selection so that Hyb-SEnc performs best with a feature dimension of 393. Similarly, as shown in Figure 8B, for the AntiTb_RD dataset using DT-RFE for feature selection makes Hyb-SEnc perform best with a feature dimension of 389. For these two datasets, the feature dimension that makes Hyb-SEnc perform best is close.

Finally, this study compared the performance of the obtained Hyb-SEnc integrated predictor on the independent test set with other existing predictors, and the comparison results are shown in Table 7. According to Table 7 and the independent test set of the AntiTb_MD dataset, Hyb-SEnc showed the best ACC, MCC and SE, which were 94.68%, 0.89 and 95.74%, respectively. On the AntiTb_RD dataset, Hyb-SEnc showed the best ACC, MCC, SE and SP, which were 95.74%, 0.91, 93.61% and 97.87%, respectively. In terms of prediction accuracy, Hyb-SEnc showed the best results in the independent test sets of the AntiTb_MD dataset and AntiTb_RD dataset. Hyb_SEnc outperforms the best existing anti-TB peptide predictors by 2.00% and 3.19% in prediction accuracy on the independent test sets of AntiTb_MD and AntiTb_RD, respectively. Therefore, Hyb-SEnc is currently the best performing model regarding prediction accuracy. Similarly, Hyb_SEnc achieved the best MCC on the independent test sets of AntiTb_MD and AntiTb_RD, which were 0.89 and 0.87, respectively. At the same time, Hyb-SEnc showed the best SE on the independent test set of the AntiTb_MD dataset, and Hyb_SEnc achieved the best results in all evaluation indicators on the independent test set of AntiTb_RD. The ROC analysis of Hyb_SEnc on the independent test sets of AntiTb_MD and AntiTb_RD is shown in Figures 9A and B. On the independent test sets of AntiTb_MD and AntiTb_RD and after ROC analysis, the areas under the ROC curve (AUC) obtained by Hyb_SEnc were 0.94, 0. 98, respectively. This result is very ideal. Therefore, in general, Hyb-SEnc shows the best overall effect and is currently the best model for predicting anti-tuberculosis peptides. At the same time, in order to further verify the effectiveness of the ensemble learning strategy, we conducted an ablation experiment, using the features used by Hyb-SEnc to train the ERT and RF models, and then compared their performance. The results are shown in Table 8. It can be seen that the integrated learning strategy greatly improves the prediction ability of the model.

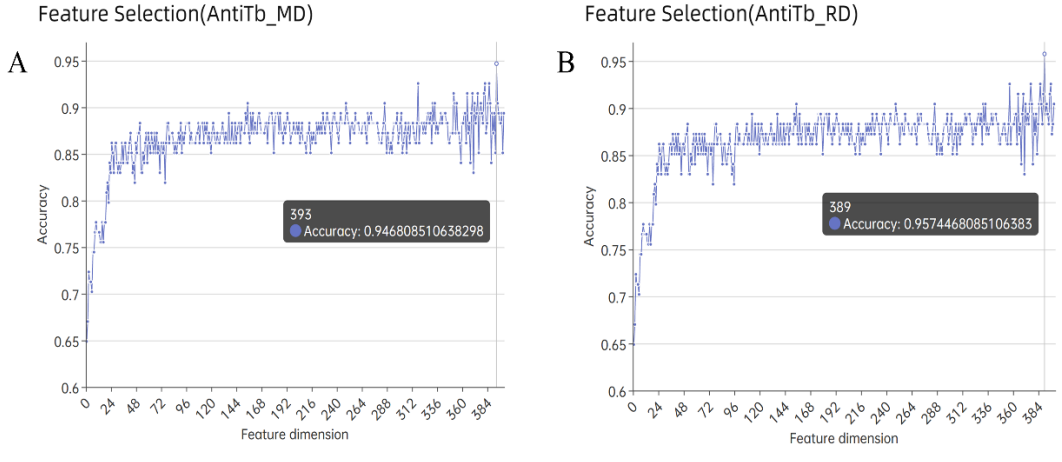


Fig. 8. Fine feature selection is performed on AntiTb_MD, and the feature vector dimension corresponds to the model performance accuracy (A). Fine feature

selection is performed on AntiTb_RD, and the feature vector dimension corresponds to the model performance accuracy (B).

TABLE VI
The performance of Hyb_SEnc constructed using different feature selection methods on the independent test sets of AntiTb_MD and AntiTb_RD. Bold fonts indicate the best results.

	Dataset	Feature Select	ACC	MCC	SE	SP
Hyb-SEnc	AntiTb_MD	DT-RFE	**0.946**	**0.893**	**0.957**	**0.936**
		RF-RFE	0.915	0.83	0.893	**0.936**
		ERT-RFE	0.904	0.81	0.872	**0.936**
	AntiTb_RD	DT-RFE	**0.957**	**0.915**	**0.936**	**0.978**
		RF-RFE	0.904	0.81	0.872	0.936
		ERT-RFE	0.893	0.794	0.829	0.957

TABLE VII
Performance of Hyb_SEnc on independent test sets of AntiTb_MD and AntiTb_RD compared with existing predictors of anti-tuberculosis peptides. Bold fonts indicate the best results.

	Predictors	ACC(%)	MCC(%)	SE(%)	SP(%)
MD Independent dataset	Antitbpred	75.90	52.00	75.01	76.70
	AtbPpred	89.40	79.00	83.01	**95.70**
	iAtbP-Hyb-EnC	92.68	**89.00**	95.24	92.96
	Hyb-SEnc	**94.68**	**89.00**	**95.74**	93.61
RD Independent dataset	Antitbpred	78.50	57.00	73.30	83.80
	AtbPpred	85.10	70.00	80.90	89.40
	iAtbP-Hyb-EnC	92.55	85.00	93.04	91.87
	Hyb-SEnc	**95.74**	**91.60**	**93.61**	**97.87**

TABLE VIII
Performance of Hyb_SEnc on independent test sets of AntiTb_MD and AntiTb_RD compared with ERT and RF. Bold fonts indicate the best results.

Dataset	Predictors	ACC(%)	MCC(%)	SE(%)	SP(%)
MD Independent dataset	ERT	90.42	80.86	91.48	89.36
	RF	87.23	75.08	93.61	80.85
	Hyb-SEnc	**94.68**	**89.00**	**95.74**	**93.61**
RD Independent dataset	ERT	87.23	75.08	80.85	93.61
	RF	86.17	73.70	**95.74**	76.59
	Hyb-SEnc	**95.74**	**91.60**	93.61	**97.87**

3.5 Implementation of the Web server

To make the final integrated model Hyb_SEnc obtained in this study more widely used, we established a user-friendly web server(http://www.bioai-lab.com/Hyb_SEnc)to promote the practical application of Hyb_SEnc. Meanwhile, datasets used in this study are all freely available for download on our web server. Next, we briefly introduce the steps of using the web server. In the first step, the user enters the sequence into the query box. Note that the input sequence must be in the Fasta sequence format. There is an example of the input sequence below the query box. Or you can upload the fasta sequence file by clicking the upload button. In the second step, the user selects a classifier. This research finally obtained two different Hyb_SEnc integrated models through two different datasets, and the user can choose one of these two models for classification. In the third step, the submit button is used to analyze the input sequence and

obtain the results. Users can enter up to 2000 sequences in a single run. The purpose of providing a web server is to make our research results more widely used.

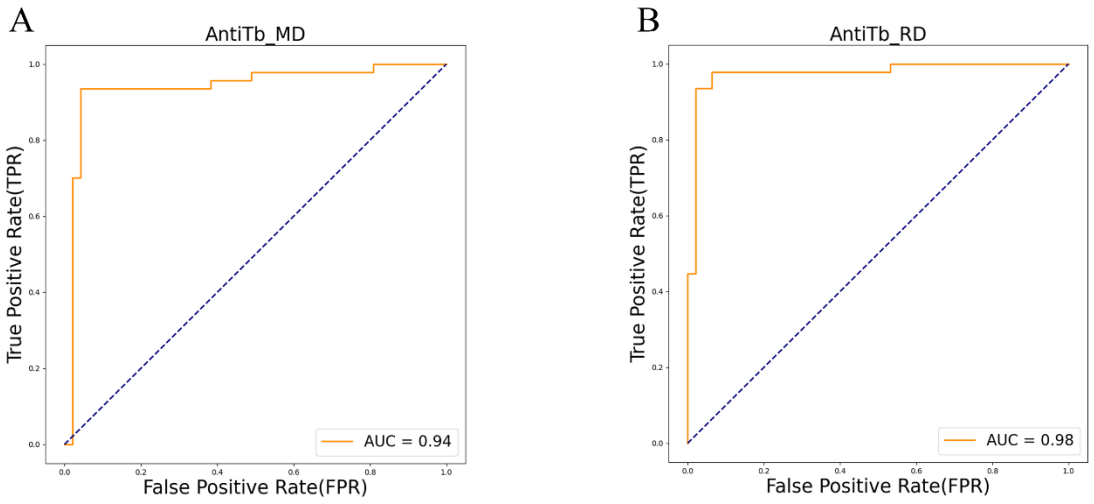


Fig. 9. ROC Analysis of the AntiTb_MD Independent dataset (A), and ROC Analysis of the AntiTb_RD Independent dataset (B).

IV. Conclusion

In this study, we propose a novel anti-tuberculosis peptide predictor Hyb_SEnc based on the hybrid eigenvector and stacked ensemble learning. First, we select five different machine learning algorithms and seven different feature encoding methods, and then, according to the performance of the baseline model obtained by training the five machine learning algorithms on the seven feature encoding methods, two machine learning algorithms with better performance were selected, namely, RF and ERT. Second, the feature vectors obtained from these seven feature encoding methods are used to train two machine learning algorithms, RF and ERT, and one of the worst performing feature encoding methods is removed from the AntiTb_MD and AntiTb_RD datasets. Then, the feature vectors obtained by the remaining five feature encoding methods are fused to construct a hybrid feature vector as Hybrid_F. According to the comparison experiment, the hybrid feature vector Hybrid_F shows a better classification effect. Finally, the hybrid feature vector Hybrid_F is used as an input, the two machine learning algorithms of RF and ERT are used as the first-level learner, and LR is used as the second-level learner to build the stacked integrated model Hyb_SEnc. Furthermore, fine-grained feature selection is performed when building an ensemble model. First, three feature selection methods, RF-RFE, DT-RFE, and ERT-RFE, were selected for feature selection, and the selected feature vectors were used for the stacking ensemble model training to obtain each Hyb_SEnc. According to the performance effect of each Hyb_SEnc on the independent test set of the AntiTb_MD dataset and the AntiTb_RD dataset, the DT-RFE feature selection method was used to build the optimal stacked ensemble model Hyb_SEnc. Finally, by comparing the existing anti-tuberculosis peptide predictors, it can be concluded that the integrated predictor Hyb_SEnc obtained in this study has the best overall performance on the independent test sets of the AntiTb_MD dataset and AntiTb_RD dataset. The prediction accuracy of Hyb_SEnc reached 94.68% and 95.74% on the independent test sets of AntiTb_MD and AntiTb_RD, respectively. Hyb_SEnc is currently the best performing anti-tuberculosis peptide predictor. In addition to the prediction of anti-tuberculosis peptides, the ideas and framework proposed in this study can be further applied to the prediction of other peptide sequences. In addition, to make Hyb_SEnc more widely used, we also created a user-friendly web server(http://www.bioai-lab.com/Hyb_SEnc) to promote the development of tuberculosis research. Hyb_SEnc is expected to be a valuable predictive tool for high-performance, high-quality identification of anti-tuberculosis peptides.

Competing Interests Statement

All authors declare that they have no conflicts of interest.

Funding

The work is supported by the National Natural Science Foundation of China (Nos. 62101100, 62262015) and Science and Technology special fund of Hainan Province (ZDYF2024GXJS01).

References

1. Taylor GM, Goyal M, Legge AJ, Shaw RJ, Young D: **Genotypic analysis of Mycobacterium tuberculosis from medieval human remains**. *Microbiology* 1999, **145**(4):899-904.
2. Chakaya J, Petersen E, Nantanda R, Mungai BN, Migliori GB, Amanullah F, Lungu P, Ntoumi F, Kumarasamy N, Maeurer M *et al*: **The WHO Global Tuberculosis 2021 Report - not so good news and turning the tide back to End TB**. *Int J Infect Dis* 2022, **124 Suppl 1**:S26-S29.

3. Cambau E, Drancourt M: **Steps towards the discovery of Mycobacterium tuberculosis by Robert Koch, 1882**. *Clin Microbiol Infect* 2014, **20**(3):196-201.
4. Zumla A, George A, Sharma V, Herbert RH, Baroness Masham of I, Oxley A, Oliver M: **The WHO 2014 global tuberculosis report--further to go**. *Lancet Glob Health* 2015, **3**(1):e10-12.
5. Iseman MD: **Tuberculosis therapy: past, present and future**. *Eur Respir J Suppl* 2002, **36**:87s-94s.
6. Joshi JM: **Tuberculosis chemotherapy in the 21 century: Back to the basics**. *Lung India* 2011, **28**(3):193-200.
7. Henninot A, Collins JC, Nuss JM: **The Current State of Peptide Drug Discovery: Back to the Future?** *J Med Chem* 2018, **61**(4):1382-1414.
8. Cao C, Wang J, Kwok D, Cui F, Zhang Z, Zhao D, Li MJ, Zou Q: **webTWAS: a resource for disease candidate susceptibility genes identified by transcriptome-wide association study**. *Nucleic acids research* 2022, **50**(D1):D1123-D1130.
9. Liu R, Zhang Z, Fu X, Yan S, Cui F: **AIPPT: Predicts anti-inflammatory peptides using the most characteristic subset of bases and sequences by stacking ensemble learning strategies**. In: *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM): 5-8 Dec. 2023 2023*; 2023: 23-29.
10. Guo X, Tiwari P, Zou Q, Ding Y: **Subspace projection-based weighted echo state networks for predicting therapeutic peptides**. *Knowledge-Based Systems* 2023, **263**:110307.
11. Khusro A, Aarti C, Barbabosa-Pliego A, Salem AZM: **Neoteric advancement in TB drugs and an overview on the anti-tubercular role of peptides through computational approaches**. *Microb Pathog* 2018, **114**:80-89.
12. Chen L, Yu L, Gao L: **Potent antibiotic design via guided search from antibacterial activity evaluations**. *Bioinformatics* 2023, **39**(2).
13. Khusro A, Aarti C, Agastian P: **Anti-tubercular peptides: A quest of future therapeutic weapon to combat tuberculosis**. *Asian Pac J Trop Med* 2016, **9**(11):1023-1034.
14. Yu L, Wang M, Yang Y, Xu F, Zhang X, Xie F, Gao L, Li X: **Predicting therapeutic drugs for hepatocellular carcinoma based on tissue-specific pathways**. *PLoS Comput Biol* 2021, **17**(2):e1008696.
15. Usmani SS, Bhalla S, Raghava GPS: **Prediction of Antitubercular Peptides From Sequence Information Using Ensemble Classifier and Hybrid Features**. *Front Pharmacol* 2018, **9**:954.
16. Ao C, Jiao S, Wang Y, Yu L, Zou Q: **Biological Sequence Classification: A Review on Data and General Methods**. *Research* 2022, **2022**:0011.
17. Cui F, Zhang Z, Cao C, Zou Q, Chen D, Su X: **Protein–DNA/RNA interactions: Machine intelligence tools and approaches in the era of artificial intelligence and big data**. *Proteomics* 2022, **22**(8):2100197.
18. Khatun S, Hasan M, Kurata H: **Efficient computational model for identification of antitubercular peptides by integrating amino acid patterns and properties**. *FEBS Lett* 2019, **593**(21):3029-3039.
19. Manavalan B, Basith S, Shin TH, Wei L, Lee G: **AtbPpred: A Robust Sequence-Based Prediction of Anti-Tubercular Peptides Using Extremely Randomized Trees**. *Comput Struct Biotechnol J* 2019, **17**:972-981.
20. Jain P, Tiwari AK, Som T: **Enhanced prediction of anti-tubercular peptides from sequence information using divergence measure-based intuitionistic fuzzy-rough feature selection**. *Soft Computing* 2020, **25**(4):3065-3086.
21. Cui F, Zhang Z, Zou Q: **Sequence representation approaches for sequence-based protein prediction tasks that use deep learning**. *Briefings in Functional Genomics* 2021, **20**(1):61-73.
22. Akbar S, Rahman AU, Hayat M, Sohail M: **cACP: Classifying anticancer peptides using discriminative intelligent model via Chou's 5-step rules and general pseudo components**. *Chemometrics and Intelligent Laboratory Systems* 2020, **196**.
23. Akbar S, Hayat M, Tahir M, Chong KT: **cACP-2LFS: Classification of Anticancer Peptides Using Sequential Discriminative Model of KSAAP and Two-Level Feature Selection Approach**. *IEEE Access* 2020, **8**:131939-131948.
24. Geetha R, Sivasubramanian S, Kaliappan M, Vimal S, Annamalai S: **Cervical Cancer Identification with Synthetic Minority Oversampling Technique and PCA Analysis using Random Forest Classifier**. *J Med Syst* 2019, **43**(9):286.

25. Akbar S, Hayat M, Kabir M, Iqbal M: **iAFP-gap-SMOTE: An Efficient Feature Extraction Scheme Gapped Dipeptide Composition is Coupled with an Oversampling Technique for Identification of Antifreeze Proteins**. *Letters in Organic Chemistry* 2019, **16**(4):294-302.
26. Ahmad J, Javed F, Hayat M: **Intelligent computational model for classification of sub-Golgi protein using oversampling and fisher feature selection methods**. *Artif Intell Med* 2017, **78**:14-22.
27. Akbar S, Ahmad A, Hayat M, Rehman AU, Khan S, Ali F: **iAtbP-Hyb-EnC: Prediction of antitubercular peptides via heterogeneous feature representation and genetic algorithm based ensemble learning model**. *Comput Biol Med* 2021, **137**:104778.
28. Jiao S, Zou Q, Guo H, Shi L: **iTTCA-RF: a random forest predictor for tumor T cell antigens**. *J Transl Med* 2021, **19**(1):449.
29. Wei L, Ye X, Xue Y, Sakurai T, Wei L: **ATSE: a peptide toxicity predictor by exploiting structural and evolutionary information based on graph neural network and attention mechanism**. *Briefings in Bioinformatics* 2021, **22**(5):bbab041.
30. Zhang X, Wei L, Ye X, Zhang K, Teng S, Li Z, Jin J, Kim MJ, Sakurai T, Cui L: **SiameseCPP: a sequence-based Siamese network to predict cell-penetrating peptides by contrastive learning**. *Briefings in Bioinformatics* 2023, **24**(1):bbac545.
31. Zhou Z, Xiao C, Yin J, She J, Duan H, Liu C, Fu X, Cui F, Qi Q, Zhang Z: **PSAC-6mA: 6mA site identifier using self-attention capsule network based on sequence-positioning**. *Computers in Biology and Medicine* 2024, **171**:108129.
32. Lv Z, Li M, Wang Y, Zou Q: **Editorial: Machine learning for biological sequence analysis**. *Frontiers in Genetics* 2023, **14**.
33. Yuan H, Liu Y, Du J, Zhou Z: **Prediction of anti-breast cancer compound activity based on Gradient Boosting Decision Tree ensemble learning**. In: *Proceedings of the 2nd International Symposium on Artificial Intelligence for Medicine Sciences.* 2021: 509-513.
34. Liu J, Song A, Zhu Z, Cen F: **Optimization and prediction of ERα activity screening of anti-breast cancer candidate based on machine learning algorithm**. In: *International Conference on Electronic Information Engineering and Computer Communication (EIECC 2021),* vol. 12172, 2022.
35. Chen Q, Meng Z, Liu X, Jin Q, Su R: **Decision Variants for the Automatic Determination of Optimal Feature Subset in RF-RFE**. *Genes (Basel)* 2018, **9**(6).
36. Lian W, Nie G, Jia B, Shi D, Fan Q, Liang Y, Kaur M: **An Intrusion Detection Method Based on Decision Tree-Recursive Feature Elimination in Ensemble Learning**. *Mathematical Problems in Engineering* 2020, **2020**:1-15.
37. Khan A, Uddin J, Ali F, Ahmad A, Alghushairy O, Banjar A, Daud A: **Prediction of antifreeze proteins using machine learning**. *Sci Rep* 2022, **12**(1):20672.
38. Deng H, Lou C, Wu Z, Li W, Liu G, Tang Y: **Prediction of anti-inflammatory peptides by a sequence-based stacking ensemble model named AIPStack**. *iScience* 2022, **25**(9):104967.
39. Fu X, Yuan Y, Qiu H, Suo H, Song Y, Li A, Zhang Y, Xiao C, Li Y, Dou L *et al*: **AGF-PPIS: A protein–protein interaction site predictor based on an attention mechanism and graph convolutional networks**. *Methods* 2024, **222**:142-151.
40. Duan H, Zhang Y, Qiu H, Fu X, Liu C, Zang X, Xu A, Wu Z, Li X, Zhang Q *et al*: **Machine learning-based prediction model for distant metastasis of breast cancer**. *Computers in Biology and Medicine* 2024, **169**:107943.
41. Jin Q, Cui H, Sun C, Meng Z, Su R: **Cascade knowledge diffusion network for skin lesion diagnosis and segmentation**. *Applied Soft Computing* 2021, **99**:106881.
42. Usmani SS, Kumar R, Kumar V, Singh S, Raghava GPS: **AntiTbPdb: a knowledgebase of anti-tubercular peptides**. *Database (Oxford)* 2018, **2018**.
43. Gogoladze G, Grigolava M, Vishnepolsky B, Chubinidze M, Duroux P, Lefranc MP, Pirtskhalava M: **DBAASP: database of antimicrobial activity and structure of peptides**. *FEMS Microbiol Lett* 2014, **357**(1):63-68.
44. Banjar A: **The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000**. *Nucleic Acids Research* 2000, **28**(1):45-48.
45. Li HL, Pang YH, Liu B: **BioSeq-BLM: a platform for analyzing DNA, RNA and protein sequences based on biological language models**. *Nucleic Acids Res* 2021, **49**(22):e129.

46. Chen Z, Zhao P, Li F, Leier A, Marquez-Lago TT, Wang Y, Webb GI, Smith AI, Daly RJ, Chou KC *et al*: **iFeature: a Python package and web server for features extraction and selection from protein and peptide sequences**. *Bioinformatics* 2018, **34**(14):2499-2502.
47. Chen Z, Zhao P, Li C, Li F, Xiang D, Chen YZ, Akutsu T, Daly RJ, Webb GI, Zhao Q *et al*: **iLearnPlus: a comprehensive and automated machine-learning platform for nucleic acid and protein sequence analysis, prediction and visualization**. *Nucleic Acids Res* 2021, **49**(10):e60.
48. Wei L, Ye X, Sakurai T, Mu Z, Wei L: **ToxIBTL: prediction of peptide toxicity based on information bottleneck and transfer learning**. *Bioinformatics* 2022, **38**(6):1514-1524.
49. Zhu H, Ao CY, Ding YJ, Hao HX, Yu L: **Identification of D Modification Sites Using a Random Forest Model Based on Nucleotide Chemical Properties**. *International Journal of Molecular Sciences* 2022, **23**(6).
50. Bhat PC, Prosper HB, Sekmen S, Stewart C: **Optimizing event selection with the random grid search**. *Computer Physics Communications* 2018, **228**:245-257.
51. Yan C, Zhu Y, Chen M, Yang K, Cui F, Zou Q, Zhang Z: **Integration tools for scRNA-seq data and spatial transcriptomics sequencing data**. *Briefings in Functional Genomics* 2024:elae002.
52. Alexander RA, Govern DM: **A New and Simpler Approximation for ANOVA Under Variance Heterogeneity**. *Journal of Educational Statistics* 1994, **19**(2):91-101.
53. He S, Guo F, Zou Q, HuiDing: **MRMD2.0: A Python Tool for Machine Learning with Feature Ranking and Reduction**. *Current Bioinformatics* 2021, **15**(10):1213-1221.
54. Zhang R, Nie F, Li X, Wei X: **Feature selection with multi-view data: A survey**. *Information Fusion* 2019, **50**:158-167.
55. Berrar D: **Cross-Validation**. In: *Encyclopedia of Bioinformatics and Computational Biology.* edn.; 2019: 542-545.
56. Zhang Z, Cui F, Su W, Dou L, Xu A, Cao C, Zou Q: **webSCST: an interactive web application for single-cell RNA-sequencing data and spatial transcriptomic data integration**. *Bioinformatics* 2022, **38**(13):3488-3489.
57. Zhang C, Ma，Yunqian: **Ensemble Machine Learning: Methods and Applications**. Boston, MA: Springer US; 2012.
58. Sagi O, Rokach L: **Ensemble learning: A survey**. *WIREs Data Mining and Knowledge Discovery* 2018, **8**(4).
59. Chatzimparmpas A, Martins RM, Kucher K, Kerren A: **Empirical Study: Visual Analytics for Comparing Stacking to Blending Ensemble Learning**. In: *2021 23rd International Conference on Control Systems and Computer Science (CSCS)*. 2021: 1-8.
60. Wu T, Zhang W, Jiao X, Guo W, Alhaj Hamoud Y: **Evaluation of stacking and blending ensemble learning methods for estimating daily reference evapotranspiration**. *Computers and Electronics in Agriculture*, vol. 184, pp. 106039, 2021.
61. Fushiki T: **Estimation of prediction error by using K-fold cross-validation**. *Statistics and Computing* 2009, **21**(2):137-146.
62. Yu L, Zheng YJ, Gao L: **MiRNA-disease association prediction based on meta-paths**. *Briefings in Bioinformatics* 2022, **23**(2).
63. Wang Y, Zhai Y, Ding Y, Zou Q: **SBSM-Pro: Support Bio-sequence Machine for Proteins**. In.; 2023: arXiv:2308.10275.
64. Luque A, Carrasco A, Martín A, de las Heras A: **The impact of class imbalance in classification performance metrics based on the binary confusion matrix**. *Pattern Recognition* 2019, **91**:216-231.
65. Huang J, Ling，C.X.: **Using AUC and accuracy in evaluating learning algorithms**. *IEEE Transactions on Knowledge and Data Engineering* 2005, **17**(3):299-310.

Xiuhao Fu is currently working toward the Master degree in electronic information, Hainan University, Haikou, China. His research interests include bioinformatics, machine learning and AI.

Hao Duan is currently working toward the Master degree in electronic information, Hainan University, Haikou, China. His research interests include bioinformatics, graph neural network and single cell sequencing.

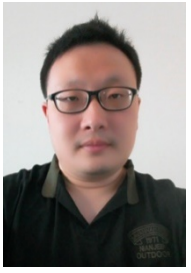
Xiaofeng Zang is currently working toward the Bachelor degree in software engineering, Hainan University, Haikou, China. She is research interests include big data and AI.

Chunling Liu is currently working toward the Bachelor degree in software engineering, Hainan University, Haikou, China. She is research interests include big data and machine learning.

Xingfeng Li is currently an associate professor in the School of Computer Science and Technology, Hainan University. He received his PhD degree in computer science from Japan Advanced Institute of Science and Technology in 2019. His research is in the areas of machine learning, affective computing and big health.

Qingchen Zhang is currently a professor in the School of Computer Science and Technology, Hainan University. He received his PhD degree in software engineering from Dalian University of Technology in 2015. He worked as postdoctoral researcher and assistant professor in St. Francis Xavier University, Canada, from 2016 to 2021.His research is in the areas of machine learning, medical big data.

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



Quan Zou (M'13-SM'17) is currently a Professor in the Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu, China. He received the BSc, MSc and the PhD degrees in computer science from Harbin Institute of Technology, China, in 2004, 2007 and 2009, respectively. He worked in Xiamen University and Tianjin University from 2009 to 2018 as an Assistant Professor, Associate Professor and Professor. His research is in the areas of bioinformatics, machine learning and parallel computing.

Several related works have been published by Science, Briefings in Bioinformatics, Bioinformatics, IEEE/ACM Transactions on Computational Biology and Bioinformatics, etc. Google scholar showed that his more than 100 papers have been cited more than 5000 times.

Dr. Zou is the editor-in-chief of Current Bioinformatics, associate editor of IEEE Access, and the editor board member of Computers in Biology and Medicine, Genes, Scientific Reports, etc. He was selected as one of the Clarivate Analytics Highly Cited Researchers in 2018 and 2022. He is also a reviewer for many impacted journals and NSFC (National Natural Science Foundation of China).

Feifei Cui is currently an associate professor in the School of Computer Science and Technology, Hainan University. She received the M.S. degree in computer application technology from Shandong University, Jinan, China, in 2012 and the Ph.D. degree in bioinformatics from the University of Tokyo, Japan in 2020. She worked as a postdoctoral researcher in the University of Electronic Science and Technology of China. Her research interests include bioinformatics, deep learning and biological data mining.

Dr. Cui is a winner of the Japanese Government (Monbukagakusho: MEXT) Scholarship Program from 2016 to 2019, and the Postdoctoral International Exchange Program from 2020 until 2022.