Frontiers in Bioscience Landmark

Front. Biosci. (Landmark Ed) **2022**; 27(1): 012
http://doi.org/10.31083/j.fbl2701012

Original Research

ET-MSF: a model stacking framework to identify electron transport proteins

Yizheng Wang[1,2], Qingfeng Pan[3,*], Xiaobin Liu[4,*], Yijie Ding[2,*]

[1]Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, 610054 Chengdu, Sichuan, China
[2]Yangtze Delta Region Institute (Quzhou), University of Electronic Science and Technology of China, 324000 Quzhou, Zhejiang, China
[3]Department of Oncology Radiology, Beidahuang Industry Group General Hospital, 150000 Harbin, Heilongjiang, China
[4]Department of Nephrology, The Affiliated Wuxi People's Hospital of Nanjing Medical University, 214023 Wuxi, Jiangsu, China
*Correspondence: qingfengpan1074@163.com (Qingfeng Pan); viplxb163@163.com (Xiaobin Liu); wuxi_dyj@163.com (Yijie Ding)
Academic Editor: Graham Pawelec
Submitted: 18 October 2021 Revised: 26 November 2021 Accepted: 9 December 2021 Published: 11 January 2022

Abstract

Introduction: The electron transport chain is closely related to cellular respiration and has been implicated in various human diseases. However, the traditional "wet" experimental method is time consuming. Therefore, it is key to identify electron transport proteins by computational methods. Many approaches have been proposed, but performance of them still has room for further improvement. **Methodological issues**: In our study, we propose a model stacking framework, which combines multiple base models. The protein features are extracted via PsePSSM from protein sequences. Features are fed into the base model including support vector machines (SVM), random forest (RF), XGBoost, etc. The results of base model are entered into logistic regression model for final process. **Results**: On the independent dataset, the accuracy and Matthew's correlation coefficient (MCC) of proposed method are 95.70% and 0.8756, respectively. Furthermore, we show that the model stacking framework outperforms single machine learning classifiers statistically. **Conclusion**: Our models are better than most known strategies for identifying electron transport proteins. Our model can be used to more precisely identify electron transport proteins.

Keywords: Electron transport chain; Ensemble learning; Model stacking; Logistic regression; Transport protein

1. Introduction

Protein is a vital component of all human cells and tissues, and it is intimately linked to life and many forms of biological activity, such as cellular respiration. Cellular respiration is the process by which organic matter passes through a series of oxidative breakdowns inside cells to form inorganic or small molecules of organic matter, releasing energy and producing Adenosine triphosphate (ATP), which is the most direct source of energy for most cellular reactions [1]. In this process, the electron transport chain is critical for storing and transferring electrons. Five protein complexes make up the electron transport chain and are named complex I, II, III, IV, and V. Electron transport proteins are made up of many electron carriers and serve a variety of molecular functions [2–4]. In studies, electron transporter abnormalities have been found to be associated with diseases such as idiopathic diabetes [5–8], Parkinson's disease [4], and Alzheimer's disease [9–12]. Therefore, the identification of electron transporter proteins is helpful in exploring the causes of human diseases and may help prevent and treat human diseases. Due to the high time and cost of identifying protein functions by traditional experimental techniques, computational approaches must be developed. The construction of meaningful feature sets and selection of appropriate classification algorithms are considered to be the two most important steps in protein classification. When constructing feature sets, some studies had taken advantage of the biochemical properties of proteins. Le *et al.* [13] extracted protein characteristics through biochemical characteristics, which improving the accuracy of identification of electron transporters. Khatun *et al.* [14] developed a model for predicting anti-inflammatory peptides using computational methods by using binary as the characteristic representation of proteins. Others employed position specific scoring matrix (PSSM) to extract evolutionary information of protein. In SulCysSite [15], binary, PSSM profiles, and pCKSAAP are combined as feature information to predict protein S-sulfenylation sites. Hasan *et al.* [16] used PSSM profiles to preserve the evolution information in proteins and developed pbPUP, a model for identifying protein pupylation sites, which had a good performance. Le *et al.* [17] proposed ET-CNN, which fed PSSM profiles into the convolutional neural networks (CNN) for electron transport protein classification. PSSM profiles and amino acid composition (AAC) were utilized by Chen *et al.* [18] to extract protein features, which were fed into radial basis function networks. In Mishra's study [19], features including amino acid composition, biochemical properties, and PSSM profiles were fed into the support vector machines (SVM) to predict transporters including electron transporters.

ET-GRU is a model proposed by Le *et al.* [20] to identify electron transport proteins. CNN is used to extract features from PSSM matrix before using deep-gated recurrent unit (GRU) for classification. In addition, traditional machine learning algorithms have been widely used. Gromiha *et al.* [21] and Ru *et al.* [22] used machine learning methods, including support vector machine, logistic regression, decision tree, random forest, and naive Bayes, to perform functional recognition of electron transport proteins. Single machine learning classification has both disadvantages and advantages. For example, the random forest has good performance on the unbalanced dataset but has high feature requirements. When XGBoost is used, excellent model performance can be achieved by adjusting a large number of complex parameters, which is totally difficult. Ensemble learning can flexibly combine various classifiers, train different models as base classifiers, and then combine them with ensemble strategies for final prediction. Common integration strategies include stacking, majority voting, bagging, boosting.

In our study, a model of stacking framework (MSF), which combines multiple base models is proposed. We represent protein features using the PSSM matrix produced by PSI-BLAST. Then, Pseudo-PSSM (PsePSSM) is used to extract evolutionary information. These features are fed into the base model. These base models are different classification algorithms, such as random forest, XGBoost, k-nearest neighbor (KNN), and SVM. These models are loosely coupled. After combining the results of each classifier, logistic regression makes the final prediction. Finally, the experimental results prove that the model of stacking framework (MSF), which is built by SVM, XGBoost, and KNN, has the best effect. We compare MSF with the single classifier and majority voting on the same independent dataset and perform a *t* test, the MSF is significantly superior to other single classifiers. In addition, MSF also perform better than most existing methods for identifying electron transport proteins.

2. Materials and methods

2.1 Datasets

In our study, we utilize the benchmark dataset released by Le *et al.* [20], which contains 1324 electron transport proteins and 4569 general transport proteins. The data were initially taken from a previous study [17], and data from UniProt release-2018_05 (on 23-May-2018) [23] and Gene Ontology (GO) release-2018-05-01 [24] were also collected. Then, in order to avoid model overfitting, the data were removed the redundant sequences with similarities of more than 30%. To solve this binary classification problem, the dataset is randomly divided into cross-validation dataset and independent dataset in a ratio of 0.85:0.15. Table 1 shows the details of the datasets.

Table 1. Statistics of all retrieved electron transport proteins and general transport proteins.

	Original	CV	IND
Electron transport	1324	1125	199
General transport	4569	3884	685

2.2 Feature extraction

Using appropriate methods to extract protein characteristics is an important step to complete the task of classification. PSSM, which retains the evolutionary information of proteins in a matrix of L rows and 20 columns, is employed as the feature extraction approach in this work. PSSM was first introduced by Jones [25], which is a commonly used method in the field of bioinformatics. It has been used in a number of bioinformatics studies with positive results [26–37]. The PSSM profiles is derived from multiple sequence alignment and contains the evolutionary information of each residue in the protein sequence. Electron transport proteins belong to a class of proteins with a specific function. Compared with other protein families, the key evolutionary information of proteins can be captured by using PSSM matrix and related feature extraction method, and then the classifier can be used to effectively identify electron transport proteins.

FASTA sequences are searched against the Uniprot database to compile position specific scoring matrices (PSSMs) for two iterations using Position Specific Iterative BLAST (PSI-BLAST [38]). The options for using BLAST+ [39] are as follows:

-num_iterations 2 *-db uniprot*

In our study, Pseudo-PSSM (PsePSSM) is employed to retain information in PSSM, and its basic idea is to consider the pseudo-amino-acid composition in PSSM [40]. This operation aims to get vectors that satisfy the algorithms and involves two steps.

The first step is the process of standardizing the PSSM. The formula is shown below:

$$p'_{i,j} = \frac{p_{i,j} - \frac{1}{20}\sum_{m=1}^{20} p_{i,m}}{\sqrt{\frac{1}{20}\sum_{n=1}^{20}\left(p_{i,n} - \frac{1}{20}\sum_{m=1}^{20} p_{i,m}\right)^2}} \quad (1)$$

The second step is to use the standardized matrix to generate Pseudo-PSSM (PsePSSM).

$$F_{PsePssm} = \begin{cases} \frac{1}{N}\sum_{i=1}^{N} p'_{i,j}; j = 1, \cdots, 20 \\ \frac{1}{N-lag}\sum_{i=1}^{N-lag}\left(p'_{i,j} - p'_{i+lag,j}\right)^2; \\ lag = 1, \cdots, 8, j = 1, \cdots, 20 \end{cases} \quad (2)$$

where lag denotes the distance between one residue and its

neighbors.

Finally, the standardized PSSM matrix is transformed into a $20 + 20 \times 8 = 180$ dimensions vector containing protein characteristics.

2.3 Classification algorithms

2.3.1 Support Vector Machine

Support Vector Machine (SVM) [41] is a supervised learning algorithm for classification. As a generalized linear classifier, the purpose of the support vector machine is to find the maximum boundary hyperplane as the decision boundary, so as to complete the classification task. SVM is widely used in classification, regression, and other tasks [42–52]. Since the dataset is linearly non-separable, the SVM with Gaussian kernel function (RBF) is employed as the fitting algorithm. C is a crucial parameter in support vector machines. The penalty factor, or error tolerance, is denoted by the C. The penalty SVM receives in the case of classification error is positively connected with the C. In addition, gamma (γ) is also an important parameter that affects the classification effect of SVM, which implicitly determines the distribution of data after mapping to the new feature space. The larger γ is, the fewer support vectors are. To discover the best combination of C and γ so that the model has a positive effect, the grid search method is used.

2.3.2 Random forest

Random forest (RF) [53] is an ensemble machine learning technique that trains and predicts samples using several trees, which has found many successful applications in the field of bioinformatics [54–62]. RF consists of many decision trees that are not related to each other. When a sample is input into RF for the classification task, the sample will be judged by each decision tree in the forest and the classification result will be obtained. The final output of the RF is a combination of the results of each decision tree in the forest. Plenty of parameters affect the RF, among which the most influential ones include the number of subtrees to be built, and the maximum growth depth of trees. In the same way, grid search is used to find suitable values for these parameters.

2.3.3 K-Nearest Neighbor

K-Nearest Neighbor (KNN) determines the categories of input samples based on the most similar samples in the feature space, which is one of the most commonly used machine learning algorithms. K value is the only parameter that the KNN algorithm needs to specify, so it has a significant effect on the result. In KNN algorithm, each sample can be represented by its nearest K neighboring values. The smaller the k value is, the less the approximate error of learning is, because a small k value can make the prediction result of the algorithm only be affected by the training instance that is close to the input instance. Again, we use grid search to find the appropriate K value.

2.3.4 Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is another advance machine learning model [63]. XGBoost has been successful in a variety of machine learning competitions [64] as well as in other fields [65–68]. It is a tool for large-scale parallel trees boosting. XGBoost has the advantages of higher accuracy and greater flexibility. The parameters that affect the XGBoost effect mainly include learning rate, minimum loss reduction required by leaf node splitting, maximum depth of the tree, and minimum weight of the leaf node. Again, use grid search to adjust these parameters.

2.3.5 Model stacking

Model stacking is a strategy of ensemble learning. The basic idea of ensemble learning is to combine multiple classifiers, and the errors encountered by one weak classifier are highly likely to be corrected by other weak classifiers. Combining multiple models can produce a model with better performance and stronger generalization ability. Typical integration strategies are bagging, boosting, stacking, and voting [69–80]. The use of bagging as an integration strategy is mainly to reduce the generalization error of models, which is achieved by combining multiple models. Bagging is implemented by using different bootstrap samples to train different models. When testing the sample input, the output of each model is voted to get the final result. Boosting's idea is to combine a series of averagely performing models using particular cost functions. Majority voting includes both soft and hard voting. Hard voting is to make statistics on the predicted result label of the base model and take the result with more occurrences as the final result, while soft voting uses the predicted probability of the base model instead of the predicted result label to complete the voting mechanism. In our study, the integration strategy we choose is stacking. In order to prevent model overfitting, we use a simple model, logistic regression, to make the final prediction. The workflow of this method is shown in Fig. 1.

2.4 Performance measures

The evaluation of the result is shown in four standard measurements, Accuracy (ACC), Matthew's correlation coefficient (MCC) [43,81–92], Sensitivity (SN), and Specificity (SP). Their formulas are as follows:

$$SN = \frac{TP}{FN + TP} \tag{3}$$

$$SP = \frac{TN}{FP + TN} \tag{4}$$

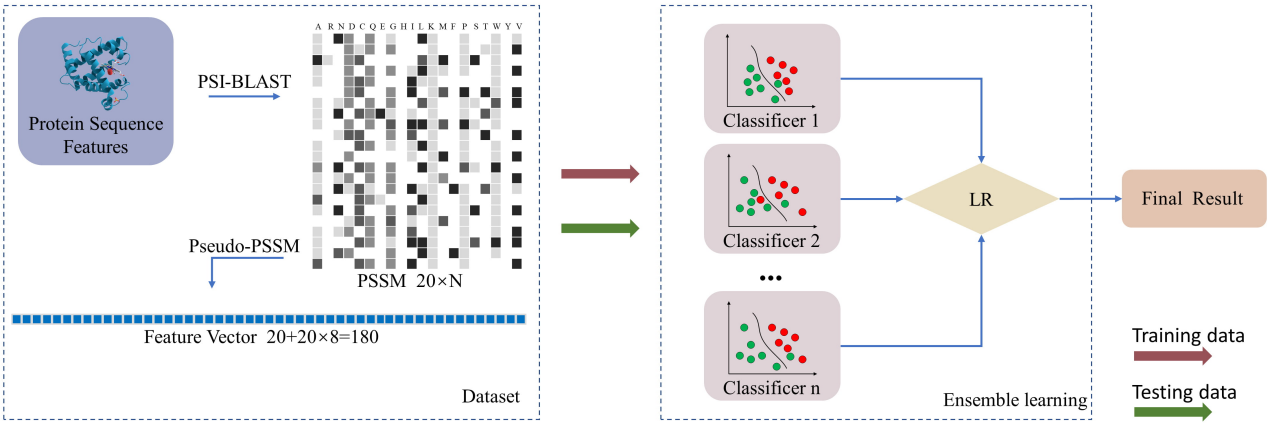

Fig. 1. The MSF workflow.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FN) \times (TP+FP) \times (TN+FP) \times (TN+FN)}} \quad (5)$$

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

where TP, TN, FN, and FP denote the number of true positive, true negative, false negative, and false positive, respectively.

3. Results and discussion

3.1 Performance comparisons with the single classifiers and using different base classifiers

We conduct two types of experiments to test model stacking frameworks. Firstly, predictors of four single classifiers are built by RF, XGBoost, KNN, and SVM, respectively. We compare their performance on the same independent dataset, and the results are shown in Table 2. Then, three classifiers using different combinations of base models are constructed: MSF (SVM, XGBoost), MSF (SVM, XGBoost, KNN), and MSF (SVM, XGBoost, KNN, RF). They employ SVM + XGBoost, SVM + XGBoost + KNN, SVM + XGBoost + KNN + RF as the base classifiers, respectively. Next, LR is employed as the last layer of the model and the results of these base classifiers are fed into the final prediction. The results are shown in Table 3. The ROC curve with AUC values and PR curve are shown in Fig. 2.

Table 2 shows that the predictor that uses SVM, which achieves the best performance of MCC = 0.8599, ACC = 95.14, SP = 96.66, and SN = 89.79%. The RF performs the worst results of MCC = 0.7558, ACC = 0.9186%, SP = 91.58, and SN = 93.20. The gaps of MCC, ACC, SP, and SN between the worst and best ones are 3.28%, 0.1041, – 3.41% and 5.08%, respectively. Fig. 2 also shows that SVM has the best performance and highest AUC values.

Table 2. The performances of single classifiers on the independent dataset.

	SN (%)	SP (%)	ACC (%)	MCC
RF	93.20	91.58	91.86	0.7558
XGBoost	91.23	93.97	93.44	0.8057
KNN	90.50	94.75	93.89	0.8203
SVM	89.79	96.66	95.14	0.8599

Table 3. The performances using different base classifiers on the independent dataset.

	SN (%)	SP (%)	ACC (%)	MCC
MSF (SVM, XGBoost)	91.44	95.98	95.02	0.8549
MSF (SVM, XGBoost, KNN)	91.71	96.82	95.70	0.8756
MSF (SVM, XGBoost, KNN, RF)	91.24	96.81	95.59	0.8725

Table 4. *T*-test analysis results of model stacking framework and single classifiers.

	p-value	log (0.05/*p*-value)
RF	3.8458×10^{-16}	14.12
XGBoost	6.1513×10^{-7}	4.91
KNN	9.6691×10^{-4}	1.71
SVM	7.7435×10^{-3}	0.81

Table 5. The performances using majority voting-based methods on the independent dataset.

	SN (%)	SP (%)	ACC (%)	MCC
Hard voting (SVM, XGBoost, KNN)	91.89	95.85	95.02	0.8546
Soft voting (SVM, XGBoost, KNN)	92.86	95.73	95.14	0.8576
MSF (SVM, XGBoost, KNN)	91.71	96.82	95.70	0.8756

Table 3 shows that the fourth predictor is ahead of the others on four measures. In all evaluative measurements, ensemble model is better than a single predictor. In Fig. 2,

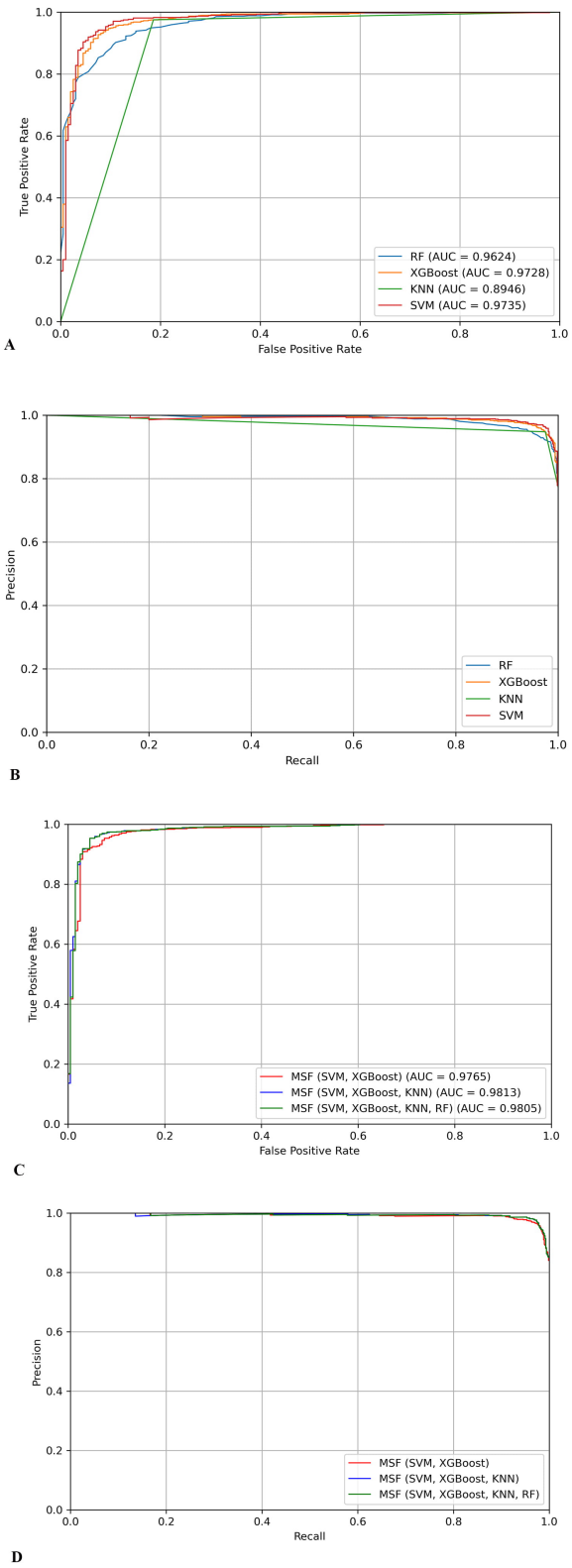

Fig. 2. The ROC and PR curves of different models. (A) The ROC curves of single classifiers. (B) The PR curves of single classifiers. (C) The ROC curves of using different base classifiers. (D) The PR curves of using different base classifiers.

comparing the AUC values of MSF and single classifiers, we also find that MSF has better performance. We believe that ensemble learning performs better than single classifiers because ensemble learning can combine different classifiers flexibly and make better use of the advantages of classifiers. In addition, different base classifiers may have different feature representations for the same data, resulting in the effect of mutual error correction, thus obtaining better performance. Compared to the first predictor, the second predictor obtain better performance by adding KNN which performs second-best among the four single classifiers. RF reduces the overall prediction performance of ensemble model. The results show that a weak base classifier may lead to the effect of model stacking.

3.2 T*-test analysis of model stacking framework and single classifiers*

To further demonstrate the statistical significance of MSF, we perform *t*-test analysis on MSF and single classifiers including RF, XGBoost, KNN, and SVM.

The *p*-value of the *t*-test are shown in Table 4. In addition, log (0.05/*p*-value) visually shows the difference between *p*-value and 0.05. The larger the value is, the more significant the difference is.

The results show that the *p*-values of all algorithms are less than 0.05, indicating that the effects of stacked model framework and single classifier are significantly different and statistically significant. In addition, RF is the algorithm with the most significant difference from the stack model.

3.3 Performance comparisons with majority voting-based methods

In order to further verify that model stacking is an appropriate integration strategy for protein classification problems, the majority voting-based method using SVM, XGBoost, and KNN is tested on the independent dataset. It contains two types of hard voting and soft voting. The results can be found in Table 5.

The gaps of MCC, ACC, SP, and SN between the Hard voting and Soft voting are 0.12%, –0.97%, –0.003%, and –0.12%, respectively. On the same independent dataset, MSF ranks first except for the SN measurements.

3.4 Stability comparisons with the single classifiers and using different base classifiers

When the model is applied, the results of the system may fluctuate due to changes in the data. In order to judge the severity of the model changes, we conduct stability tests. We conduct five times of 5-fold cross-validation to test the stability of the algorithm, and calculate the mean and standard deviation of four metrics including SN, SP, ACC and MCC, as shown in Table 6.

The results show that MSF achieves the best mean performance except that the SN value is worse than that of RF. In stability, MSF's four metrics rank 2,5,3, and 3 respec-

Table 6. The means and standard deviations for the 5 × 5-fold cross-validations.

	SN	SP	ACC	MCC
RF	92.96 ± 0.036046	91.48 ± 0.002203	91.70 ± 0.005477	0.7514 ± 0.017186
XGBoost	90.65 ± 0.016817	94.25 ± 0.004943	93.53 ± 0.004603	0.8090 ± 0.014010
KNN	89.06 ± 0.012900	95.22 ± 0.004504	93.91 ± 0.003070	0.8223 ± 0.009366
SVM	89.41 ± 0.008326	96.57 ± 0.003200	94.98 ± 0.001693	0.8554 ± 0.005167
MSF (SVM, XGBoost, KNN)	91.50 ± 0.010614	96.63 ± 0.008257	95.50 ± 0.004375	0.8697 ± 0.013990

Table 7. Performance comparisons to existing methods.

	SN (%)	SP (%)	ACC (%)	MCC
ET-CNN	80.3	94.4	92.3	0.71
ET-GRU	79.8	95.9	92.3	0.77
MSF (SVM, XGBoost, KNN)	91.7	96.8	95.7	0.88

tively among all classifiers.

3.5 Comparisons of existing methods

To further test our method's sophisticated and superior performance, we compare other existing works under the same data set. These models include ET-CNN [17] and ET-GRU [20].

Table 7 shows results of comparisons between our model and other approaches. It is easy to observe that MSF (SVM, XGBoost, KNN) ranks first with ACC = 95.7, MCC = 0.88, SN = 91.7%, and SP = 96.8%.

We believe that the main reason why ET-MSF performs better than ET-CNN and ET-GRU may be that our algorithm is more suitable for the task. Due to the characteristics of the neural network, only a large number of samples can make the network better fitting. However, the dataset of electron transport protein is relatively small, which limits the growth of neural network model size and leads to poor model effect. Therefore, using the ensemble strategy to combine machine learning classifiers can have a better effect.

3.6 Web server development

We provide a simple web server that can be freely accessible at http://82.156.89.65/ to allow readers to evaluate and use our approaches online. The online version of ET-MSF uses the Java language and the Spring Boot framework. ET-MSF can be used by biologists to identify electron transport proteins online. Biologists can obtain the model's prediction probability by simply entering the protein's amino acid sequence(s) in a standard FASTA file format. Biologists can also download our public datasets and models at https://github.com/Kinkou626/ET-MSF and run them on own computer.

4. Conclusions

In our study, a model of stacking framework is proposed to extract the features of protein sequences and identify electron transporter proteins by integrating multiple classifiers. Previously, the model of stacking framework has not been applied to electron transporter protein recognition tasks. We use an independent dataset to evaluate model. Model of stacking frameworks using different base model combinations are compared experimentally. Comprehensive experiments show that the model of stacking framework via SVM, XGBoost, and KNN is the best model, which achieve the ACC and MCC values of 95.70% and 0.8756 respectively. Compared with existing methods, MSF also achieves significant improvements in all measurements. Our method will be an effective bioinformatics tool. And it can also be used to recognize protein functions in other types of proteins.

Author contributions

YW did the experiments and wrote the manuscript. YW, QP, XL, and YD designed the method. YW, QP, XL, and YD revised the manuscript. All authors have read and approved the final manuscript.

Ethics approval and consent to participate

Not applicable.

Acknowledgment

Not applicable.

Funding

The work was supported by the National Natural Science Foundation of China (No. 61922020), the Sichuan Provincial Science Fund for Distinguished Young Scholars (2021JDJQ0025), and the Special Science Foundation of Quzhou (2021D004).

Conflict of interest

The authors declare no conflict of interest.

References

[1] Chance B, Williams GR. The respiratory chain and oxidative phosphorylation. Advances in Enzymology and Related Subjects of Biochemistry. 1956; 17: 65–134.

[2] Foyer CH, Harbinson J. Oxygen metabolism and the regulation of photosynthetic electron transport. In: Causes of photooxidative stress and amelioration of defense systems in plants (pp. 1–42). CRC Press: Boca Raton. 2019.

[3] Mrozek D, Malysiak B, Kozielski S. 'An optimal alignment of proteins energy characteristics with crisp and fuzzy similarity awards', 2007 IEEE International Fuzzy Systems Conference. London, UK. 2007.
[4] Hu Y, Qiu S, Cheng L. Integration of Multiple-Omics Data to Analyze the Population-Specific Differences for Coronary Artery Disease. Computational and Mathematical Methods in Medicine. 2021; 2021: 7036592.
[5] Ritov VB, Menshikova EV, Azuma K, Wood R, Toledo FGS, Goodpaster BH, *et al*. Deficiency of electron transport chain in human skeletal muscle mitochondria in type 2 diabetes mellitus and obesity. American Journal of Physiology-Endocrinology and Metabolism. 2010; 298: E49–E58.
[6] Zou Q, Qu K, Luo Y, Yin D, Ju Y, Tang H. Predicting Diabetes Mellitus with Machine Learning Techniques. Frontiers in Genetics. 2018; 9: 515
[7] Qu K, Zou Q, Shi H. Prediction of diabetic protein markers based on an ensemble method. Frontiers in Bioscience-Landmark. 2021; 26: 207–221.
[8] Parker WD, Boyson SJ, Parks JK. Abnormalities of the electron transport chain in idiopathic parkinson's disease. Annals of Neurology. 1989; 26: 719–723.
[9] Parker WD, Filley CM, Parks JK. Cytochrome oxidase deficiency in Alzheimer's disease. Neurology. 1990; 40: 1302–1303.
[10] Xu L, Liang G, Liao C, Chen G, Chang C. K-Skip-n-Gram-RF: A Random Forest Based Method for Alzheimer's Disease Protein Identification. Frontiers in Genetics. 2020; 10: 33.
[11] Hu Y, Sun J, Zhang Y, Zhang H, Gao S, Wang T, *et al*. Rs1990622 variant associates with Alzheimer's disease and regulates TMEM106B expression in human brain tissues. BMC Medicine. 2021; 19: 11.
[12] Hu Y, Zhang H, Liu B, Gao S, Wang T, Han Z, *et al*. Rs34331204 regulates TSPAN13 expression and contributes to Alzheimer's disease with sex differences. Brain. 2020; 143: e95–e95.
[13] Le N, Nguyen T, Ou Y. Identifying the molecular functions of electron transport proteins using radial basis function networks and biochemical properties. Journal of Molecular Graphics and Modelling. 2017; 73: 166–178.
[14] Khatun M, Hasan M, Kurata H. PreAIP: computational prediction of anti-inflammatory peptides by integrating multiple complementary features. Frontiers in Genetics. 2019; 10: 129.
[15] Hasan MM, Guo D, Kurata H. Computational identification of protein S-sulfenylation sites by incorporating the multiple sequence features information. Molecular BioSystems. 2017; 13: 2545–2550.
[16] Hasan MM, Zhou Y, Lu X, Li J, Song J, Zhang Z. Computational Identification of Protein Pupylation Sites by Using Profile-Based Composition of k-Spaced Amino Acid Pairs. PLoS ONE. 2015; 10: e0129635.
[17] Le N, Ho Q, Ou Y. Incorporating deep learning with convolutional neural networks and position specific scoring matrices for identifying electron transport proteins. Journal of Computational Chemistry. 2017; 38: 2000–2006.
[18] Chen S, Ou Y, Lee T, Gromiha MM. Prediction of transporter targets using efficient RBF networks with PSSM profiles and biochemical properties. Bioinformatics. 2011; 27: 2062–2067.
[19] Mishra NK, Chang J, Zhao PX. Prediction of membrane transport proteins and their substrate specificities using primary sequence information. PLoS ONE. 2014; 9: e100278.
[20] Le NQK, Yapp EKY, Yeh H. ET-GRU: using multi-layer gated recurrent units to identify electron transport proteins. BMC Bioinformatics. 2019; 20: 377.
[21] Gromiha MM, Yabuki Y. Functional discrimination of membrane proteins using machine learning techniques. BMC Bioinformatics. 2008; 9: 135.
[22] Ru X, Li L, Zou Q. Incorporating Distance-Based top-n-gram and Random Forest to Identify Electron Transport Proteins. Journal of Proteome Research. 2019; 18: 2931–2939.
[23] Bateman A, Martin MJ, O'Donovan C, Magrane M, Apweiler R, Alpi E, *et al*. UniProt: a hub for protein information. Nucleic Acids Research. 2014; 43: D204–D212.
[24] Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, *et al*. Gene Ontology: tool for the unification of biology. Nature Genetics. 2000; 25: 25–29.
[25] Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. Journal of Molecular Biology. 1999; 292: 195–202.
[26] Su C, Chen C, Ou Y. Protein disorder prediction by condensed PSSM considering propensity for order or disorder. BMC Bioinformatics. 2006; 7: 319.
[27] Le N-Q-K, Ou Y-Y. Prediction of FAD binding sites in electron transport proteins according to efficient radial basis function networks and significant amino acid pairs. BMC Bioinformatics. 2016; 17: 298.
[28] Li Z, Zhao Y, Pan G, Tang J, Guo F. A Novel Peptide Binding Prediction Approach for HLA-DR Molecule Based on Sequence and Structural Information. BioMed Research International. 2016; 2016: 3832176.
[29] Mrozek D, Malysiak-Mrozek B, Kozielski S. Alignment of Protein Structure Energy Patterns Represented as Sequences of Fuzzy Numbers. 2009 Annual Meeting of the North American Fuzzy Information Processing Society 2009; 35–40.
[30] Hong Z, Zeng X, Wei L, Liu X. Identifying enhancer–promoter interactions with neural network based on pre-trained DNA vectors and attention mechanism. Bioinformatics. 2019; 36:1037–1043.
[31] Zeng X, Liao Y, Liu Y, Zou Q. Prediction and Validation of Disease Genes Using HeteSim Scores. IEEE/ACM Transactions on Computational Biology and Bioinformatics. 2017; 14: 687–695.
[32] Zeng X, Lin W, Guo M, Zou Q. A comprehensive overview and evaluation of circular RNA detection tools. PLoS Computational Biology. 2017; 13: e1005420.
[33] Cai L, Wang L, Fu X, Xia C, Zeng X, Zou Q. ITP-Pred: an interpretable method for predicting, therapeutic peptides with fused features low-dimension representation. Briefings in Bioinformatics. 2021; 22: bbaa367.
[34] Cheng L, Qi C, Zhuang H, Fu T, Zhang X. GutMDisorder: a comprehensive database for dysbiosis of the gut microbiota in disorders and interventions. Nucleic Acids Research. 2020; 48: D554–D560.
[35] Cheng L, Hu Y, Sun J, Zhou M, Jiang Q. DincRNA: a comprehensive web-based bioinformatics toolkit for exploring disease associations and ncRNA function. Bioinformatics. 2018; 34: 1953–1956.
[36] Wu Y, Lu X, Shen B, Zeng Y. The Therapeutic Potential and Role of miRNA, lncRNA, and circRNA in Osteoarthritis. Current Gene Therapy. 2019; 19: 255–263.
[37] Yu L, Wang M, Yang Y, Xu F, Zhang X, Xie F, *et al*. Predicting therapeutic drugs for hepatocellular carcinoma based on tissue-specific pathways. PLOS Computational Biology. 2021; 17: e1008696.
[38] Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, *et al*. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Research. 1997; 25: 3389–3402.
[39] Camacho C, Coulouris G, Avagyan V, Ma N, Papadopoulos J, Bealer K, *et al*. BLAST+: architecture and applications. BMC Bioinformatics. 2009; 10: 421.
[40] Chou K, Shen H. MemType-2L: a web server for predicting membrane proteins and their types by incorporating evolution information through Pse-PSSM. Biochemical and Biophysical

Research Communications. 2007; 360: 339–345.
[41] Cortes C, Vapnik V. Support-vector networks. Machine Learning. 1995; 20: 273–297.
[42] Tao Z, Li Y, Teng Z, Zhao Y. A Method for Identifying Vesicle Transport Proteins Based on LibSVM and MRMD. Computational and Mathematical Methods in Medicine. 2020; 2020: 1–9.
[43] Jiang Q, Wang G, Jin S, Li Y, Wang Y. Predicting human microRNA-disease associations based on support vector machine. International Journal of Data Mining and Bioinformatics. 2013; 8: 282–293.
[44] Su R, Liu X, Jin Q, Liu X, Wei L. Identification of glioblastoma molecular subtype and prognosis based on deep MRI features. Knowledge-Based Systems. 2021; 232: 107490.
[45] Su R, Wu H, Xu B, Liu X, Wei L. Developing a Multi-Dose Computational Model for Drug-Induced Hepatotoxicity Prediction Based on Toxicogenomics Data. IEEE/ACM Transactions on Computational Biology and Bioinformatics. 2019; 16: 1231–1239.
[46] Liu J, Su R, Zhang J, Wei L. Classification and gene selection of triple-negative breast cancer subtype embedding gene connectivity matrix in deep neural network. Briefings in Bioinformatics. 2021. (in press)
[47] Cheng L, Yang H, Zhao H, Pei X, Shi H, Sun J, *et al*. MetSigDis: a manually curated resource for the metabolic signatures of diseases. Briefings in Bioinformatics. 2019; 20: 203–209.
[48] Lu X, Zhao S. Gene-based Therapeutic Tools in the Treatment of Cornea Disease. Current Gene Therapy. 2019; 19: 7–19.
[49] Tahir M, Idris A. MD-LBP: An Efficient Computational Model for Protein Subcellular Localization from HeLa Cell Lines Using SVM. Current Bioinformatics. 2020; 15: 204–211.
[50] Meng C, Guo F, Zou Q. CWLy-SVM: a support vector machine-based tool for identifying cell wall lytic enzymes. Computational Biology and Chemistry. 2020; 87: 107304.
[51] Kuo J, Chang C, Chen C, Liang H, Chang C, Chu Y. Sequence-based Structural B-cell Epitope Prediction by Using Two Layer SVM Model and Association Rule Features. Current Bioinformatics. 2020; 15: 246–252.
[52] Hasan MM, Basith S, Khatun MS, Lee G, Manavalan B, Kurata H. Meta-i6mA: an interspecies predictor for identifying DNA N6-methyladenine sites of plant genomes by exploiting informative features in an integrative machine-learning framework. Briefings in Bioinformatics. 2021; 22: bbaa202.
[53] Breiman L. Random Forests. Machine Learning. 2001; 45: 5–32.
[54] Qi Y. Random forest for bioinformatics. In: Ensemble machine learning (pp. 307–323). Springer: Berlin/Heidelberg, Germany. 2012.
[55] Wei L, Xing P, Shi G, Ji Z, Zou Q. Fast Prediction of Protein Methylation Sites Using a Sequence-Based Feature Selection Technique. IEEE/ACM Transactions on Computational Biology and Bioinformatics. 2019; 16: 1264–1273.
[56] Wei L, Su R, Wang B, Li X, Zou Q, Gao X. Integration of deep feature representations and handcrafted features to improve the prediction of N6-methyladenosine sites. Neurocomputing. 2019; 324: 3–9.
[57] Su R, Liu X, Wei L, Zou Q. Deep-Resp-Forest: a deep forest model to predict anti-cancer drug response. Methods. 2019; 166: 91–102.
[58] Cheng L, Han X, Zhu Z, Qi C, Wang P, Zhang X. Functional alterations caused by mutations reflect evolutionary trends of SARS-CoV-2. Briefings in Bioinformatics. 2021; 22: 1442–1450.
[59] Chen X, Shi W, Deng L. Prediction of Disease Comorbidity Using HeteSim Scores based on Multiple Heterogeneous Networks. Current Gene Therapy. 2019; 19: 232–241.
[60] Ao C, Yu L, Zou Q. RFhy-m2G: Identification of RNA N2-methylguanosine Modification Sites Based on Random Forest and Hybrid Features. Methods. 2021. (in press)
[61] Ao C, Yu L, Zou Q. Prediction of bio-sequence modifications and the associations with diseases. Briefings in Functional Genomics. 2021; 20: 1–18.
[62] Hasan MM, Alam MA, Shoombuatong W, Deng H, Manavalan B, Kurata H. NeuroPred-FRL: an interpretable prediction model for identifying neuropeptide using feature representation learning. Briefings in Bioinformatics. 2021. (in press)
[63] Chen T, Guestrin C: Xgboost. A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785–794). Association for Computing Machinery: New York, NY, United States. 2016.
[64] Tianqi C, Tong H. Higgs Boson Discovery with Boosted Trees. 2014 International Conference on High-Energy Physics and Machine Learning. Valencia, Spain, 2-9 July 2014.
[65] Torlay L, Perrone-Bertolotti M, Thomas E, Baciu M. Machine learning-XGBoost analysis of language networks to classify patients with epilepsy. Brain Informatics. 2017; 4: 159–169.
[66] Cai L, Ren X, Fu X, Peng L, Gao M, Zeng X. IEnhancer-XG: interpretable sequence-based enhancers and their strength predictor. Bioinformatics. 2021; 37: 1060–1067.
[67] Yu X, Zhou J, Zhao M, Yi C, Duan Q, Zhou W, *et al*. Exploiting XG Boost for Predicting Enhancer-promoter Interactions. Current Bioinformatics. 2020; 15: 1036–1045.
[68] Hasan MM, Schaduangrat N, Basith S, Lee G, Shoombuatong W, Manavalan B. HLPpred-Fuse: improved and robust prediction of hemolytic peptide and its activity by fusing multiple feature representation. Bioinformatics. 2020; 36: 3350–3356.
[69] Breiman L. Bagging predictors. Machine Learning. 1996; 24: 123–140.
[70] Małysiak-Mrozek B, Baron T, Mrozek D. Spark-IDPP: high-throughput and scalable prediction of intrinsically disordered protein regions with Spark clusters on the Cloud. Cluster Computing. 2019; 22: 487–508.
[71] Wei L, Xing P, Zeng J, Chen J, Su R, Guo F. Improved prediction of protein-protein interactions using novel negative samples, features, and an ensemble classifier. Artificial Intelligence in Medicine. 2017; 83: 67–74.
[72] Wei L, Wan S, Guo J, Wong KK. A novel hierarchical selective ensemble classifier with bioinformatics application. Artificial Intelligence in Medicine. 2017; 83: 82–90.
[73] Wei L, Tang J, Zou Q. Local-DPP: an improved DNA-binding protein prediction method by exploring local evolutionary information. Information Sciences. 2017; 384: 135–144.
[74] Wang Z, He W, Tang J, Guo F. Identification of Highest-Affinity Binding Sites of Yeast Transcription Factor Families. Journal of Chemical Information and Modeling. 2020; 60: 1876–1883.
[75] Ding Y, Tang J, Guo F. Identification of Protein-Ligand Binding Sites by Sequence Information and Ensemble Classifier. Journal of Chemical Information and Modeling. 2017; 57: 3149–3161.
[76] Fu X, Cai L, Zeng X, Zou Q. StackCPPred: a stacking and pairwise energy content-based prediction of cell-penetrating peptides and their uptake efficiency. Bioinformatics. 2020; 36: 3028–3034.
[77] Yu L, Xia M, An Q. A network embedding framework based on integrating multiplex network for drug combination prediction. Briefings in Bioinformatics. 2021. (in press)
[78] Ru X, Cao P, Li L, Zou Q. Selecting Essential MicroRNAs Using a Novel Voting Method. Molecular Therapy - Nucleic Acids. 2019; 18: 16–23.
[79] Zhu H, Du X, Yao Y. ConvsPPIS: Identifying Protein-protein Interaction Sites by an Ensemble Convolutional Neural Network with Feature Graph. Current Bioinformatics. 2020; 15: 368–

378.
[80] Sultana N, Sharma N, Sharma KP, Verma S. A Sequential Ensemble Model for Communicable Disease Forecasting. Current Bioinformatics. 2020; 15: 309–317.
[81] Xu Z, Luo M, Lin W, Xue G, Wang P, Jin X, *et al*. DLpTCR: an ensemble deep learning framework for predicting immunogenic peptide recognized by T cell receptor. Briefings in Bioinformatics. 2021; 22: bbab335
[82] Huang Y, Zhou D, Wang Y, Zhang X, Su M, Wang C, *et al*. Prediction of transcription factors binding events based on epigenetic modifications in different human cells. Epigenomics. 2020; 12: 1443–1456.
[83] Zhang L, Xiao X, Xu ZC. iPromoter-5mC: A Novel Fusion Decision Predictor for the Identification of 5-Methylcytosine Sites in Genome-Wide DNA Promoters. Frontiers in Cell and Developmental Biology. 2020; 8: 614.
[84] Wang H, Tang J, Ding Y, Guo F. Exploring associations of non-coding RNAs in human diseases via three-matrix factorization with hypergraph-regular terms on center kernel alignment. Briefings in Bioinformatics. 2021 22: bbaa409.
[85] Ding Y, Tang J, Guo F. Identification of Drug-Target Interactions via Dual Laplacian Regularized least Squares with Multiple Kernel Fusion. Knowledge-Based Systems. 2020; 204: 106254.
[86] Ding Y, Tang J, Guo F. Identification of drug-target interactions via fuzzy bipartite local model. Neural Computing and Applications. 2020; 32: 10303–10319.
[87] Zeng X, Zhu S, Liu X, Zhou Y, Nussinov R, Cheng F. DeepDR: a network-based deep learning approach to in silico drug repositioning. Bioinformatics. 2019; 35: 5191–5198.
[88] Zeng X, Zhu S, Lu W, Liu Z, Huang J, Zhou Y, *et al*. Target identification among known drugs by deep learning from heterogeneous networks. Chemical Science. 2020; 11: 1775–1797.
[89] Zhai Y, Chen Y, Teng Z, Zhao Y. Identifying Antioxidant Proteins by Using Amino Acid Composition and Protein-Protein Interactions. Frontiers in Cell and Developmental Biology. 2020; 8: 591487.
[90] Guo Z, Wang P, Liu Z, Zhao Y. Discrimination of Thermophilic Proteins and Non-thermophilic Proteins Using Feature Dimension Reduction. Frontiers in Bioengineering and Biotechnology. 2020; 8: 584807.
[91] Jin Q, Cui H, Sun C, Meng Z, Su R. Free-form tumor synthesis in computed tomography images via richer generative adversarial network. Knowledge-Based Systems. 2021; 218: 106753.
[92] Wu X, Yu L. EPSOL: sequence-based protein solubility prediction using multidimensional embedding. Bioinformatics. 2021; 37: 4314–4320.