frontiers | Frontiers in Endocrinology

ORIGINAL RESEARCH
published: 26 April 2022
doi: 10.3389/fendo.2022.849549

# Identifying Pupylation Proteins and Sites by Incorporating Multiple Methods

*Wang-Ren Qiu*, Meng-Yue Guan, Qian-Kun Wang, Li-Liang Lou and Xuan Xiao*

School of Information Engineering, Jingdezhen Ceramic Institute, Jingdezhen, China

Pupylation is an important posttranslational modification in proteins and plays a key role in the cell function of microorganisms; an accurate prediction of pupylation proteins and specified sites is of great significance for the study of basic biological processes and development of related drugs since it would greatly save experimental costs and improve work efficiency. In this work, we first constructed a model for identifying pupylation proteins. To improve the pupylation protein prediction model, the KNN scoring matrix model based on functional domain GO annotation and the Word Embedding model were used to extract the features and Random Under-sampling (RUS) and Synthetic Minority Over-sampling Technique (SMOTE) were applied to balance the dataset. Finally, the balanced data sets were input into Extreme Gradient Boosting (XGBoost). The performance of 10-fold cross-validation shows that accuracy (ACC), Matthew's correlation coefficient (MCC), and area under the ROC curve (AUC) are 95.23%, 0.8100, and 0.9864, respectively. For the pupylation site prediction model, six feature extraction codes (i.e., TPC, AAI, One-hot, PseAAC, CKSAAP, and Word Embedding) served to extract protein sequence features, and the chi-square test was employed for feature selection. Rigorous 10-fold cross-validations indicated that the accuracies are very high and outperformed its existing counterparts. Finally, for the convenience of researchers, PUP-PS-Fuse has been established at https://bioinfo.jcu.edu.cn/PUP-PS-Fuse and http://121.36.221.79/PUP-PS-Fuse/as a backup.

**Keywords: pupylation, multiple features, post-translational modification, chi-square test, word embedding**

OPEN ACCESS

***Edited by:***
*Xianquan Zhan,*
*Shandong First Medical University,*
*China*

***Reviewed by:***
*Na Li,*
*Central South University, China*
*Canhua Huang,*
*Sichuan University, China*

****Correspondence:***
*Wang-Ren Qiu*
*qiuone@163.com*
*Xuan Xiao*
*jdzxiaoxuan@163.com*

***Specialty section:***
*This article was submitted to Molecular and Structural Endocrinology, a section of the journal Frontiers in Endocrinology*

***Received:*** *06 January 2022*
***Accepted:*** *07 March 2022*
***Published:*** *26 April 2022*

***Citation:***
*Qiu W-R, Guan M-Y, Wang Q-K, Lou L-L and Xiao X (2022) Identifying Pupylation Proteins and Sites by Incorporating Multiple Methods. Front. Endocrinol. 13:849549. doi: 10.3389/fendo.2022.849549*

## 1 INTRODUCTION

Pupylation is a kind of prokaryotic ubiquitin-like protein (Pup), a posttranslational protein modification (PTM) that occurs in actinomycetes, and has made a great contribution to the life process of many cells (1, 2). Ubiquitylation is one of the most common PTM modifications (3). In eukaryotes, ubiquitylation modification plays an important role in DNA repair, transcription regulation, control signal transduction, endocytosis, and sorting (4); research has shown that ubiquitylation modification is closely related to human health, such as lung cancer, breast cancer, type II diabetes, and other complex diseases (5–8). Pupylation is similar to ubiquitin in that Pup is attached to specific lysine residues. Since the PTM small protein modification was originally discovered in prokaryotes, the Pup in Mycobacterium tuberculosis (Mtb) plays an important role in the selection of protein degradation (5).

To better understand the biological mechanism of pupylation, the basic goal and fundamental task is to accurately and effectively predict the pupylation proteins and sites. For identifying PTM proteins, to the best of our knowledge, Qiu is the first one to have tried to identify phosphorylated (9) and acetylated (10) proteins, and nobody has done a similar work on pupylation protein until now. For a predictive analysis of pupylation sites, Liu proposed a GPS-PUP predictor for predicting pupylation sites with a group-based prediction system (GPS) method (11). Tung developed an iPUP predictor that implemented the support vector machine (SVM) algorithm with the composition of pairs of k-space amino acids (CKSAAP) (12). Chen designed a predictor called PupPred based on support vector machines (SVM), in which amino acid pairs were used to encode lysine-centered peptides (13). Hasan established a web server named pbPUP (14), which was a profile-based feature method to predict pupylation sites. Recently, FN Auliah developed PUP-Fuse web server for predicting pupylation sites (15); this algorithm was based on a variety of sequence features to predict pupylation sites. Although these algorithms could output higher specificity, their sensitivity scores are much lower.

In this work, a framework has been developed for predicting pupylation proteins and sites named as PUP-PS-Fuse, shown in **Figure 1**. In predicting the pupylation protein model, the KNN scoring matrix, the Word Embedding model (16–18), the Synthetic Minority Oversampling Technique(SMOTE) (19), and Random Under-sampling(RUS) (20) were applied to enhance the operation engine. Moreover, in the pupylation site prediction model, TPC (15, 21), AAI (22, 23), One-Hot (24), PseAAC (25, 26), CKSAAP (21, 27, 28), and Word Embedding (16–18) were used for feature extraction, and the chi-square test (15, 29, 30) was used to reduce the dimensionality of the feature space. Both these two models were verified with 10-fold cross-validation and compared with other existing predictors, the performance proved that this work is promising for the issue.

## 2 MATERIALS

### 2.1 Datasets for Predicting Pupylation Proteins

In this work, the negative samples were collected from UniProKB (2021_4), and the positive sample set was composed of 35 pupylation proteins collected from UniProKB and 233 pupylation proteins from PupDB (31). At least one pupylation

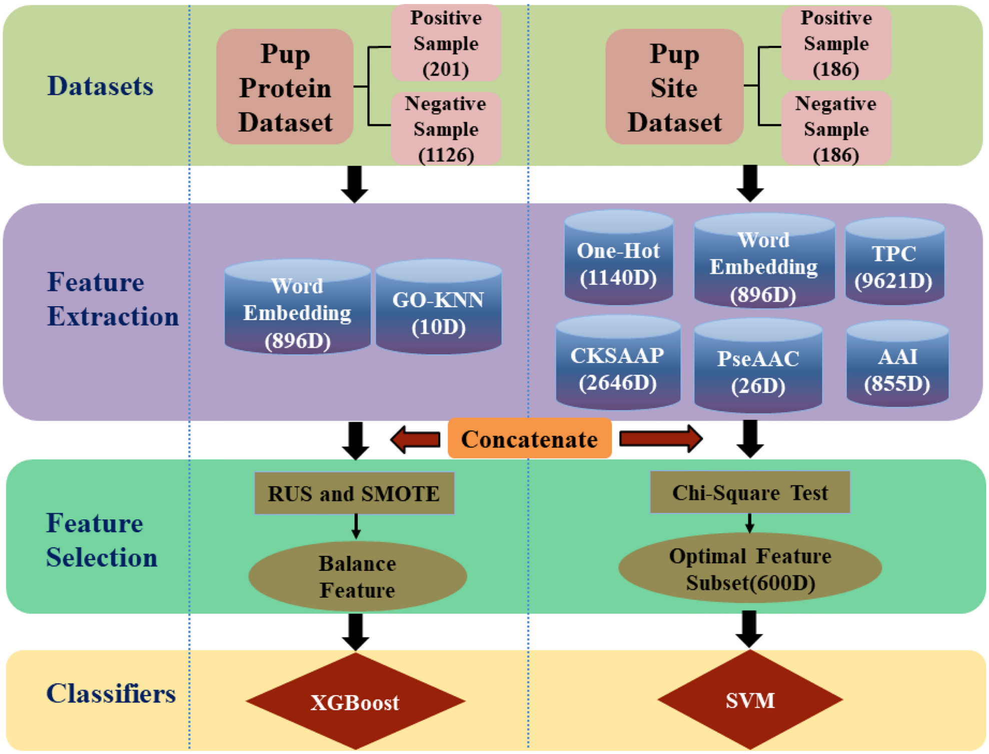

**FIGURE 1** | The framework of PUP-PS-Fuse (rounded squares represent data sets, cylinders represent feature extraction methods, rectangles and ellipses represent feature selection methods, and diamonds represent classifiers. RUS is the abbreviation of Random Under-sampling and SMOTE is the abbreviation of Synthetic Minority Over-sampling).

site must exist in any positive protein sequence, and none of the pupylation sites must appear in the negative samples. A given protein sequence can be expressed as P=$R_1R_2R_3...R_i...R_L$; here, $R_i$ represents the *i*th amino acid residue, and *L* represents the length of the protein sequence.

In order to make the results more rigorous, CD-HIT was used to remove 30% of the redundancy from 268 positive sampling as and 1,463 negative samples. Finally, 201 positive samples and 1,126 negative samples were collected for the proposed benchmark with a positive–negative ratio of 1:5.6.

## 2.2 Datasets for Predicting Pupylation Sites

This article used the same data set as that of Aulia (15). The data set was retrieved and obtained from the publications of PupDB (31) and contained 233 pupylation proteins which were subject to a cutting of redundancy treatment to remove those sequences that had ≥80% pairwise sequence identity with any other. After strictly following the aforementioned procedures, the training set consists of 186 amino acid fragments with pupylation site as positive samples and 372 negative samples without any pupylation site. As a result, the positive–negative ratio is 1:2. Since the imbalance of the data will affect the prediction results of the model, we balanced the training set with a positive–negative ratio of 1:1 (186 positive samples and 186 negative samples) by randomly deleting negative samples. The test set is composed of 87 positive samples and 191 negative samples by randomly extracting from the benchmark data set. **Table 1** summarizes the data sets for predicting pupylation proteins and pupylation sites.

In order to formulate the pupylation site sequence in more detail and more comprehensively, the sequence fragment of the potential pupylation site can be expressed in the form of formula (1):

$$\boldsymbol{\theta_\delta(K) = R_1R_2 - R_{\delta-1}R_\delta K R_{\delta+1}R_{\delta+2}\cdots R_{2\delta-1}R_{2\delta}} \quad (1)$$

Where $R_1$ to $R_\delta$ represent the amino acid residues on the left of *K*, $R_{\delta+1}$ to $R_{2\delta}$ represent the amino acid residues on the right of *K*, δ is an integer, and the middle *K* means *Lysine* (32). In addition, the peptide sequence $\theta_\delta(K)$ can be divided into $\theta_\delta^+(K)$ and $\theta_\delta^-(K)$ (see formula (2)), where $\theta_\delta^+(K)$ represents a pupylation protein sequence fragment whose center point is *K*, and $\theta_\delta^-(K)$ denotes non-pupylation protein sequence fragments whose center point is *K*. The sliding window method was used to segment pupylation protein sequences with different window sizes. Judging from the analysis of the pupylation protein sequence preferred by FN Aulia et al. (15), it can be seen that the prediction is the best when the window size is 57 with δ = 28.

**TABLE 1** | Data set for prediction of pupylation protein and pupylation site.

| Datasets | Positive | Negative | Ratio |
|---|---|---|---|
| Pupylation proteins | 201 | 1126 | 1:5.6 |
| Pupylation site training | 186 | 186 | 1:1 |
| Pupylation site test | 87 | 191 | 1:2.2 |

*Positive represents the number of positive samples, and Negative represents the number of negative samples.*

When the sequence fragments were divided, in order to make the site sequence equal in length, the missing amino acids were filled in with *X* residues. As a result, the pupylation site data set adopts the form of formula (2):

$$\boldsymbol{\theta_\delta(K) = \theta_\delta^+(K) \cup \theta_\delta^-(K)} \quad (2)$$

Among them, the subset of positive samples $\theta_\delta^+(K)$ represents a true pupylation site segment with *K* at its center, and the subset of negative samples $\theta_\delta^-(K)$ represents the false pupylation site fragment.

# 3 FEATURE EXTRACTION AND METHODS

## 3.1 Feature Extraction Methods for Predicting Pupylation Proteins

The basic step for predicting pupylation protein is to extract features of the protein sequence, and it is a key step that affects the effectiveness of the prediction model. When predicting pupylation protein, we chose GO-KNN (10) and Word Embedding coding schemes to extract protein sequence information.

### 3.1.1 GO-KNN

GO-KNN (10) is based on the KNN scoring matrix of functional domain GO annotations to extract features. In this study, we need to obtain the GO information of all proteins. For a protein without any GO information, we replace it with GO terms of its homologous protein and then calculate the distance between any two protein sequences. Taking protein $R_1$ and $R_2$ as example, their GO annotations can be expressed by $R_{GO}^1 = \{GO_1^1, GO_2^1, \cdots, GO_M^1\}$ and $R_{GO}^2 = \{GO_1^2, GO_2^2, \cdots, GO_N^2\}$, $GO_i^1$ and $GO_i^2$ represent the *i*th GO of the proteins $R_1$ and $R_2$, respectively, and *M* and *N* are the numbers of their GO terms. The feature extraction steps are listed as follows:

(a). Calculating the distance between two proteins, as in formula (3).

$$\boldsymbol{Distance(R_1, R_2) = 1 - \frac{\lfloor R_{GO}^1 \cap R_{GO}^2 \rfloor}{\lfloor R_{GO}^1 \cup R_{GO}^2 \rfloor}} \quad (3)$$

Where ∪ and ∩ represent the intersection and union of sets, and ⌊ ⌋ represents the number of elements in the set.

(b) Sorting all the calculated distances from small to large.

(c) Calculating the percentage of positive samples in the *Y* neighbors.

In this study, the *Y* values were selected in order of 2, 4, 8, 16, 32, 64, 128, 256, and 1,024. Finally, a 10-dimensional feature vector was formed. Therefore, the digital feature vector of protein $R_1$ can be expressed as: $(x_1, x_{2,\ldots,} x_{10})$.

### 3.1.2 Word Embedding

Word Embedding (16–18) is a method for converting words in text into digital vectors. The Word Embedding process was used to embed the high-dimensional space containing all the number of words into a low-dimensional continuous vector space, each word or phrase was mapped to a vector in the real number domain, and the word vector was generated as a result of the Word Embedding. In this study, we quoted the word embedding method of Qiu (33, 34). This briefly introduces how word embedding was applied in this research as described below.

Step 1: Firstly, the pupylation protein sequence was split into fragments and a wordbook is created. In this study, we used three different word embedding models, and the pupylation protein sequence is cut into different fragment lengths. Their fragment lengths can be set to 2, 3, or 4, respectively, and the step size of the moving window is 1.

Step 2: The CBOW (Continuous Bag-of-Words) model was used to train the data. In order to speed up the training speed of word vectors, the negative sampling technique (35) and backpropagation algorithm (36) were adopted in the CBOW model. At this step, the dimension sizes of the word vectors were selected as 128, 256, and 512, respectively, and we then obtained three vectors $W_{128}$, $W_{256}$, and $W_{512}$ for a given protein sequence.

Step 3: >A protein sequence was represented by combining CBOW vectors. At this step, we merge the features of each pupylation protein sequence of the three aforementioned words vector, as shown in formula (4), and finally get an 896-dimensional vector.

$$V = W_{128} \oplus W_{256} \oplus W_{512} \tag{4}$$

Among them, $W_{128}$, $W_{256}$, and $W_{512}$ mean 128-, 256-, and 512-dimensional word vectors, and ⊕ means to concatenate a two-word vector.

## 3.2 Feature Extraction Methods for Predicting Pupylation Sites

For predicting pupylation sites, TPC (15, 21), AAI (22, 23), One-Hot (21, 37), PseAAC (25, 26), CKSAAP (21, 27, 28), and Word Embedding (17, 18) coding schemes were involved in extracting protein fragment [for example, formula (2)] information and are briefly described as follows.

### 3.2.1 TPC

The first feature extraction algorithm applied for predicting pupylation sites in this paper is TPC (15, 21) which codes protein fragment information by calculating the frequency of occurrence of three consecutive amino acid pairs. Bian et al. (38) identified mitochondrial proteins of *Plasmodium*. In this method, we divide the number of occurrences of each of the three consecutive amino acid pairs in the fragment by the total number of all possible tripeptides [refer to formula (5)], and finally form a 9,261-dimensional digital feature vector.

$$\boldsymbol{p_i = \frac{N_i}{\sum_{1}^{9261} N_i}} \tag{5}$$

where $N_\mathrm{i}$ represents the number of occurrences of the *i*th three consecutive amino acid pairs in the fragment.

### 3.2.2 AAI

The second algorithm, AAI code, is based on AAindex (22, 23), which is a database that collects more than 500 amino acid indexes. After evaluating the different physicochemical and biological properties of amino acids, the top 15 useful and informative amino acid indexes selected by FN Auliah et al. (15) were used in this paper (fifteen types of AAI properties can be found at https://www.mdpi.com/1422-0067/22/4/2120/s1), with a window sequence length of 57. Therefore, AAI encoding produced 855 (57 × 15) dimensional feature vectors.

### 3.2.3 One-Hot

One-Hot coding (21, 37) is based on the 0–1 coding scheme. In this coding scheme, each amino acid is represented by a 20-dimensional binary vector. For example, alanine A is transformed into a vector (10000000000000000000), cysteine C is transformed into a vector (01000000000000000000), tyrosine Y is transformed into a vector (00000000000000000001), etc. In this study, a pseudo-amino acid code *X* was selected to represent it, which is represented by a (00000000000000000000) vector. The sequence length of the window is 57, so the total dimension of the proposed One-Hot feature vector is 20×(2δ+1), i.e., 1,140, dimensions.

### 3.2.4 PseAAC

PseAAC (25, 26) coding has been widely used in the study of protein and protein-related problems. It can be called a “pseudo-amino acid composition” model to represent protein samples. Here, six physical and chemical properties of amino acids, hydrophobicity, hydrophilicity, molecular side chain mass, PK1, PK2, and PI, were selected to convert the protein sequence into the feature vector. The parameters ω and λ were set to 0.05 and 5, respectively [the values of ω and λ are clearly explained by Chou (39) et al.]. Finally, a 25-dimensional digital feature vector is formed.

$$p_i = \begin{cases} \frac{f_i}{\sum_{i=1}^{20} f_i + \omega \sum_{J=1}^{\lambda} \theta_j} (1 \le i \le 20) \\ \frac{\omega \theta_{i-20}}{\sum_{i=1}^{20} f_i + \omega \sum_{j=1}^{\lambda} \theta_j} (20 + 1 \le i \le 20 + \lambda) \end{cases} \tag{6}$$

### 3.2.5 CKSAAP

CKSAAP (21, 27, 28) coding is a coding scheme based on *K*-spaced amino acid pairs. In the coding process, a protein sequence contains 441 (21 × 21) amino acid pairs (AA, AC, AD,…, XX) and is expressed by formula (7).

$$\left(\frac{F_{AA}}{F_N}, \frac{F_{AC}}{F_N}, \frac{F_{AD}}{F_N}, \cdots, \frac{F_{XX}}{F_N}\right) 441 \tag{7}$$

Where, $F_{AA}$, $F_{AC}$, $F_{AD}$, $F_{XX}$, represents the number of times the corresponding amino acid pair appears in the protein sequence, and $L$ is used in this article to represent the length of the protein sequence, $F_N = L - k$ - 1. For each k, 441 pairs of residues are formed, where k represents the space between two amino acids, the values of $k$ are 0, 1, 2, 3, 4, 5, and the best $k_{max}$ setting is 5. Therefore, each corresponding protein sequence can be represented with a 2,646 (21 × 21 × ($k_{max}$ +1)) dimensional feature vector.

## 3.3 Data Balancing and Feature Selection

In the model of pupylation protein prediction, the number of positive samples is 201 and the number of negative samples is 1126, and the ratio of positive to negative samples is approximately 1:5.6. Since it is an unbalanced data set, Random Under-sampling (RUS) (20) and Synthetic Minority Oversampling (SMOT) (19, 20) were used to process the sample data. Actually, the RUS is a very simple and popular under-sampling technique and the SMOT is one of the most popular methods in oversampling proposed by Chawla et al. (40).

In the model of pupylation site prediction, fusion of multiple features would generate a high-dimensional vector, and there may be some redundant or irrelevant features. Therefore, the chi-square test (15, 29, 30) was used to select the most beneficial feature. The chi-square test was first proposed by Karl Pearson (41), usually called the Pearson chi-square test, which is currently the most popular non-parametric(or no distribution) test based on the hypothesis of the chi-square $\chi^2$ distribution test method (42). In the model, the first 600-dimensional features were selected to get a better prediction result.

# 4 MODEL EVALUATION METRICS AND OPERATION ENGINE

## 4.1 Model Evaluation Metrics

In this study, four indicators were used to evaluate the performance of the model. They are Accuracy (ACC) (43), Sensitivity (SN), Specificity (SP), and Matthews Correlation Coefficient (MCC) (44–47), which are defined as Eq. (8).

$$\begin{cases} Sn = \frac{TP}{TP+FN} \\ Sp = \frac{TN}{TN+FP} \\ ACC = \frac{TP+TN}{TP+FP+TN+FN} \\ MCC = \frac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)\times(TP+FN)\times(TN+FP)\times(TN+FN)}} \end{cases} \quad (8)$$

In addition, the prediction accuracy can also be measured and analyzed using the ROC curve. For the prediction method, the ROC (48) curve plots the true positive rate (Sn) and false positive rate (Sp) of all possible thresholds as a function of the relationship. The calculation of AUC also provides a comprehensive understanding of the proposed prediction method. Generally, the closer the AUC (49) value is to 1, the better the prediction method.

## 4.2 Operation Engine

Most of the classification algorithms can handle the data with the digital vector; thus, this work tried diverse approaches include Random Forest (RF), Support Vector Machine (SVM), K nearest neighbor (KNN), eXtreme Gradient Boosting (XGBoost), and Ensemble Learning. Since they have been widely used in various fields such as marketing management (50), bioinformatics (51), and image retrieval (52), we would not repeat their principles in this manuscript in detail.

In fact, the Random Forest (RF) (51, 53) algorithm is based on the classification and regression tree (CART) (54) technology which is formed by integrating multiple decision trees through the idea of integrated learning. In the RF model, each decision tree is a classifier. For a given sample, each tree will get a classification result. All the voting results are integrated, and the final output is the category with the most votes. The SVM (55) is a supervised learning model whose main idea is to find the hyperplane that distinguishes the two types, to maximize the margin, some points in the sample that are closest to the hyperplane; these points are called support vectors. The KNN (56, 57) is a supervised learning model, and its main idea is to determine which category it belongs to when predicting a new value based on the category of the nearest *K* points. XGBoost (58) is an open-source machine learning project developed by Chen et al. It efficiently implements the GBDT (59) algorithm and has made many improvements to the algorithm and engineering.

Ensemble learning (60) is an important method for improving prediction accuracy in current data mining and machine learning. It is frequently used in the field of machine learning (5) due to its "fault tolerance." It has better classification results than individual classifiers. The ensemble method is a meta-algorithm that combines several machine learning techniques into a predictive model. There are three commonly used frameworks for ensemble learning: Bagging (61) to reduce variance, Boosting (62) to reduce bias, and Stacking (63) to improve prediction results. In this research, we used the Stacking ensemble learning algorithm. The main idea of Stacking is as follows: we firstly train multiple different models, and then use the output of each model trained before as input to train a model to get a final output. For predicting Pupylation sites, we use three base classifiers, namely, RF, SVM, and KNN, and then use LogisticRegression (LR) to classify the results of the base classification to get the final classification results.

# 5 RESULTS AND DISCUSSION

## 5.1 Results and Discussion of Pupylation Proteins Prediction

### 5.1.1 Effect of the Different Features

In this study, the two single feature encoding methods are GO-KNN and Word Embedding, and 10 dimensions and 896 dimensions are obtained respectively. These two kinds of features have been fused into a 906-D feature vector PUP-P-Fuse. Through the 10-fold cross-folding verification, the prediction results of different features are shown in **Table 2**.

From **Table 2**, we can know that the prediction results after fusion are not as good as we expected; the best prediction performance is GO-KNN's with ACC of 94.36%, Sn of 77.08%, Sp of 97.45%, MCC of 0.7731, and AUC of 0.9530, which are slightly higher than those of CBOW and PUP-P-Fuse (see to the first 4 line of **Table 2**).

### 5.1.2 Effect of the RUS and SMOTE

Using Random Under-sampling (RUS) and Synthetic Minority Over-sampling (SMOTE) to balance the data, and then through

**TABLE 2** | The prediction results of different feature extraction and balance methods for predicting pupylation proteins.

| | Feature | ACC (%) | Sn (%) | Sp (%) | MCC | AUC |
|---|---|---|---|---|---|---|
| Unbalanced | GO-KNN | 94.36 | 77.08 | 97.45 | 0.7731 | 0.9530 |
| | CBOW | 91.91 | 67.42 | 96.27 | 0.6700 | 0.9553 |
| | PUP-P-Fuse | 92.07 | 60.25 | 97.77 | 0.6615 | 0.9647 |
| Balanced | PUP-P-Fuse | **95.40** | **92.03** | 96.00 | **0.8327** | **0.9840** |

*GO-KNN and CBOW represent two feature extraction methods for predicting pupylation proteins, and PUP-P-Fuse is a fusion of the above two methods.*
*The bold values are means the best performance of the column with the same metric and are showed in following tables with the same meaning.*

10-fold cross-folding verification, the prediction results of ACC, Sn, Sp, MCC, and AUC on balanced and unbalanced data sets were obtained and are shown in **Table 2**.

From the last line of **Table 2**, we can see that the PUP-P-Fuse's ACC, Sn, MCC, and AUC predictive indicators have increased by 3%, 32%, 17%, and 2%, respectively, after the RUS and SMOTE technology balance. Therefore, the results show that multifeature fusion (PUP-P-Fuse) can improve the performance. In order to better analyze the influence of different features on pupylation protein prediction, the results obtained by two single coding and fusion features are as shown in **Figure 2**.

From **Figure 2**, we can see that the ACC, Sn, Sp, MCC, and AUC of GO-KNN are 93.37%, 82.64%, 95.36%, 0.7519, and 0.9509, respectively. Those of CBOW and PUP-P-Fuse are denoted with red and green bars, respectively. Compared with GO-KNN and CBOW's ACC, Sn, Sp, MCC, and AUC predictive indicators, the PUP-P-Fuse increased by 2%–4%, 10%–11%, 0.6%–2%, 8%–13%, and 3%, respectively. In summary, all indicators of PUP-P-Fuse are higher than the other two models after data balancing. Therefore, it is proper to use RUS and SMOT in this issue.

### 5.1.3 Effect of Classifiers

Classifiers play an important role in prediction. In this work, we used the above five classifiers to identify pupylation proteins. After 10-fold cross-folding verification, the results of ACC, Sn, Sp, MCC, and AUC of each classifier are shown in **Table 3**. From **Table 3**, we can see that XGBoost gained the best performance on each evaluation index. In order to better compare the effects of different classifiers, the prediction results of the five classifiers are as shown in **Figure 3**.

The area under the ROC curve can evaluate the predictive performance of the model. It is seen in **Figure 3** that the XGBoost classifier, of which AUC is 0.9840, is the best choice for the proposed model.

### 5.1.4 Effect of Features on the Independent Dataset

To verify the effect of the PUP-P-Fuse model, we used 67 pupylation proteins and 134 negative samples for independent testing; PUP-P-Fuse has the highest performance, as shown in **Table 4**. It can be seen that the effect of the PUP-P-Fuse model is still very good. However, from **Table 4** we can see that the overall

**TABLE 3** | The prediction results of different classifiers for predicting pupylation proteins.

| Algorithms | Acc (%) | Sn (%) | Sp (%) | MCC | AUC |
|---|---|---|---|---|---|
| XGBoost | **95.40** | **92.03** | **96.00** | **0.8327** | **0.9840** |
| Ensemble Learning | 93.87 | 90.61 | 94.48 | 0.7874 | 0.9788 |
| SVM | 91.36 | 93.65 | 90.96 | 0.7335 | 0.9689 |
| RF | 92.87 | 82.40 | 94.75 | 0.7355 | 0.9703 |
| KNN | 83.88 | 96.90 | 81.55 | 0.6104 | 0.9585 |

*The bold values are means the best performance of the column with the same metric and are showed in following tables with the same meaning.*

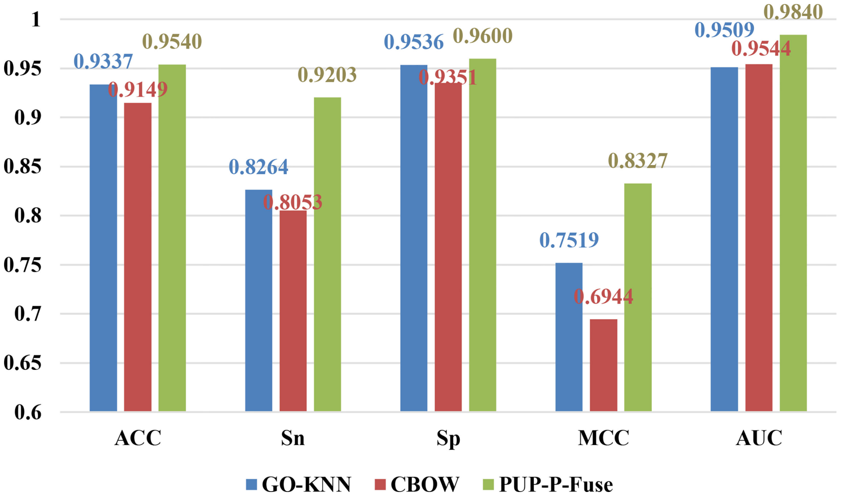

**FIGURE 2** | The prediction results of different characteristics on balanced data for predicting pupylation proteins.

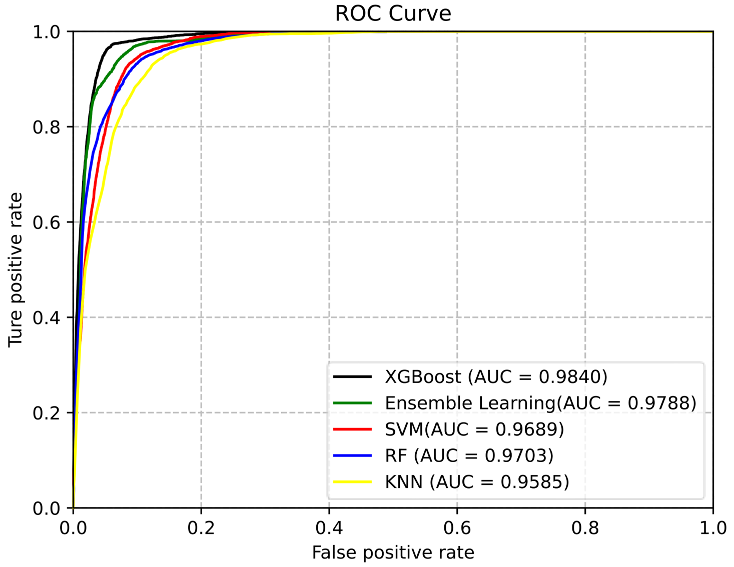

**FIGURE 3** | ROC curves of different classifiers for predicting the pupylation protein.

**TABLE 4** | The prediction results of different classifiers on the testing set of pupylation proteins.

| Algorithms | Acc (%) | Sn (%) | Sp (%) | MCC | AUC |
|---|---|---|---|---|---|
| XGBoost | 84.66 | 80.99 | 86.62 | 0.6630 | 0.9251 |
| Ensemble Learning | 85.34 | 80.96 | 87.41 | 0.6738 | 0.9376 |
| SVM | 85.48 | 88.78 | 83.85 | 0.6955 | 0.9317 |
| RF | 84.55 | 79.15 | 87.79 | 0.6571 | 0.9270 |
| KNN | 78.56 | 83.97 | 75.61 | 0.5653 | 0.8868 |

performance of the SVM classifier is better than those of other classifiers.

## 5.2 Results and Discussion of Pupylation Site Prediction

### 5.2.1 Effect of Features on the Training Dataset

In this study, six single-feature codes are AAI, One-Hot, PseAAC, Word Embedding, CKSAAP, and TPC, and the feature PUP-S-Fuse was obtained after fusion. The six features are coded separately and obtained 855, 1140, 26, 896, and 2,646 dimensions, respectively. Through 10-fold cross-folding verification, we choose the SVM classifier for training. Without feature selection, we obtain the prediction results of different feature extractions with a ratio of positive samples to negative samples of 1:1, as shown in **Table 5**.

From **Table 5**, we can see that the ACC, Sp, MCC, and AUC indicators of TPC are all higher than other single codes, and the Sn indicators of Word Embedding are all higher than other single codes. The fusion feature code PUP-S-Fuse performs better than any single feature on ACC, Sn, Sp, MCC, and AUC indicators. Therefore, feature fusion is very necessary for this issue.

### 5.2.2 Effect of the Chi-Square Test on the Training Dataset

As regards the model for predicting the pupylation site, we selected different *K* values for the chi-square test and compared them and found that the prediction effect has been relatively greatly improved after the chi-square test was used to select features.

It is seen in **Table 6** that when the *K* value is selected as 600, the ACC, Sn, and MCC of the pupylation site are predicted to be higher than other *K* values. When the *K* value is selected as 1,000,

**TABLE 5** | The effect of different feature extraction methods on the training set of pupylation sites.

| Features | ACC (%) | Sn (%) | Sp (%) | MCC | AUC |
|---|---|---|---|---|---|
| AAI | 56.71 | 56.21 | 57.52 | 0.1380 | 0.6148 |
| One-Hot | 57.49 | 59.49 | 55.95 | 0.1550 | 0.6296 |
| PseAAC | 61.56 | 62.00 | 61.64 | 0.2367 | 0.6597 |
| Word Embedding | 69.92 | 73.36 | 66.55 | 0.4001 | 0.7645 |
| CKSAAP | 68.84 | 68.92 | 69.20 | 0.3818 | 0.7596 |
| TPC | 70.36 | 70.69 | 70.65 | 0.4143 | 0.7697 |
| PUP-S-Fuse | **74.00** | **80.00** | **68.55** | **0.4883** | **0.7951** |

*The bold values are means the best performance of the column with the same metric and are showed in following tables with the same meaning.*

**TABLE 6** | The effect of feature fusion Pup-S-Fuse by using the chi-square test for predicting pupylation sites.

| Features | ACC (%) | Sn (%) | Sp (%) | MCC | AUC |
|---|---|---|---|---|---|
| K = 200 | 89.09 | 88.82 | 89.57 | 0.7830 | 0.9531 |
| K = 400 | 91.21 | 92.89 | 89.52 | 0.8256 | 0.9565 |
| **K = 600** | **92.30** | **93.97** | **90.71** | **0.8477** | 0.9599 |
| K = 800 | 91.99 | 93.31 | 90.55 | 0.8400 | 0.9634 |
| K = 1,000 | 92.00 | 92.27 | **91.78** | 0.8394 | **0.9641** |
| K = 1,200 | 90.70 | 91.77 | 89.70 | 0.8145 | 0.9604 |

*The bold values are means the best performance of the column with the same metric and are showed in following tables with the same meaning.*

**TABLE 7 |** The prediction results of different classifiers for predicting pupylation sites.

| Algorithms | Acc (%) | Sn (%) | Sp (%) | MCC | AUC |
|---|---|---|---|---|---|
| EL | **92.30** | 93.97 | **9071** | **0.8477** | 0.9599 |
| SVM | 91.72 | **95.27** | 88.59 | 0.8377 | **0.9659** |
| RF | 86.72 | 87.50 | 86.24 | 0.7361 | 0.9347 |
| KNN | 81.34 | 90.37 | 75.63 | 0.6706 | 0.9388 |
| XGBoost | 78.49 | 79.02 | 77.94 | 0.5703 | 0.8622 |

*EL, ensemble learning.*

*The bold values are means the best performance of the column with the same metric and are showed in following tables with the same meaning.*

the Sp and AUC values of the pupylation site are higher than those of other *K* values. Therefore, from the overall effect, we finally selected 600 for predicting the pupylation site.

### 5.2.3 Effect of Classifiers on the Training Dataset

Choosing the right machine learning (ML) algorithm is also a crucial step for predicting results. When predicting pupylation sites, we used RF, SVM, KNN, Ensemble Learning (EL), and XGBoost algorithms. In order to verify the effectiveness and superiority of the EL algorithm used to predict pupylation sites, we compared these algorithms through 10-fold cross-validation on the same training set. The prediction results are shown in **Table 7**.

From **Table 7**, although we know that the prediction effect of the EL classifier and SVM classifier is better, the overall prediction effect of the EL is better than that of the SVM. The prediction results of RF, KNN, and XGBoost are relatively poor. In order to evaluate the performance of the classifier more comprehensively, the ROC curves of different classifiers are as shown in **Figure 4**.

From **Figure 4**, we can clearly see that the area under the ROC curve of EL and SVM is the largest, and the AUC of EL is about 2%–10% higher than that of other ML models. Therefore, EL was selected as the best classifier for predicting pupylation sites.

### 5.2.4 Comparison With Other Methods on Independent Datasets

In order to compare PUP-S-Fuse with the existing five methods (GPS-PUP, iPUP, PUPS, PbPUP, and PUP-Fuse), tests were performed on the same independent set which contains 86 pupylation sites and 1,136 non-pupylation sites from 71 pupylation proteins. PUP-S-Fuse and PUP-Fuse were trained with the same training data set mentioned above, and the other four methods were quoted from the references. In the fairly compared performance, PUP-S-Fuse provided the highest performance, as shown in **Table 8**.

From **Table 8**, we know that the performance of PUP-S-Fuse on the test set is also better than that of PUP-Fuse. Acc, Sn, Sp, and MCC are increased by 9%, 19%, 6%, and 24%, respectively, which proves that PUP-S-Fuse is superior to existing predictors.

**TABLE 8 |** Comparison of methods on Independent Dataset for predicting pupylation sites.

| Methods | Acc (%) | Sn (%) | Sp (%) | MCC | AUC |
|---|---|---|---|---|---|
| iPUP | 73 | 40 | 88 | 0.32 | |
| GPS-PUP | 68 | 21 | 89 | 0.13 | |
| PUPS | 67 | 17 | 89 | 0.08 | |
| pbPUP | 79 | 48 | 82 | 0.45 | |
| PUP-Fuse | 82 | 59 | 91 | 0.55 | |
| PUP-S-Fuse | **91.35** | **78.26** | **97.38** | **0.7953** | **0.9550** |

*The bold values are means the best performance of the column with the same metric and are showed in following tables with the same meaning.*

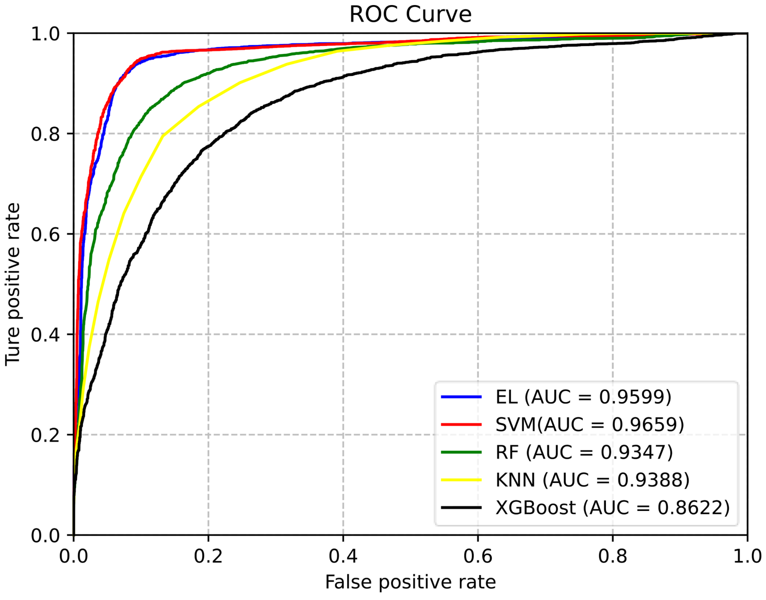

**FIGURE 4 |** The ROC curves of different classification methods for predicting pupylation sites. (EL is the abbreviation of ensemble learning).

## 6 WEB SERVER AND USER GUIDE

The actual application value of a prediction method can be significantly improved if it has a web server that can be viewed by the public; accordingly, the PUP-PS-Fuse web server has been established. To maximize the convenience of most experimental scientists, a guide for users is provided below.

Step 1. Opening the web server at "https://bioinfo.jcu.edu.cn/PUP-PS-Fuse," the server consists of four main modules, namely, Pupylation Protein, Pupylation Site, Download (data download), and **Help** (website usage guide). You will see the top page of PUP-PS-Fuse on your computer screen.

Step 2. In the Pupylation Protein prediction module, you can enter the protein sequence in the input file box, but it must be in FASTA format. You can also click the example button where you will see that there are a correct example and an incorrect example as well as the text input format. Click the Close button, and you will return to the pupylation Protein prediction interface. Click the Submit button to get the prediction results. After 20 seconds or so since your submitting, you will see the following on the screen of your computer: "The Pupylation protein list includes …" and "The non-Pupylation protein list includes …"

Step 3. In the Pupylation Site prediction module, you can enter the protein sequence in FASTA format in the input file box. In the example_site submodule, you will see that there are a correct example and an incorrect example as well as the text input format. Click the Close button, and you will return to the pupylation Site prediction interface. Click the Submit button to get the predicted results. After 2 min or so since your submitting, you will see the following on the screen of your computer: 'The number of "K" is X. Location $M_1$, $M_2$, $M_3$, … is(are) predicted to be Pupylation Site(s).'

In the Download module, you can download the Pupylation protein dataset and Pupylation site dataset (also available in the **Supplementary Material**). By the way, you can click on the Help button to see a brief introduction about the predictors.

## 7 CONCLUSION

PUP-PS-Fuse was developed to predict pupylation proteins and sites. In order to predict pupylation proteins, GO-KNN and Word Embedding served as feature extraction methods. In the work, GO-KNN extracted features based on the KNN score matrix of functional domain GO annotations, and Word Embedding converted information of the amino acid sequence into digital feature vectors. In addition, RUS and SMOT technology were used to deal with the imbalance of the data set to reduce the negative impact of imbalance on the model. Finally, the XGBoost classifier was selected to make predictions. In order to predict pupylation sites, six feature extraction codes and one fusion feature extraction code are used, named as TPC, AAI, One-Hot, PseAAC, CKSAAP, Word Embedding, and PUP-S-Fuse. In order to improve the computational efficiency and eliminate the redundancy and noise generated by the fusion feature, the chi-square test served to reduce the dimensionality of the fusion feature. The selected feature subset was input into the Ensemble Learning for classification, and then 10-fold cross-folding was used for verification. The performance of PUP-S-Fuse is evaluated based on an independent test data set, and compared with other existing methods, it is concluded that the predictive performance of PUP-S-Fuse is better than other existing methods. These processes only require calculation models and do not require any physical and chemical experiments, which saves experimental costs and improves work efficiency. We hope that this work will be helpful for dealing with some related biological problems with computational methods.

## DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**. Further inquiries can be directed to the corresponding authors.

## AUTHOR CONTRIBUTIONS

W-RQ conceived and designed the experiments. M-YG, Q-KW, and L-LL performed the extraction of features, model construction, model training, and evaluation. M-YG drafted the manuscript. XX and W-RQ supervised this project and revised the manuscript. All authors contributed to the article and approved the submitted version.

## FUNDING

This work was supported by grants from the National Natural Science Foundation of China (Nos. 31760315, 62162032, 61761023) and Natural Science Foundation of Jiangxi Province, China (No. 20202BAB202007).

## SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fendo.2022.849549/full#supplementary-material

## REFERENCES

1. Li T, Chen Y, Li T, Jia C. Recognition of Protein Pupylation Sites by Adopting Resampling Approach. *Molecules* (2018) 23(12):3097–110. doi: 10.3390/molecules23123097
2. Barandun J, Delley CL, Weber-Ban E. The Pupylation Pathway and Its Role in Mycobacteria. *BMC Biol* (2012) 10(1):95. doi: 10.1186/1741-7007-10-95
3. Garcia BA, Hake SB, Diaz RL, Kauer M, Morris SA, Recht J, et al. Organismal Differences in Post-Translational Modifications in Histones H3 and H4. *J Biol Chem* (2007) 282(10):7641–55. doi: 10.1074/jbc.M607900200

4. Herrmann J, Lerman LO, Lerman A. Ubiquitin and Ubiquitin-Like Proteins in Protein Regulation. *Circ Res* (2007) 100(9):1276–91. doi: 10.1161/01.RES.0000264500.11888.f0
5. Afolabi LT, Saeed F, Hashim H, Petinrin OO. Ensemble Learning Method for the Prediction of New Bioactive Molecules. *PloS One* (2018) 13(1):e0189538. doi: 10.1371/journal.pone.0189538
6. Faus H, Haendler B. Post-Translational Modifications of Steroid Receptors. *BioMed Pharmacother* (2006) 60(9):520–8. doi: 10.1016/j.biopha.2006.07.082
7. Poulsen C, Akhter Y, Jeon AH, Schmitt-Ulms G, Meyer HE, Stefanski A, et al. Proteome-Wide Identification of Mycobacterial Pupylation Targets. *Mol Syst Biol* (2010) 6(1):386. doi: 10.1038/msb.2010.39
8. Imkamp F, Rosenberger T, Striebel F, Keller PM, Amstutz B, Sander P, et al. Deletion of Dop in Mycobacterium Smegmatis Abolishes Pupylation of Protein Substrates *In Vivo*. *Mol Microbiol* (2010) 75(3):744–54. doi: 10.1111/j.1365-2958.2009.07013.x
9. Qiu WR, Sun BQ, Xiao X, Xu D, Chou KC. Iphos-PseEvo: Identifying Human Phosphorylated Proteins by Incorporating Evolutionary Information Into General PseAAC *via* Grey System Theory. *Mol Inform* (2017) 36(5-6):1600010. doi: 10.1002/minf.201600010
10. Qiu WR, Xu A, Xu ZC, Zhang CH, Xiao X. Identifying Acetylation Protein by Fusing Its PseAAC and Functional Domain Annotation. *Front Bioeng Biotechnol* (2019) 7:311. doi: 10.3389/fbioe.2019.00311
11. Liu Z, Ma Q, Cao J, Gao X, Ren J, Xue Y. GPS-PUP: Computational Prediction of Pupylation Sites in Prokaryotic Proteins. *Mol Biosyst* (2011) 7(10):2737–40. doi: 10.1039/c1mb05217a
12. Tung, Chun-Wei. Prediction of Pupylation Sites Using the Composition of K-Spaced Amino Acid Pairs. *J Theor Biol* (2013) 336(Complete):11–7. doi: 10.1016/j.jtbi.2013.07.009
13. Chen X, Qiu JD, Shi SP, Suo SB, Liang RP. Systematic Analysis and Prediction of Pupylation Sites in Prokaryotic Proteins. *PloS One* (2013) 8(9):e74002. doi: 10.1371/journal.pone.0074002
14. 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(6):e0129635. doi: 10.1371/journal.pone.0129635
15. Auliah FN, Nilamyani AN, Shoombuatong W, Alam MA, Hasan MM, Kurata H. PUP-Fuse: Prediction of Protein Pupylation Sites by Integrating Multiple Sequence Representations. *Int J Mol Sci* (2021) 22(4) 2120. doi: 10.3390/ijms22042120
16. Thapa N, Chaudhari M, McManus S, Roy K, Newman RH, Saigo H. DeepSuccinylSite: A Deep Learning Based Approach for Protein Succinylation Site Prediction. *BMC Bioinf* (2020) 21(3):1–10. doi: 10.1186/s12859-020-3342-z
17. Yang KK, Wu Z, Bedbrook CN, Arnold FH, Wren J. Learned Protein Embeddings for Machine Learning. *Bioinformatics* (2018) 34(15):2642–8. doi: 10.1093/bioinformatics/bty178
18. Wang H, Wang Z, Li Z, Lee TY. Incorporating Deep Learning With Word Embedding to Identify Plant Ubiquitylation Sites. *Front Cell Dev Biol 8 (September* (2020) 2020):572195. doi: 10.3389/fcell.2020.572195
19. Das S, Datta S, Chaudhuri BB. Handling Data Irregularities in Classification: Foundations, Trends, and Future Challenges. *Pattern Recognition* (2018) 81:674–93. doi: 10.1016/j.patcog.2018.03.008
20. Kim M-J, Kang D-K, Kim HB. Geometric Mean Based Boosting Algorithm With Over-Sampling to Resolve Data Imbalance Problem for Bankruptcy Prediction. *Expert Syst Appl* (2015) 42(3):1074–82. doi: 10.1016/j.eswa.2014.08.025
21. Chen YZ, Tang YR, Sheng ZY, Zhang Z. Prediction of Mucin-Type O-Glycosylation Sites in Mammalian Proteins Using the Composition of K-Spaced Amino Acid Pairs. *BMC Bioinf* (2008) 9:101. doi: 10.1186/1471-2105-9-101
22. Kawashima S, Pokarowski P, Pokarowska M, Kolinski A, Katayama T, Kanehisa M. AAindex: Amino Acid Index Database, Progress Report 2008. *Nucleic Acids Res* (2008) 36(Database issue):D202–205. doi: 10.1093/nar/gkm998
23. Kawashima S, Kanehisa M. AAindex: Amino Acid Index Database. *Nucleic Acids Res* (2000) 28(1):374. doi: 10.1093/nar/28.1.374
24. Charoenkwan P, Nantasenamat C, Hasan MM, Shoombuatong W. Meta-iPVP: A Sequence-Based Meta-Predictor for Improving the Prediction of Phage Virion Proteins Using Effective Feature Representation. *J Comput Aided Mol Des* (2020) 34(10):1105–16. doi: 10.1007/s10822-020-00323-z
25. Cheng X, Xiao X, Chou KC. Ploc_Bal-Mgneg: Predict Subcellular Localization of Gram-Negative Bacterial Proteins by Quasi-Balancing Training Dataset and General PseAAC. *J Theor Biol* (2018) 458:92–102. doi: 10.1016/j.jtbi.2018.09.005
26. Chou KC. Some Remarks on Protein Attribute Prediction and Pseudo Amino Acid Composition. *J Theor Biol* (2011) 273(1):236–47. doi: 10.1016/j.jtbi.2010.12.024
27. Hasan MM, Khatun MS, Kurata H. iLBE for Computational Identification of Linear B-Cell Epitopes by Integrating Sequence and Evolutionary Features. *Genomics Proteomics Bioinf* (2020) 18(5):593–600. doi: 10.1016/j.gpb.2019.04.004
28. Khatun MS, Hasan MM, Kurata H. PreAIP: Computational Prediction of Anti-Inflammatory Peptides by Integrating Multiple Complementary Features. *Front Genet* (2019) 10:129. doi: 10.3389/fgene.2019.00129
29. Koziol JA. On Maximally Selected Chi-Square Statistics. *Biometrics* (1991) 47 (4):1557–61. doi: 10.2307/2532406
30. McHugh ML. The Chi-Square Test of Independence. *Biochem Med (Zagreb)* (2013) 23(2):143–9. doi: 10.11613/bm.2013.018
31. Tung CW. PupDB: A Database of Pupylated Proteins. *BMC Bioinf* (2012) 13 (1):40. doi: 10.1186/1471-2105-13-40
32. Hasan MAM, Ahmad S. Mlysptmpred: Multiple Lysine PTM Site Prediction Using Combination of SVM With Resolving Data Imbalance Issue. *Natural Sci* (2018) 10(09):370–84. doi: 10.4236/ns.2018.109035
33. Wang P, Huang X, Qiu W, Xiao X. Identifying GPCR-Drug Interaction Based on Wordbook Learning From Sequences. *BMC Bioinf* (2020) 21(1):150. doi: 10.1186/s12859-020-3488-8
34. Qiu W, Lv Z, Hong Y, Jia J, Xiao X. BOW-GBDT: A GBDT Classifier Combining With Artificial Neural Network for Identifying GPCR-Drug Interaction Based on Wordbook Learning From Sequences. *Front Cell Dev Biol* (2020) 8:623858(1789). doi: 10.3389/fcell.2020.623858
35. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed Representations of Words and Phrases and Their Compositionality. *Adv Neural Inf Process Syst* (2013) 3111–9. doi: 10.48550/arXiv.1301.3781
36. Bottou L. "Large-Scale Machine Learning With Stochastic Gradient Descent,". In: *Proceedings of COMPSTAT'2010*. Springer (2010). p. 177–86.
37. Rodríguez P, Bautista MA, Gonzàlez J, Escalera S. Beyond One-Hot Encoding: Lower Dimensional Target Embedding. *Image Vision Computing* (2018) 75:21–31. doi: 10.1016/j.imavis.2018.04.004
38. Bian H, Guo M, Wang J. Recognition of Mitochondrial Proteins in Plasmodium Based on the Tripeptide Composition. *Front Cell Dev Biol* (2020) 8:578901(875). doi: 10.3389/fcell.2020.578901
39. Chou KC. Prediction of Protein Cellular Attributes Using Pseudo-Amino Acid Composition. *Proteins: Structure Function Genet* (2001) 44(1):60–0. doi: 10.1002/prot.1072
40. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic Minority Over-Sampling Technique. *J Artif Intell Res* (2002) 16:321–57. doi: 10.1613/jair.953
41. Pandis N. The Chi-Square Test. *Am J Orthod Dentofacial Orthop* (2016) 150 (5):898–9. doi: 10.1016/j.ajodo.2016.08.009
42. Sharpe D. Chi-Square Test Is Statistically Significant: Now What? *Pract Assessment Res Eval* (2015) 20(1):8. doi: 10.7275/tbfa-x148
43. Manavalan B, Shin TH, Kim MO, Lee G. PIP-EL: A New Ensemble Learning Method for Improved Proinflammatory Peptide Predictions. *Front Immunol* (2018) 9:1783. doi: 10.3389/fimmu.2018.01783
44. Su R, Hu J, Zou Q, Manavalan B, Wei L. Empirical Comparison and Analysis of Web-Based Cell-Penetrating Peptide Prediction Tools. *Brief Bioinform* (2020) 21(2):408–20. doi: 10.1093/bib/bby124
45. Shoombuatong W, Schaduangrat N, Pratiwi R, Nantasenamat C. THPep: A Machine Learning-Based Approach for Predicting Tumor Homing Peptides. *Comput Biol Chem* (2019) 80:441–51. doi: 10.1016/j.compbiolchem.2019.05.008
46. Schaduangrat N, Nantasenamat C, Prachayasittikul V, Shoombuatong W. Meta-iAVP: A Sequence-Based Meta-Predictor for Improving the Prediction of Antiviral Peptides Using Effective Feature Representation. *Int J Mol Sci* (2019) 20(22):5743–67. doi: 10.3390/ijms20225743
47. Win TS, Malik AA, Prachayasittikul V, JE SW, Nantasenamat C, Shoombuatong W. HemoPred: A Web Server for Predicting the Hemolytic

Activity of Peptides. *Future Med Chem* (2017) 9(3):275–91. doi: 10.4155/fmc-2016-0188

48. Centor RM. Signal Detectability: The Use of ROC Curves and Their Analyses. *Med Decis Making* (1991) 11(2):102–6. doi: 10.1177/0272989X9101100205
49. Jiménez-Valverde A. Insights Into the Area Under the Receiver Operating Characteristic Curve (AUC) as a Discrimination Measure in Species Distribution Modelling. *Global Ecol Biogeogr* (2012) 21(4):498–507. doi: 10.1111/j.1466-8238.2011.00683.x
50. Cui D, Curry D. Prediction in Marketing Using the Support Vector Machine. *Marketing Sci* (2005) 24(4):595–615. doi: 10.1287/mksc.1050.0123
51. Cai CZ, Han LY, Ji ZL, Chen X, Chen YZ. SVM-Prot: Web-Based Support Vector Machine Software for Functional Classification of a Protein From Its Primary Sequence. *Nucleic Acids Res* (2003) 31(13):3692–7. doi: 10.1093/nar/gkg600
52. Tong S, Chang E. Support Vector Machine Active Learning for Image Retrieval. *Proc Ninth ACM Int Conf Multimed* (2001) 107–18. doi: 10.1145/500141.500159
53. Zavaljevski N, Stevens FJ, Reifman J. Support Vector Machines With Selective Kernel Scaling for Protein Classification and Identification of Key Amino Acid Positions. *Bioinformatics* (2002) 18(5):689–96. doi: 10.1093/bioinformatics/18.5.689
54. Gordon AD, Breiman L, Friedman JH, Olshen RA, Stone CJ. Classification and Regression Trees. *Biometrics* (1984) 40(3):358. doi: 10.2307/2530946
55. Noble WS. What Is a Support Vector Machine? *Nat Biotechnol* (2006) 24(12):1565–7. doi: 10.1038/nbt1206-1565
56. Gao J, Thelen JJ, Dunker AK, Xu D. Musite, a Tool for Global Prediction of General and Kinase-Specific Phosphorylation Sites. *Mol Cell Proteomics* (2010) 9(12):2586–600. doi: 10.1074/mcp.M110.001388
57. Kowalski BR, Bender CF. K-Nearest Neighbor Classification Rule (Pattern Recognition) Applied to Nuclear Magnetic Resonance Spectral Interpretation. *Analytical Chem* (2002) 44(8):1405–11. doi: 10.1021/ac60316a008
58. Chen T, He T, Benesty M, Khotilovich V, Tang Y, Cho H. Xgboost: Extreme Gradient Boosting. *R Package version 0.4-2* (2015) 1(4):1–4. doi: 10.1145/2939672.2939785
59. Friedman JH. Greedy Function Approximation: A Gradient Boosting Machine. *Ann Stat* (2001) 29(5):1189–232. doi: 10.2307/2699986
60. Simopoulos CMA, Weretilnyk EA, Golding GB. Prediction of Plant lncRNA by Ensemble Machine Learning Classifiers. *BMC Genomics* (2018) 19(1):316. doi: 10.1186/s12864-018-4665-2
61. Galar M, Fernandez A, Barrenechea E, Bustince H, Herrera F. A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Trans Syst Man Cybernetics Part C (Applications Reviews)* (2012) 42(4):463–84. doi: 10.1109/tsmcc.2011.2161285
62. Svetnik V, Wang T, Tong C, Liaw A, Sheridan RP, Song Q. Boosting: An Ensemble Learning Tool for Compound Classification and QSAR Modeling. *J Chem Inf Model* (2005) 45(3):786–99. doi: 10.1021/ci0500379
63. Agarwal S, Chowdary CR. A-Stacking and A-Bagging: Adaptive Versions of Ensemble Learning Algorithms for Spoof Fingerprint Detection. *Expert Syst Appl* (2020) 146:113160. doi: 10.1016/j.eswa.2019.113160

**Conflict of Interest:** The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

**Publisher's Note:** All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.