Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 18(6): 9132–9147.
DOI: 10.3934/mbe.2021450
Received: 06 September 2021
Accepted: 07 October 2021
Published: 25 October 2021

Research article

Predicting S-nitrosylation proteins and sites by fusing multiple features

Wang-Ren Qiu*, Qian-Kun Wang, Meng-Yue Guan, Jian-Hua Jia and Xuan Xiao*

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

* **Correspondence:** Email: qiuone@163.com, jdzxiaoxuan@163.com.

Abstract: Protein S-nitrosylation is one of the most important post-translational modifications, a well-grounded understanding of S-nitrosylation is very significant since it plays a key role in a variety of biological processes. For an uncharacterized protein sequence, it is a very meaningful problem for both basic research and drug development when we can firstly identify whether it is a S-nitrosylation protein or not, and then predict the specific S-nitrosylation site(s). This work has proposed two models for identifying S-nitrosylation protein and its PTM sites. Firstly, three kinds of features are extracted from protein sequence: KNN scoring of functional domain annotation, PseAAC and bag-of-words based on the physical and chemical properties of amino acids. Secondly, the synthetic minority oversampling technique is used to balance the data sets, and some state-of-the-art classifiers and feature fusion strategies are performed on the balanced data sets. In the five-fold cross-validation for predicting S-nitrosylation proteins, the results of Accuracy (ACC), Matthew's correlation coefficient (MCC) and area under ROC curve (AUC) are 81.84%, 0.5178, 0.8635, respectively. Finally, a model for predicting S-nitrosylation sites has been constructed on the basis of tripeptide composition (TPC) and the composition of k-spaced amino acid pairs (CKSAAP). To eliminate redundant information and improve work efficiency, elastic nets are employed for feature selection. The five-fold cross-validation tests have indicated the promising success rates of the proposed model. For the convenience of related researchers, the web-server named "RF-SNOPS" has been established at http://www.jci-bioinfo.cn/RF-SNOPS

Keywords: S-nitrosylation; random forest; post-translational modification; multiple features; identification

1. Introduction

Protein post-translational modification is an important chemical process, which plays a key role in regulating cell functions [1] and also changes the physical and chemical properties of protein. More than 400 post-translational modifications including methylation [2], acetylation [3], phosphorylation [4], and S-nitrosylation (SNO) [5] have been discovered so far. As SNO is a reversible post-translational modification of proteins, a large number of studies have shown that SNO plays an important role in multiple biological processes such as redox signal transduction [6], cell signal transduction [7], cell senescence [8], and transcription [9]. SNO is also related to many human diseases such as cancer [10], Alzheimer's disease [11], and chronic renal failure [12]. Therefore, a well-grounded understanding of SNO is of great significance for the study of basic biological processes [9,13] and the development of drugs [14]. In recent years, many SNO sites have been identified through molecular signals [15,16], but identification of SNO sites still faces some challenges including low accuracy, time-consuming and labor-intensive. With the continuous development of computer technology, a large number of computational models have been used to predict the specific sites of SNO modification.

Many post-translational modifications of proteins have been detected by a variety of computational models. Qiu identified phosphorylated [17] and acetylated [18] proteins with GO notations. GPS-SNO [19], SNOSite [20], iSNO-PseAAC [21], PreSNO [5] and RecSNO [22] have been applied to the prediction of SNO sites. The GPS-SNO, SNOSite and iSNO-PseAAC models use relatively small data sets. In addition, many negative samples in these data sets are now experimentally verified as positive samples. The data sets used by PreSNO and RecSNO are relatively large and new, but there is still room for improvement in the performance of the model.

On the basis of previous research, this work established two models for predicting SNO proteins and sites. In predicting SNO proteins, a bag-of-words model has been proposed on the basis of KNN scoring matrix obtained from proteins' GO annotation information [18], PseAAC [23,24] of amino acids sequence. Fusion of multiple features can more comprehensively reflect the information of the protein sequence and improve the prediction results. A combination of oversampling technique and random deletion method are applied to balance the training set since the issue is involved in imbalanced data sets. In predicting SNO sites, two feature extraction methods, TPC [25] and CKSAAP [26], are used to extract the features of protein sequence fragments. In order to eliminate the redundancy and noise information of the original feature space, elastic nets [27] are used to reduce the dimensionality of the feature space after the fusion strategy is performed on the original features. Random Forest severed as the classifiers and be verified with 5-fold cross-validation. The specific flow chart is shown in Figure 1.

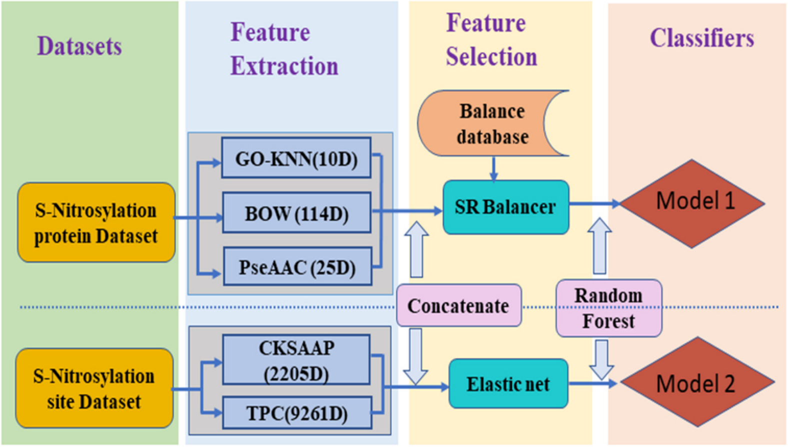

Figure 1. The framework of RF-SNOPS.

2. Materials

2.1. Datasets for prediction SNO proteins

To obtain a scientifical prediction result, a strict benchmark data set is essential. The UniProKB has been accepted by most bioinformatics researchers. Here, the negative samples are extracted from the UniProKB and the positive sample are extracted from Xie's [28], which is a high-quality data set based on extensive literature research. The protein sequence can be expressed as:

$$P = R_1R_2R_3 \cdots R_i \cdots R_L \tag{1}$$

where R_i represents the i-th amino acid residue, and L represents the length of the protein sequence.

In order to identify SNO proteins, we constructed a benchmark data set similar to dataset of Hasan et al. [5], which consists of 3113 SNO proteins. Every one of the positive samples, i.e., SNO proteins, has at least one SNO site. For negative samples, we randomly selected 18,047 proteins without any SNO site from the UniProKB. In order to make the results more rigorous, the CD-HIT was used to remove 30% of the 3113 positive samples and 18, 047 negative samples. Finally, 2192 positive samples and 7809 negative samples are collected in the proposed benchmark data set.

2.2. Datasets for predicting SNO Sites

The benchmark data set for predicting SNO sites are the same as Hasan et al. [5], which consists of 3383 positive samples and 3365 negative samples. A potential SNO(C) site-containing peptide sample can be generally expressed by

$$\overline{\mathbf{P}_\xi} = R_{-\xi}R_{-(\xi-1)} \cdots R_{-2}R_{-1}\ C\ R_{+1}R_{+2} \cdots R_{+(\xi-1)}R_{+\xi} \tag{2}$$

where the subscript ξ is an integer, $R_{-\xi}$ represents the ξ-th upstream amino acid residue from the center, the $R_{+\xi}$ the ξ-th downstream amino acid residue, and so forth. If the number of left or right residues of the center C is less than ξ, then the pseudo amino acid "X" would be used to supplement the sequence. The $(2\xi + 1)$-tuple peptide sample $\overline{\mathbf{P}_\xi}$ can be further classified into the following two categories:

$$\overline{\mathbf{P}_\xi} \in \begin{cases} \overline{\mathbf{P}}_\xi^+, & \text{if its center is a SON site} \\ \overline{\mathbf{P}}_\xi^-, & \text{other wise} \end{cases} \tag{3}$$

where $\overline{\mathbf{P}}_\xi^+$ denotes a true SNO segment C with at its center, $\overline{\mathbf{P}}_\xi^-$ a corresponding false SNO segment, and the symbol $\in$ means "a member of" in the set theory.

3. Feature extraction

3.1. Feature extraction methods for predicting SNO proteins

3.1.1. GO-KNN

GO-KNN [18] features were extracted on the basis of the GO annotations of proteins. In this work, we need to find out the GO terms of all protein sequences and calculate the distance between proteins. Take protein P_1 as an example, for anyone of other proteins, for example, P_2, then their GO terms can be listed as $P_{GO}^1 = \{GO_1^1, GO_2^1, \cdots, GO_M^1\}$ and $P_{GO}^2 = \{GO_1^2, GO_2^2, \cdots, GO_N^2\}$ are obtained. If there is no GO term for a protein, we will replace it with GO terms of its homologous protein. The distance between two proteins can be calculated with Eq (4):

$$Distance(P_1, P_2) = 1 - \frac{\lfloor P_{GO}^1 \cap P_{GO}^2 \rfloor}{\lfloor P_{GO}^1 \cup P_{GO}^2 \rfloor} \tag{4}$$

where, GO_i^1 and GO_i^2 represent the i-th GO of P_1 and P_2, respectively. The M and N represent the numbers of GO, respectively, $\cup$ and $\cap$ are the union and intersection in the set theory, and $\lfloor\ \rfloor$ represents the number of elements in the set. Then, the GO-KNN features could be extracted according to the following steps: 1) Sorting the calculated distances in ascending order; 2) Selecting the first k near neighbors of the test protein; 3) Calculating the percentage of positive samples in the k neighbors. In this study, k were selected as 2, 4, 8, 16, 62, 64, 128, 256, 512, 1024. In this way, a 10-dimensional feature vector $(x_1, x_2, \cdots, x_{10})$ could represents the protein P_1.

3.1.2. BOW

A bag-of-words model [29] based on the physical and chemical properties of protein has been used in identifying GPCR-drug interaction. The main steps are listed as follows: 1) Encoding the protein sequence with its physical and chemical properties. Up to now, scientists have obtained various physical and chemical properties of 20 common amino acids [30]. After careful experimental comparison, hydrophilicity was selected as an indicator for the proposed model. 2) Designing wordbooks for protein. When the window sizes are 1, 2 and 3, and the step size of the moving window is 1, the coding sequence is divided into segments of different lengths. Segments of length 1 form wordbook WB_1, segments of length 2 form wordbook WB_2, and segments of length 3 form wordbook WB_3. When the window size is 2, the step size of moving the window is still 1. But the window at this time is different from the above, it is separated by an amino acid. At this time, the coding sequence is divided into fragments of length 2, and these fragments form the wordbook WB_4. 3) Clustering the word books. We divided the words in the wordbook WB_1 into 20 sub-groups according to the types of amino acids. Words in WB_2, WB_3 and WB_4 were clustered with K-means algorithm. The numbers of clusters were 16, 62 and 16. 4) Calculating the ratio of the number of each amino acid to the total number of words in the vocabulary with Eq (5).

$$X_i^{WB_j} = \frac{x_i^{WB_j}}{N} \quad i = 1, \dots, K \quad j = 1,2,3,4 \tag{5}$$

here, K is the number of clusters in the wordbook WB_j, $X_i^{WB_j}$ is the number of words in the i-th category of the wordbook WB_j, and N is the total number of words in the wordbook WB_j. Then a 114-D feature vector was formed for a given protein sequence, i.e. $\left(X_1^{WB_1}, \dots, X_{20}^{WB_1}, X_1^{WB_2}, \dots, X_{16}^{WB_2}, X_1^{WB_3}, \dots X_{62}^{WB_3}, X_1^{WB_4} \dots, X_{16}^{WB_4}\right)$.

3.1.3. PseAAC

PseAAC [23,24] is a very popular feature for bioinformatics. In this work, six physical and chemical properties of hydrophobicity, hydrophilicity, molecular side chain mass, PK1, PK2 and PI are used. We first used Eq (6) to transform the original physical and chemical properties of amino acids:

$$W_a(i) = \frac{W_a^0(i) - \sum_{i=1}^{20} \frac{W_a^0(i)}{20}}{\sqrt{\frac{\sum_{i=1}^{20}\left[W_a^0(i) - \sum_{i=1}^{20} \frac{W_a^0(i)}{20}\right]^2}{20}}} \tag{6}$$

where, $a \in \{1,2,\cdots,6\}$ and $i \in \{1,2,\cdots,20\}$. $W_a^0(i)$ represents the value of the ath original physical and chemical properties of the ith amino acid. We substitute the values of the transformed physical and chemical properties with Eq (7):

$$\Theta(R_i, R_j) = \frac{1}{6}\sum_{a=1}^{6}\left[W_a(R_j) - W_a(R_i)\right]^2 \tag{7}$$

where, $W_1(R_j)$represents the value of hydrophobicity of R_j. By analogy, $W_6(R_i)$ represents the PI value of R_i. Then the correlation factor of each layer can be obtained by using the Eq (8):

$$\theta_\lambda = \frac{1}{L-\lambda}\sum_{i=1}^{L-\lambda} \Theta(R_i, R_{i+\lambda}) \quad \lambda < \mathrm{L} \tag{8}$$

where θ_λ represents the correlation factor of the λ-th layer of the protein sequence. Finally, the protein sequence is converted into a feature by Eq (9):

$$x_i = \begin{cases} \frac{f_i}{\sum_{i=1}^{20} f_i + \omega \sum_{j=1}^{\lambda} \theta_j} (1 \le i \le 20) \\ \\ \frac{\omega \theta_{i-20}}{\sum_{i=1}^{20} f_i + \omega \sum_{j=1}^{\lambda} \theta_j} (20 + 1 \le i \le 20 + \lambda) \end{cases} \tag{9}$$

here, f_i represents the frequency of the i-th amino acid, ω is 0.5, and λ is 5. In this way, a 25-dimensional feature vector is formed.

3.2. Data balancing methods for predicting SNO protein

In order to reduce the adverse effect of unbalanced data on the performance of the model, many methods for dealing with unbalanced data have been proposed, such as Synthetic Minority Oversampling Technique [31] (SMOTE) and Random Under Sampler [32] (RUS). SMOTE is a method proposed by Chawla et al. It has been used to predict protein sites [27] and improve the prognostic assessment of lung cancer [33]. RUS is a very simple and popular method of under-sampling. It can be used in pediatric pneumonia detection [34] and convolutional neural network performance improvement issues [35]. In this study, we combined these two methods to process the data. SMOTE is used to oversample the positive samples, and RUS is used to under-sample the negative samples. In the end, the number of processed positive samples is equal to negative samples. The specific process is shown in Figure 2.

3.3. Feature extraction methods for predicting SNO sites

3.3.1. CKSAAP

CSKAAP [25] has been widely used in protein site prediction [26] since it can effectively express internal laws for a given protein sequence. The protein fragment is composed of 20 common amino acids and a pseudo amino acid, which contains 441 residue pairs (AA, AC, ..., XX) for each l. Here l represents the space between each residue pair. The following formula is used to calculate the characteristics of the fragment:

$$\left(\frac{N_{AA}}{N_T},\frac{N_{AC}}{N_T},\frac{N_{AD}}{N_T},\cdots,\frac{N_{XX}}{N_T}\right)_{441} \tag{10}$$

here $N_{AA}, N_{AC}, \cdots$ represent the number of times the corresponding amino acid pair appears in the fragment, L is the length of the protein fragment. $N_T = L - l - 1$. In this study, the values of l are 0, 1, 2, 3, 4, and the corresponding N_T are 40, 39, 38, 37 and 36, respectively. Then, a 2205-D feature vector is formed.

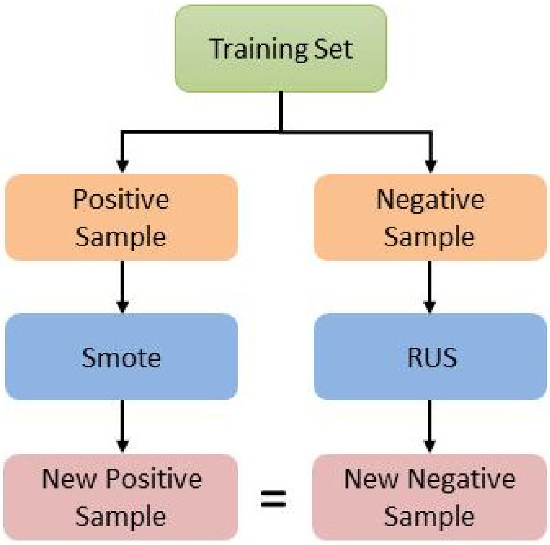

Figure 2. Balance database processing.

3.3.2. TPC

Based on the structural properties of proteins, researchers have proposed the tripeptide composition (TPC). It has been used to predict protein subcellular localization [36] identification of plasmodium mitochondrial proteins [37]. TPC calculates the frequency of three consecutive amino acids, and then a protein fragment can be represented by a 9261-dimensional vector.

$$P_i = \frac{N_i}{\sum_1^{9261} N_i} \tag{11}$$

where N_i represents the total number of i-th in 9261 tripeptides.

3.4. Feature selection for predicting SNO sites

The elastic net proposed by Zou and Hastie [38] is an effective feature selection method. By introducing the L_1, L_2 norm into a simple linear regression model, the elastic net can not only perform continuous shrinkage and automatically select variables at the same time, but also predict related variables. At present, elastic nets have been widely used in protein site prediction [27,39] and achieved good results.

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 models. They are accuracy (ACC), sensitivity (SN), specificity (SP) and Matthews correlation coefficient (MCC) [40], which are defined by Eq (12):

$$\begin{cases} Sn=\frac{TP}{TP+FN} \\ Sp=\frac{TN}{TN+FP} \\ ACC=\frac{TP+TN}{TP+FP+TN+FN} \\ MCC=\frac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)\times(TP+FN)\times(TN+FP)\times(TN+FN)}} \end{cases} \tag{12}$$

In predicting SNO proteins, TP indicates the number of proteins that are predicted to have SNO sites and actually have SNO sites, and TN is the number of proteins that are predicted to have no SNO sites that are actually not have SNO sites. FP is the number of proteins without SNO sites but predicted to have SNO sites, FN is the number of proteins with SNO sites but predicted to have no SNO sites. In addition, the area under the ROC curve AUC is also used to evaluate this model.

In predicting SNO sites, TP indicates the number of actual SNO sites predicted to be SNO sites, and TN indicates the number of non-SNO sites predicted to be not SNO sites. FP is the number of non-SNO sites predicted to be SNO sites, and FN is the number of actual SNO sites predicted to be non-SNO sites.

4.2. Operation engine

4.2.1. Random Forest

Random Forest [41] is an algorithm that integrates multiple trees through the idea of ensemble learning. Its basic unit is decision tree. As a highly flexible machine learning algorithm, Random Forest (RF) has been widely used in data analysis [42], biological information [43] and technological development [44].

4.2.2. Naive Bayes

Naive Bayes (NB) [45] is a simple and effective classifier, which is widely used in software defect prediction [46], medical diagnosis [47] and biological information [48]. NB is based on the Bayes theorem and the assumption of the conditional independence of features, which greatly reduces the complexity of the classification algorithm.

4.2.3. K Nearest Neighbor

K Nearest Neighbor (KNN) [49] is one of the supervised machine learning algorithms, which is widely used in face recognition [50] , disease research [51] and engineering applications [52]. Its main idea is to judge the category of the predicted value based on the category of the k points closest to the predicted value.

4.2.4. eXtreme Gradient Boosting

XGBoost [53,54] is an improved algorithm for boosting based on GBDT [55]. XGBoost is an integrated lifting algorithm that integrates many basic models to form a strong model. Because of its advantages such as good prediction effect and high training efficiency, XGBoost has been widely used in the field of data analysis.

5. Results and discussion

5.1. Results and discussion for SNO proteins prediction

5.1.1. Effect of features

In this research, GO-KNN, BOW and PseAAC three kinds of feature extraction methods were used to encode the protein sequence, and obtained 10-D, 114-D and 25-D feature vectors, respectively. These three kinds of features were fused into a 149-D feature vector ALL. Through the 5-fold cross-validation, the prediction results obtained by different feature extraction are shown in Table 1.

Table 1. The predict results of different features.

Feature	Acc (%)	Sn (%)	Sp (%)	MCC	AUC
GO-KNN	82.72	48.36	92.37	0.4533	0.8521
BOW	78.83	13.04	97.30	0.1969	0.7359
PseAAC	79.38	19.43	96.22	0.2503	0.7616
ALL	83.77	49.49	93.40	0.4840	0.8593

It can be seen from Table 1 that different features obtained varied prediction results. Among the three methods, GO-KNN has the highest ACC, Sn, MCC and AUC, of which are 82.72%, 48.36%, 0.4533 and 0.8521 respectively. The ACC, Sn, MCC and AUC of BOW are the lowest, of which are 78.83%, 13.04%, 0.1969 and 0.7359, respectively. But the Sp of BOW is 97.30%, which is the highest. After combining these three characteristics, ACC, Sn, Sp, MCC, AUC are 83.77%, 49.49%, 93.40%, 0.4840, 0.8593, respectively. Among them, ACC, Sn, MCC and AUC are all higher than those produced by GO-KNN. The results show that multi-feature fusion can improve a number of indicators. In order to better analyze the influence of different features on the prediction of SNO proteins, the prediction results obtained from the three features and their fusion features are shown in Figure 3.

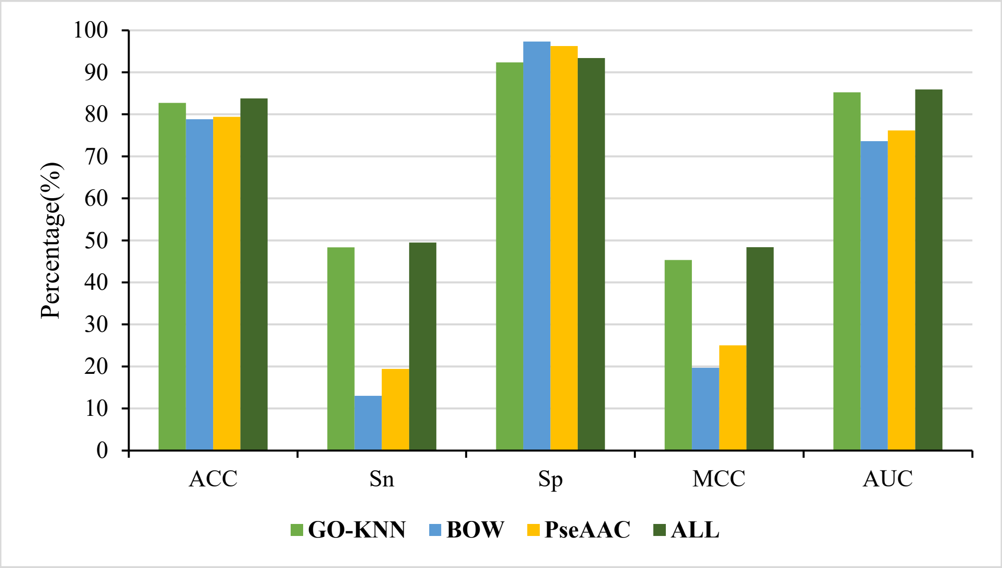

Figure 3. Comparison of prediction results on different features.

It can be seen from Figure 3 that the three features and their fusion affect the five evaluation indicators to some extent. They are less effective on Sn and MCC, and better on ACC, Sp and AUC. Comparing these four feature codes, the ACC, Sn, MCC and AUC of the fusion feature ALL are improved. Multi-feature fusion can reflect sequence information more comprehensively, thereby improving prediction ability. Therefore, multi-feature fusion can be used to predict SNO proteins.

5.1.2. Effect of the SR balancer

Here, SMOTE and RUS are denoted as SR balancer. We input the pre-balanced and post-balanced data sets into the model, and passed the 5-fold cross-validation to obtain the prediction results of ACC, Sn, Sp, MCC, AUC on the balanced and unbalanced data sets, as shown in the Table 2.

Table 2. Comparison of predict results before and after SR Balancer.

	Acc (%)	Sn (%)	Sp (%)	MCC	AUC
Imbalance	83.77	49.49	93.40	0.4840	0.8593
Balance	81.84	70.82	84.93	0.5178	0.8635

It can be seen from Table 2 that the balanced Sn and Sp are relatively balanced. In addition, Sn, MCC and AUC have improved. Therefore, in summary, it is very necessary to balance the dataset.

5.1.3. Effect of classifiers

Classifiers play an important role in model prediction. This work used the above four classifiers to identify SNO proteins. After 5-fold cross-validation, the results of each classifier for ACC, Sn, Sp, MCC and AUC are shown in Table 3. It can be seen from Table 3 that the effect of random forest on various evaluation indicators is the best. In order to better compare the effects of different classifiers, the prediction results of the four classifiers are shown in Figure 4.

Table 3. The prediction results of different classifiers.

Algorithms	Acc (%)	Sn (%)	Sp (%)	MCC	AUC (%)
RF	81.84	70.82	84.93	0.5178	0.8635
NB	63.81	78.37	59.73	0.3154	0.7710
KNN	71.97	83.44	68.75	0.4366	0.8360
XGBoost	80.73	70.07	83.72	0.4953	0.8553

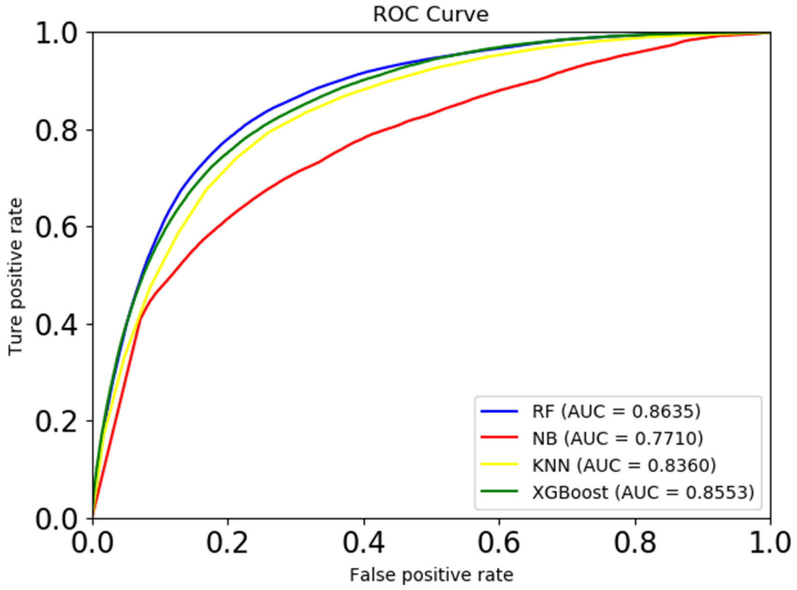

Figure 4. The ROC curves of different classification methods.

The area under the ROC curve can evaluate the predictive performance of the model. It can be seen from Figure 4 that when the random forest is used as a classifier, the area under the ROC curve is the largest. Therefore, random forest is the best choice for the proposed model.

5.2. Results and discussion for SNO sites prediction

5.2.1. Effect of features

In this study, two kinds of features, CKSAAP and TPC, were used, and the 2205-dimension and 9261-dimension feature vectors were obtained on the basis of above-mentioned algorithms. In order to better reflect the information of protein fragments, these features are fused into a 11,466-dimension feature vector. Through 5-fold cross-validation, the prediction results obtained by different feature extraction are shown in Table 4.

Table 4. The prediction results of different feature extraction methods.

Feature	Acc (%)	Sn (%)	Sp (%)	MCC	AUC
CKSAAP	73.97	83.67	64.27	0.4891	0.8036
TPC	71.38	66.07	76.74	0.4305	0.8069
ALL	75.36	86.39	64.31	0.5201	0.8196

It can be seen from Table 4 that the ACC, Sn and MCC of CKSAAP are higher than those of TPC. TPC performs better than CKSAAP on Sp and AUC. After feature fusion, Acc, Sn, MCC and AUC are all higher than single feature. Therefore, feature fusion is necessary for this issue.

5.2.2. Effect of elastic net

Multi-information fusion can more comprehensively extract protein sequence information, but redundancy and noise information will also be generated. The dimensionality reduction method can not only retain important features, but the computational efficiency of the model will also be improved. In this paper, elastic net was used to reduce the dimensionality of the fused feature data set, and obtain the feature subset of 704. After 5-fold cross-validation, the prediction results of Random Forest are shown in Table 5.

The features after dimensionality reduction using elastic nets, except for Sn, all other evaluation indicators have been improved. In addition, because the feature dimension is greatly reduced after dimensionality reduction, the efficiency of the model is also significantly improved.

Table 5. Results before and after feature selection.

	Acc (%)	Sn (%)	Sp (%)	MCC	AUC
All	75.36	86.39	64.31	0.5201	0.8196
Elastic net	76.02	85.68	66.33	0.5304	0.8260

5.2.3. Effect of different classifiers

Four kinds of classifiers, Random Forest, Naive Bayes, K-Nearest Neighbor and XGBoost, were tested in this work for predicting SNO sites. After 5-fold cross-validation, the results were shown in Table 6. From Table 6 we can get that Naive Bayes and K-Nearest Neighbors are relatively inferior. Except for Sp, all indicators of Random Forest were the best. In order to evaluate the performance of the classifier more comprehensively, the ROC curves of different classifiers are shown in Figure 5.

From Figure 5, we can clearly see that the area under the ROC curve of the random forest is the largest. Therefore, random forest has been selected as the classifier of the proposed model.

Table 6. The prediction results of different classifiers.

Algorithms	Acc (%)	Sn (%)	Sp (%)	MCC	AUC (%)
RF	76.02	85.68	66.33	0.5304	0.8260
NB	69.74	79.46	59.98	0.4022	0.7605
KNN	63.63	46.39	81.00	0.2923	0.7246
XGBoost	72.88	74.37	71.40	0.4580	0.8015

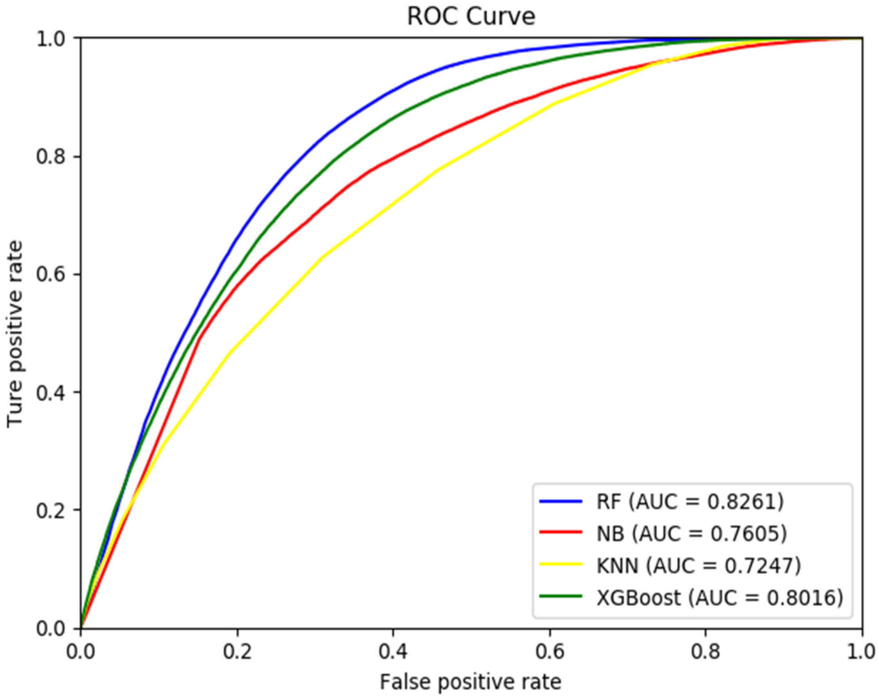

Figure 5. The ROC curves of different classification methods.

5.2.4. Comparison with other methods

To further evaluate the performance of this model, and we compared it with the PreSNO and RecSNO models. The prediction results of three different methods for the same data set are shown in Table 7. From Table 7, we can see that the ACC, Sn and MCC models of this model are all the highest. In addition, the Sp and AUC of the model in this paper also have good results. Therefore, the performance of this model is better than PreSNO and RecSNO.

Table 7. Comparison of the RF-SNOPS with other methods.

Feature	Acc (%)	Sn (%)	Sp (%)	MCC	AUC
PreSNO	70%	54%	86%	0.42	0.84
RecSNO	72%	79%	66%	0.45	0.79
RF-SNOPS	76.02	85.68	66.33	0.5304	0.8260

6. Conclusions

In order to identify SNO proteins, we used GO-KNN, BOW and PseAAC to extract the sequence information. GO-KNN extracted KNN neighbor information based on protein GO information, and BOW and PseAAC extracted protein sequence information based on physical and chemical properties. In addition, we used the SR balancer to process the unbalanced data set, reduce the negative impact of the unbalance on the model. Finally, Random Forest was used to make predictions. For predicting SNO sites, CKSAAP and TPC were used to extract protein fragment information. In order to improve the computational efficiency and eliminate the redundancy and noise generated by the fusion features, we used elastic nets to reduce the dimensionality of the fusion features. These processes only need to require calculation models without any physical and chemical experiments, which can save experimental costs and improves work efficiency. We hope that this work will be helpful for solving biological problems with computational methods.

Acknowledgments

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

Conflict of interest

The authors have declared that no competing interest exists.

References

1. I. Gusarov, E. Nudler, Protein S-nitrosylation: enzymatically controlled, but intrinsically unstable, post-translational modification, *Mol. Cell*, **69** (2018), 351–353.
2. W. Deng, Y. Wang, L. Ma, Y. Zhang, S. Ullah, Y. Xue, Computational prediction of methylation types of covalently modified lysine and arginine residues in proteins, *Briefings Bioinf.*, **18** (2016), 647–658.
3. L. Kiemer, J. D. Bendtsen, N. Blom, NetAcet: prediction of N-terminal acetylation sites, *Bioinformatics*, **21** (2005), 1269–1270.
4. Y. D. Khan, N. Rasool, W. Hussain, S. A. Khan, K. C. Chou, iPhosT-PseAAC: identify phosphothreonine sites by incorporating sequence statistical moments into PseAAC, *Anal. Biochem.*, **550** (2018), 109–116.

5. M. M. Hasan, B. Manavalan, M. S. Khatun, H. Kurata, Prediction of S-nitrosylation sites by integrating support vector machines and Random Forest, *Mol. Omics*, **15** (2019), 451–458.
6. V. Fernando, X. Zheng, Y. Walia, V. Sharma, J. Letson, S. Furuta, S-Nitrosylation: an emerging paradigm of redox signaling, *Antioxid. (Basel)*, **8** (2019), 404.
7. H. Hayashi, D. T. Hess, R. Zhang, K. Sugi, H. Gao, B. L. Tan, et al., S-nitrosylation of β-arrestins biases receptor signaling and confers ligand independence, *Mol. Cell*, **70** (2018), 473–487.
8. S. Rizza, S. Cardaci, C. Montagna, G. Di Giacomo, D. De Zio, M. Bordi, et al., S-nitrosylation drives cell senescence and aging in mammals by controlling mitochondrial dynamics and mitophagy, *Proc. Natl. Acad. Sci.*, **115** (2018), 3388–3397.
9. F. Li, P. Sonveaux, Z. N. Rabbani, S. Liu, B. Yan, Q. Huang, et al., Regulation of HIF-1α stability through S-nitrosylation, *Mol. Cell*, **26** (2017), 63–74.
10. Z. Wang, *Protein* S-nitrosylation and cancer, *Cancer Lett.*, **320** (2012), 123–129.
11. T. S. Wijasa, M. Sylvester, N. Brocke-Ahmadinejad, S. Schwartz, F. Santarelli, V. Gieselmann, et al., Quantitative proteomics of synaptosome S-nitrosylation in Alzheimer's disease, *J. Neurochem.*, **152** (2020), 710–726.
12. M. Piroddi, A. Palmese, F. Pilolli, A. Amoresano, P. Pucci, C. Ronco, F. Galli, Plasma nitroproteome of kidney disease patients, *Amino Acids*, **40** (2011), 653–667.
13. A. Nott, P. M. Watson, J. D. Robinson, L. Crepaldi, A. Riccio, S-nitrosylation of histone deacetylase 2 induces chromatin remodelling in neurons, *Nature*, **455** (2008), 411–415.
14. G. Huang, J. Li, C. Zhao, Computational prediction and analysis of associations between small molecules and binding-associated S-nitrosylation sites, *Molecules*, **23** (2018), 954.
15. J. R. Burgoyne, P. Eaton, A rapid approach for the detection, quantification, and discovery of novel sulfenic acid or S-nitrosothiol modified proteins using a biotin-switch method, *Methods Enzymol.*, **473** (2010), 281–303.
16. G. Hao, B. Derakhshan, L. Shi, F. Campagne, S. S. Gross, SNOSID, a proteomic method for identification of cysteine S-nitrosylation sites in complex protein mixtures, *Proc. Natl. Acad. Sci. U. S. A.*, **103** (2006), 1012–1017.
17. W. R. Qiu, B. Q. Sun, X. Xiao, D. Xu, K. C. Chou, iPhos-PseEvo: identifying human phosphorylated proteins by incorporating evolutionary information into general PseAAC via grey system theory, *Mol. Inf.*, **36** (2017), 1600010.
18. W. R. Qiu, A. Xu, Z. C. Xu, C. H. Zhang, X. Xiao, Identifying acetylation protein by fusing its PseAAC and functional domain annotation, *Front. Bioeng. Biotechnol.*, **7** (2019), 311.
19. Y. Xue, Z. Liu, X. Gao, C. Jin, L. Wen, X. Yao, J. Ren, GPS-SNO: computational prediction of protein S-nitrosylation sites with a modified GPS algorithm, *PLoS One*, **5** (2010), e11290.
20. T. Y. Lee, Y. J. Chen, T. C. Lu, H. D. Huang, Y. J. Chen, SNOSite: exploiting maximal dependence decomposition to identify cysteine S-nitrosylation with substrate site specificity, *PLoS One*, **6** (2011), e21849.
21. Y. Xu, J. Ding, L. Y. Wu, K. C. Chou, iSNO-PseAAC: predict cysteine S-nitrosylation sites in proteins by incorporating position specific amino acid propensity into pseudo amino acid composition, *PLoS One*, **8** (2010), e55844.
22. A. Siraj, T. Chantsalnyam, H. Tayara, K. T. Chong, RecSNO: prediction of protein S-nitrosylation sites using a recurrent neural network, *IEEE Access*, **9** (2021), 6674–6682.

23. X. Cheng, X. Xiao, K. C. Chou, pLoc_bal-mGneg: predict subcellular localization of gram-negative bacterial proteins by quasi-balancing training dataset and general PseAAC, *J. Theor. Biol.*, **458** (2018), 92–102.
24. K. C. Chou, Some remarks on protein attribute prediction and pseudo amino acid composition, *J. Theor. Biol.*, **273** (2011), 236–247.
25. Y. Z. Chen, Y. R. Tang, Z. Y. Sheng, Z. Zhang, Prediction of mucin-type O-glycosylation sites in mammalian proteins using the composition of k-spaced amino acid pairs, *BMC Bioinf.*, **9** (2008), 101.
26. F. N. Auliah, A. N. Nilamyani, W. Shoombuatong, M. A. Alam, M. M. Hasan, H. Kurata, PUP-fuse: prediction of protein pupylation sites by integrating multiple sequence representations, *Int. J. Mol. Sci.*, **22** (2021), 2120.
27. Y. Liu, Z. Yu, C. Chen, Y. Han, B. Yu, Prediction of protein crotonylation sites through LightGBM classifier based on SMOTE and elastic net, *Anal. Biochem.*, **609** (2020), 113903.
28. Y. Xie, X. Luo, Y. Li, L. Chen, W. Ma, J. Huang, et al., DeepNitro: prediction of protein nitration and nitrosylation sites by deep learning, *Genomics, Proteomics Bioinf.*, **16** (2018), 294–306.
29. W. Qiu, Z. Lv, Y. Hong, J. Jia, X. Xiao, 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.*, **8** (2021), 623858.
30. S. Kawashima, M. Kanehisa, AAindex: amino acid index database, *Nucleic Acids Res.*, **28** (2000), 374.
31. N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, *J. Artif. Intell. Res.*, **16** (2002), 321–357.
32. M. J. Kim, D. K. Kang, H. B. Kim, Geometric mean based boosting algorithm with over-sampling to resolve data imbalance problem for bankruptcy prediction, *Expert Syst. Appl.*, **42** (2015), 1074–1082.
33. S. Yan, W. Qian, Y. Guan, B. Zheng, Improving lung cancer prognosis assessment by incorporating synthetic minority oversampling technique and score fusion method, *Med. Phys.*, **43** (2016), 2694–2703.
34. N. Habib, M. M. Hasan, M. M. Reza, M. M. Rahman, Ensemble of CheXNet and VGG-19 feature extractor with Random Forest classifier for pediatric pneumonia detection, *SN Comput. Sci.*, **1** (2020), 359.
35. K. Ghosh, A. Sarkar, A. Banerjee, S. Chatterjee. Performance improvement of convolutional neural network using random under sampling, in *Advances in Smart Communication Technology and Information Processing*, Springer, (2021), 207–217.
36. S. Hua, Z. Sun, Support vector machine approach for protein subcellular localization prediction, *Bioinformatics*, **17** (2001), 721–728.
37. H. Bian, M. Guo, J. Wang, Recognition of mitochondrial proteins in plasmodium based on the tripeptide composition, *Front. Cell Dev. Biol.*, **8** (2020), 578901.
38. H. Zou, T. Hastie, Regularization and variable selection via the elastic net, *J. R. Stat. Soc. Ser. B (Stat. Methodol.)*, **67** (2005), 301–320.
39. G. Chen, M. Cao, J. Yu, X. Guo, S. Shi, Prediction and functional analysis of prokaryote lysine acetylation site by incorporating six types of features into Chou's general PseAAC, *J. Theor. Biol.*, **461** (2019), 92–101.

40. W. R. Qiu, B. Q. Sun, X. Xiao, Z. C. Xu, J. H. Jia, K. C. Chou, iKcr-PseEns: identify lysine crotonylation sites in histone proteins with pseudo components and ensemble classifier, *Genomics*, **110** (2018), 239–246.
41. L. Breiman, Random Forests. *Mach. Learn.*, **45** (2001), 5–32.
42. H. Talebi, L. J. M. Peeters, A. Otto, R. Tolosana-Delgado, A truly spatial Random Forests algorithm for geoscience data analysis and modelling, *Math. Geosci.*, 2021.
43. Z. Qiu, Q. Liu, Protein-protein interaction site prediction using random forest proximity distance, *J. Bioinf. Comput. Biol.*, **19** (2021), 2050042.
44. S. Cabras, M. E. Castellanos, E. Staffetti, A Random Forest application to contact-state classification for robot programming by human demonstration, *Appl. Stochastic Models Bus. Ind.*, **32** (2016), 209–227.
45. N. Friedman, D. Geiger, M. Goldszmidt, Bayesian network classifiers, *Mach. Learn.*, **29** (1997), 131–163.
46. R. Queiroz, T. Berger, K. Czarnecki, Towards predicting feature defects in software product lines, in *Proceedings of the 7th International Workshop on Feature-Oriented Software Development*, Amsterdam, Netherlands, Association for Computing Machinery, (2016), 58–62.
47. H. Bohra, A. Arora, P. Gaikwad, R. Bhand, M. R. Patil, Health prediction and medical diagnosis using Naive Bayes, *Ijarcce*, **6** (2017), 32–35.
48. M. S. Ahmed, M. Shahjaman, E. Kabir, M. Kamruzzaman, Prediction of protein acetylation sites using kernel Naive Bayes classifier based on protein sequences profiling, *Bioinformation*, **14** (2018), 213–218.
49. F. Nigsch, A. Bender, B. van Buuren, J. Tissen, E. Nigsch, J. B. Mitchell, Melting point prediction employing K-Nearest Neighbor algorithms and genetic parameter optimization, *J. Chem. Inf. Model.*, **46** (2006), 2412–2422.
50. A. Wirdiani, P. Hridayami, A. Widiari, K. Rismawan, P. Candradinata, I. Jayantha, Face identification based on K-Nearest Neighbor, *Sci. J. Inf.*, **6** (2019), 150–159.
51. O. Borgohain, M. Dasgupta, P. Kumar, G. Talukdar, Performance analysis of nearest neighbor, K-Nearest Neighbor and weighted K-Nearest Neighbor for the cassification of Alzheimer disease, in *Soft Computing Techniques and Applications*, S. Borah, R. Pradhan, N. Dey, P. Gupta, Ed., Springer, Singapore, (2021), 295–304.
52. S. Huang, M. Huang, Y. Lyu, A novel approach for sand liquefaction prediction via local mean-based pseudo nearest neighbor algorithm and its engineering application, *Adv. Eng. Inf.*, **41** (2019), 100918.
53. T. Chen, C. Guestrin, XGBoost: a scalable tree boosting system, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, USA, Association for Computing Machinery, (2016), 785–794.
54. I. Babajide Mustapha, F. Saeed, Bioactive molecule prediction using extreme gradient boosting, *Molecules*, **21** (2016), 983.
55. J. H. Friedman, Greedy function approximation: a gradient boosting machine, *Ann. Stat.*, **29** (2001), 1189–1232.