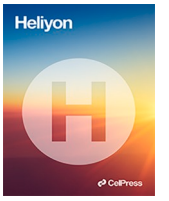
Heliyon 10 (2024) e23187

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon



SNO-DCA: A model for predicting *S*-nitrosylation sites based on densely connected convolutional networks and attention mechanism

Jianhua Jia [a,*], Peinuo Lv [a], Xin Wei [b], Wangren Qiu [a]

[a] *Computer Department, Jingdezhen Ceramic University, Jingdezhen, 330403, China*
[b] *Business School, Jiangxi Institute of Fashion Technology, Nanchang, 330201, China*

ARTICLE INFO

Keywords:
S-nitrosoylation
Dense convolutional networks
Attention mechanisms
Deep learning

ABSTRACT

Protein *S*-nitrosylation is a reversible oxidative reduction post-translational modification that is widely present in the biological community. *S*-nitrosylation can regulate protein function and is closely associated with a variety of diseases, thus identifying *S*-nitrosylation sites are crucial for revealing the function of proteins and related drug discovery. Traditional experimental methods are time-consuming and expensive; therefore, it is necessary to explore more efficient computational methods. Deep learning algorithms perform well in the field of bioinformatics sites prediction, and many studies show that they outperform existing machine learning algorithms. In this work, we proposed a deep learning algorithm-based predictor SNO-DCA for distinguishing between *S*-nitrosylated and non-*S*-nitrosylated sequences. First, one-hot encoding of protein sequences was performed. Second, the dense convolutional blocks were used to capture feature information, and an attention module was added to weigh different features to improve the prediction ability of the model. The 10-fold cross-validation and independent testing experimental results show that our SNO-DCA model outperforms existing *S*-nitrosylation sites prediction models under imbalanced data. In this paper, a web server prediction website: https://sno.cangmang.xyz/SNO-DCA/was established to provide an online prediction service for users. SNO-DCA can be available at https://github.com/peanono/SNO-DCA.

1. Introduction

As one of the most important PTMs, *S*-nitrosylation (SNO) is the process of covalent binding of nitro ions to protein cysteine residues [1]. NO is a new type of messenger molecule that participates in and regulates various life activities and plays an important role in human functions such as the nervous system, immune system, cardiovascular and cerebrovascular. NO generate nitrogen oxides with oxygen and thus *S*-nitrosylate proteins containing free hydrophobic groups, as well as *trans*-nitrosylation reactions with protein cysteine hydrophobic groups. In brief, SNOcan directly or indirectly regulate protein function [2]. In addition, SNOis also involved in vital activities such as ion channels, influencing kinase activity, and DNA activity to participate in cell signaling, apoptosis, etc. Levels of *S*-nitrosylated products are associated with various human diseases such as cardiovascular disease [3], cancer [4], diabetes, asthma, chronic renal failure [5], and Alzheimer's disease [6]. Accurate identification of protein SNOsites are essential for 1relevant drug

* Corresponding author.
E-mail address: jjh163yx@163.com (J. Jia).

https://doi.org/10.1016/j.heliyon.2023.e23187
Received 4 May 2023; Received in revised form 22 November 2023; Accepted 29 November 2023
Available online 3 December 2023

development and understanding of fundamental biological processes, therefore, it is necessary for understanding the underlying cellular and protein mechanisms as well as disease prevention and treatment.

The traditional biochemical methods for predicting SNOsites include biotin switch assay (BSA) [7], BSA combined with protein sequencing technology [8], and BSA combined with isotope-coded affinity tag (ICAT) [9]. In the era of big data, as the volume of data increases, these experimental methods can no longer meet the prediction demand, and there is an urgent need for effective computational means to improve the efficiency of SNO sites prediction.

In recent years, machine learning has become a common method for protein SNO sites prediction. Hao et al. [10] proposed the SNOSID method and introduced Support Vector Machine (SVM). Xue et al. [11] used the amino acid substitution matrix to calculate the *S*-nitrosylated peptide sequences and obtained the corresponding score to develop a GPS-SNO predictor. Lee et al. [12]developed a SNO sites predictor using maximal dependence decomposition (MDD) to divide the SNO sites into different groups and used SVM to generate a prediction model for each MDD cluster motif. iSNO-PseAAC predictor proposed by XU et al. [13] used PseAAC to represent protein sequence information, constructed a position-specific amino acid Propensity (PSAAP) matrix, and predicted sites using a conditional random field (CRF) algorithm. Although the above methods have achieved several results, the training sets are small, the features are not comprehensive, and the potential information is easily ignored. And machine learning needs to rely on a variety of manually obtained features, which cannot extract advanced features of protein sequences. Deep learning models are less affected by the manual intervention and can extract protein sequence features in depth.

Xie et al. [14] [][][]first developed DeepNitro, a prediction tool for nitro sites based on deep learning algorithms, which used four coding features to construct a neural network model with an eight-layer architecture. Hasan et al. [15] undersampled Xie's dataset to form a training set with an equal number of positive and negative samples and used four different coding schemes to develop a prediction tool PreSNO by integrating SVM and random forest (RF) methods. Siraj et al. [16] applied the method of natural language processing (NLP) to protein feature coding and developed the RecSNO predictor using a bi-directional long and short-term memory network BiLSTM architecture to process sequence data. Although DeepNitro predictors are trained on unbalanced data sets, their features are subject to more manual intervention and there is much room for further improvement in terms of accuracy. Both PreSNO and RecSNO used balanced training sets, while predictions on balanced sets are rarely encountered in practice, and predictions under imbalanced data will be more practical and more rigorous.

To solve the above problems, densely connected convolutional networks (DenseNets) [17]and attention mechanisms are introduced in this study to propose a new prediction model for SNOsites called SNO-DCA. We reduced the intervention caused by manual feature extraction by using only one-hot encoding, focus onhigh-dimensional features by densely connected convolutional networks to extract sequence high-level features. The Efficient Channel Attention Module (ECA-Net) [18] to evaluate the importance of features on weighting. Fully connected layers to capture the relationship between high-dimensional features, the introduction of class weights to solve the problem of data imbalance, and finally softmax classification. The experimental results show that our model outperforms existing models and can effectively identify SNO sites under imbalanced data.

2. Materials and methods

2.1. Benchmark dataset

A high-quality dataset is the cornerstone of scientific research. In this study, our benchmark dataset was derived from Xie et al. [14], which is based on an extensive literature search and a summary of previously reported datasets. 4762 SNO sites were obtained from 3113 proteins. Of these, the experimentally validated SNO sites were positive samples and all other samples were negative sites. To reduce the risk of overfitting the model due to sequence redundancy [19], the collected protein sequences were clustered using a CD-HIT with a threshold of 40 %. Finally, we retained 3409 positive sites and 17,103 negative sites as the training set and 365 positive sites and 3354 negative sites as the independent test set.

Potential peptide samples containing SNO sites can be expressed as shown in Equation (1):

$$P_{\zeta}(C) = R_{-\zeta}R_{-(\zeta-1)\ldots}R_{-2}R_{-1}CR_{+1}R_{+2\ldots}R_{+(\zeta-1)}R_{+\zeta} \tag{1}$$

where P is the peptide, the center C is the cysteine, ζ is an integer, and R_(-ζ) and R_(+ζ) represent the ξth upstream amino acid residue and downstream amino acid residue of cysteine C, respectively. We used a sliding window to extract the fragments with a window size $L = 2\xi+1$. In this paper, we set $\xi = 20$ and fill the missing amino acids with X if the peptide sequence length is less than 20. The statistical information of the dataset is shown in Table 1.

Table 1
Statistics for the original sample data.

Original data set	Positive site	Negative site
Training set	3409	17,103
Independent test set	365	3354

2.2. One-hot encoding

In this work, we encode the protein sequence using one-hot encoding. This encoding is represented by the binary numbers 0 and 1. The amino acids in the protein sequence corresponding to an index of 1 and other positions are 0 [20], for example, alanine is encoded as (10⋯0). There are 20 amino acids in total, in addition, the unknown amino acid is set to X. So, for the sequence fragment of length L, the final two-dimensional sparse matrix of $L \times 21$ dimensions is obtained. In this paper, L is 41, then we have a 41×21 (861) dimensional vector matrix.

2.3. Model structure

In this paper, we used deep learning algorithms to build a prediction model which can explore the deep features of SNO sites. In this model, we extracted high-level features from a matrix of sequences that have been one-hot encoded by a densely connected convolutional network. The features are fed into an efficient channel attention mechanism to weigh the important features. Finally, after four fully connected layers, the output high-level features are fed to the softmax layer for classification, which effectively predicts the SNO sites. The structure of the SNO-DCA model is shown in Fig. 1. For the information encoded by one-hot, advanced features are extracted through the densely connected convolutional networks. Input the features into the attention mechanism of the efficient channel to weigh the important features. After four fully-connected layers, the output advanced features are input to the softmax layer for classification.

2.4. Densely connected convolutional networks to extract features

It has been shown that DenseNets have better performance compared to traditional Convolutional Neural Network (CNN) and Residual Networks (ResNet) [21]. DenseNets use a dense connectivity mechanism, where the input of each layer is the output of all previous layers in the channel dimension. This not only mitigates the gradient disappearance to some extent but also enhances the transmission of features and achieves better performance with fewer parameters and less computation. The propagation process of different networks is shown in Fig. 2(A-C).

Before applying the dense convolution block, the one-hot encoded vector of length L is firstly convolved by one-dimensional convolution to generate a low-level feature map of protein sequence information using the ELU activation function, which implements the nonlinear transformation. As shown in Equation (2).

$$h^0 = a(E \times W + b) \tag{2}$$

where E is one-hot encoding, W is the weight matrix with size $21 \times S \times D$, 21 is the length of one-hot encoding, S is the convolutional kernel size, D is the number of convolutional kernels, and b is the bias term. Here, S is set to 4 and D is set to 96. h^0 is the output of the one-dimensional convolutional layer of size $L \times D$.

The output vector of the feature encoding is the input vector of the dense convolutional network. The dense convolutional block uses a series of convolutional operations to obtain a high-dimensional feature representation map, as shown in Equation (3).

$$h^k = a\left(\left[h^0; h^1; \ldots; h^{k-1}\right] \times W^{'} + b^{'}\right) \tag{3}$$

where h^{k-1} denotes the feature vector generated by the $(k-1)^{th}$ convolution in the dense convolution block. $W^{'} \in R^{D^{'} \times S \times D^{''}}$ is the weight matrix, $D^{'}$ is determined by K and denotes the number of convolution kernels in the kth convolution layer, $D^{''} = 32$. $b^{'}$ is the bias term, $[h^0; h^1; \ldots; h^{k-1}]$ indicates that the outputs $h^0; h^1; \ldots; h^{k-1}$ of the dense convolution blocks are concatenated along the feature dimension.

Then, a transition layer is used between the two dense convolution blocks to perform convolution and pooling operations on the feature maps of the obtained protein sequence information. The transition layer is shown as Equations (4) and (5).

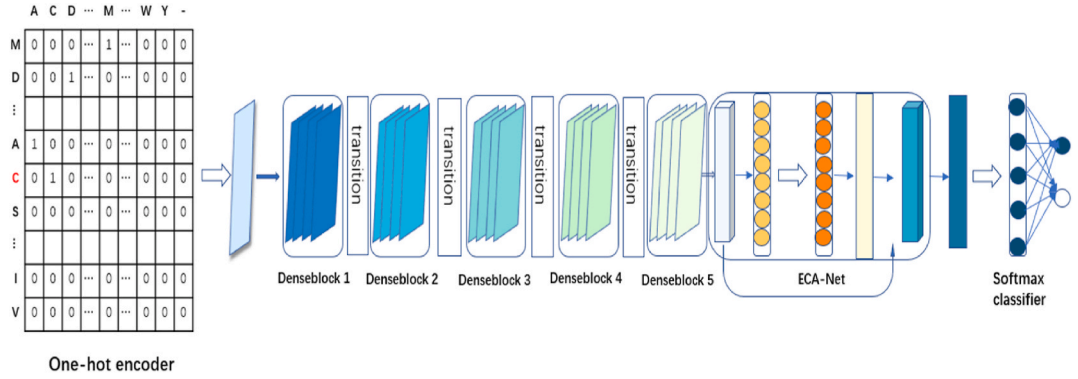


Fig. 1. The model framework of SNO-DCA.

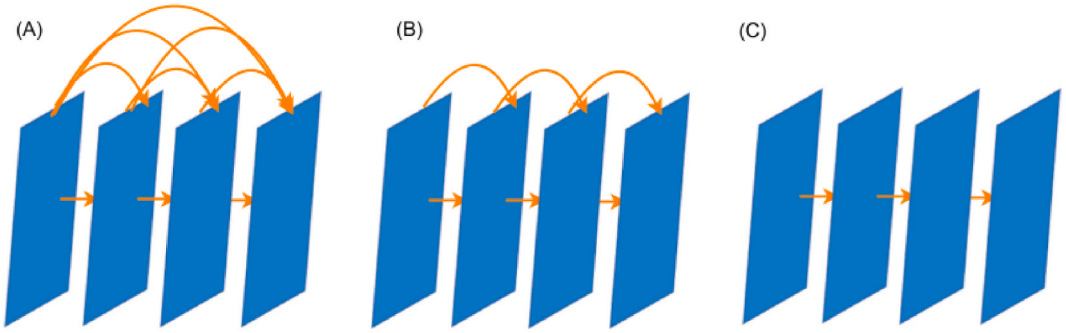


Fig. 2. Communication process of different networks (A) Densely connected convolution network structure. (B) Short-circuit connection mechanism of Residual Network. (C) Convolutional Neural Network structure.

$$h^k = a\left(\left[h^0; h^1; \ldots; h^{k-1}\right] * W^{''} + b^{''}\right) \tag{4}$$

$$h^{(trans)} = f_{AvgPooling1D}(X) \tag{5}$$

where $W^{'} \in R^{(D^{'}+D^{'})\times S^{'}\times(D^{'}+D^{'})}$ is the weight matrix, $S^{'}$ is the size of the convolution kernel set to 1, and $b^{''}$ is the bias. Finally, the average pooling operation is applied to h^k to reduce the dimensionality and obtain the output $h^{(trans)}$ of the transition layer.

Multiple dense convolutional blocks and transition layers are connected in series to construct a dense convolutional network. In this study, the number of dense convolutional blocks is set to 5. Finally, we can extract the high-level features of protein sequences *h*.

2.5. -net

Since different features have different levels of importance, we add ECA-Net to each module to weigh the features to enhance useful information and improve model prediction. ECA-Net is implemented based on global average pooling and adaptive computation of one-dimensional convolution of the convolution kernel size k, which effectively avoids dimensionality reduction (Fig. 3). After using global average pooling (GAP) aggregation convolution features without dimension reduction, the ECA module first adaptively determines the kernel size k, then conducts one-dimensional convolution, and the sigmoid function learns the channel attention.

First, the global averaging pooling operation is performed on the high-level features extracted from the dense convolutional blocks, and the features *h* with width *w* and height *H* and a number of channels *C* are compressed into features $\widehat{h}$ of size $1 \times 1 \times C$. As shown in Equation (6).

$$\widehat{h} = f_{GAP}(h) \tag{6}$$

Then a one-dimensional convolution operation is performed to capture the information between different channels, and the feature size obtained after one-dimensional convolution remains $1 \times 1 \times C$. As shown in Equation (7).

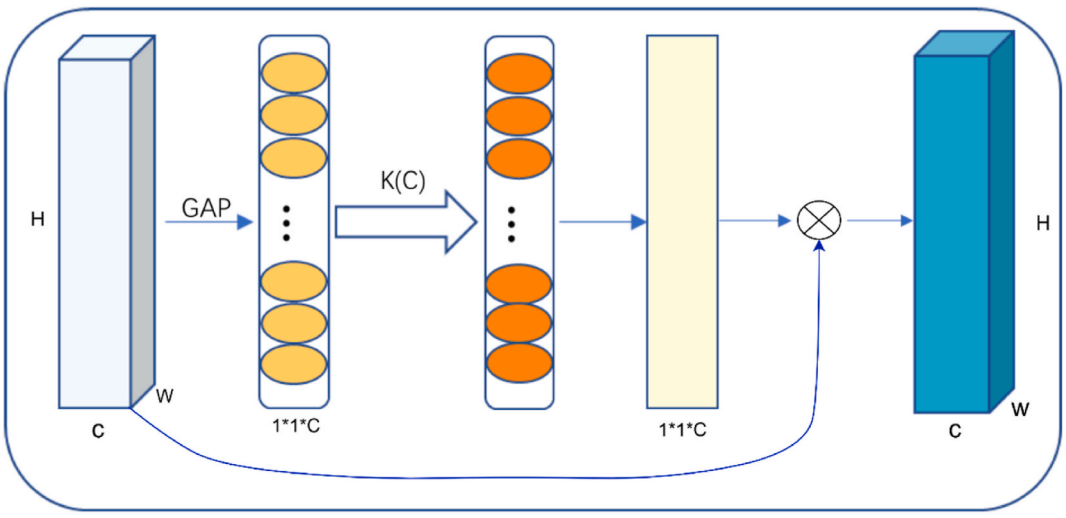


Fig. 3. The efficient channel attention model (ECA-Net).

$$q = a(C1D_k(\widehat{h})) \tag{7}$$

where q is the weight of each channel and a is the Sigmoid activation function. Where *C1D* denotes a one-dimensional convolution and the convolution kernel size is k, i.e., the coverage of cross-channel information interaction, and k is proportional to the number of channels C. Since the channel dimension, C is usually set to the power of 2, a nonlinear mapping relation φ is introduced as shown in Equation (8).

$$C = \varphi(k) = 2^{(\gamma \times k - b)} \tag{8}$$

Given the channel dimension C, the adaptive function of the one-dimensional convolutional kernel size is shown in Equation (9).

$$k = \left| \frac{log_{2(C)}}{\gamma} + \frac{b}{\gamma} \right| \tag{9}$$

where $\gamma = 2$ and $b = 1$. Finally, the weights q is multiplied channel by channel with the original input high-level feature maps to obtain the weighted high-level feature maps. As shown in Equation (10).

$$\widetilde{h} = h \times q \tag{10}$$

Next, all information is integrated by four fully connected layers with 80, 40, 40 and 20 units, respectively, and a softmax classifier is used to classify the SNO sites, and the predicted classes of the samples are obtained after weighting and activation operations, as shown in Equation (11).

$$p(y = i|x) = \frac{exp\left(W_i^s.\widetilde{h} + b_i^s\right)}{\sum_{j=1}^{2} exp\left(W_j^s.\widetilde{h} + b_j^s\right)} \tag{11}$$

where W_i^s and W_j^s are the weight matrices and b^s is the bias. $p(y = i|x)$ denotes the probability that sample x is predicted to be class i, $i \in \{0, 1\}$.

Imbalanced data will affect the prediction ability of the model, which is the problem to be solved [22–24]. For this reason, we adopt the method of class weights and set the weight ratio of positive and negative samples to 5:1. Increasing the influence of positive samples allows the model to learn the sequence mechanism of SNO samples better, thus improving the prediction ability of the model.

2.6. Performance evaluation

In this study, we evaluate the performance of the model by utilizing both ten-fold cross-validation and independent test sets. We calculated four statistical metrics: sensitivity (Sn), specificity (Sp), accuracy (Acc), and Mathews correlation coefficient (MCC) [25–27] with the following equations. As shown in Equation 12–15

$$\text{Sn} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{12}$$

$$\text{Sp} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{13}$$

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{14}$$

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP}) \times (\text{TP} + \text{FN}) \times (\text{TN} + \text{FN}) \times (\text{TN} + \text{FP})}} \tag{15}$$

Sn was used as a measure of positive accuracy, i.e., an index of accuracy in identifying SNO sites. Sp was used as a measure of negative accuracy, i.e., an index of accuracy in identifying non-SNO sites. Acc represented the proportion of correctly classified samples to the total number of samples, and MCC was considered a balanced measure [28]. Here TP, TN, FP, and FN denote true positive, true negative, false positive, and false negative, respectively. In addition, we used the receiver operating characteristic (ROC) area under the ROC curve (AUC) as a measure of model goodness [29]. It is obvious that the larger AUC value indicates, the higher model performance.

3. Results and discussion

3.1. Optimization of important hyperparameters

In order to find the optimal network configuration for the model SNO-DCA, which leads to the training of a better site predictor. A few important hyperparameters in the model SNO-DCA, including the number of convolutional kernels in the first CNN layer, the

number of dense blocks and the number of convolutional layers in the dense blocks, were determined by ten-fold cross-validation.

Firstly, the number of convolution kernels of the first layer CNN and the layers of the dense blocks were preset to 96 and 4. Adjusted the number of dense blocks, the best performance of the model is achieved when the number of dense blocks were 5 (Fig. 4A). Secondly, after fixing the number of dense blocks, we adjusted the number of convolution layers in the dense blocks, and determined that the best result is achieved when the number of layers were 4 (Fig. 4B). Lastly, we fixed the two parameters and then adjusted the number of convolutional kernels in the first layer of CNN to reach the best performance when the number of filters were 96 (Fig. 4C).

3.2. SNO-DCA ablation experiments

To verify the necessity of each component in SNO-DCA, we designed an ablation experiment by ten-fold validation using the training dataset. The experiments included the use of densely connected convolutional networks and the introduction of the ECA-Net layer in the model. The benchmark model adopted the concatenation of the original protein sequence information as input and extracted features by traditional CNNs.

The experimental results are shown in Table 2, different columns represent different model architectures. The proposed model performed best overall when densely connected convolutional networks and ECA-Net were used. First, to further verify the advantages of dense convolutional networks, we replaced conventional CNNs with DenseNets. Then, ECA-Net was added after each dense convolutional network. The results of ten-fold cross-validation showed that all metrics are significantly improved and dense convolutional networks have significant advantages over conventional CNNs. In addition, ECA-Net also could raise the model performance.

3.3. Cross-validation performance

We used ten-fold cross-validation to test the results. For an accurate comparison, we used the same training set and test set as the DeepNitro model. The comparison results of all metrics are shown in Table 3. The AUC of the SNO-DCA model is 0.77, and the ROC curve is shown in Fig. 5.

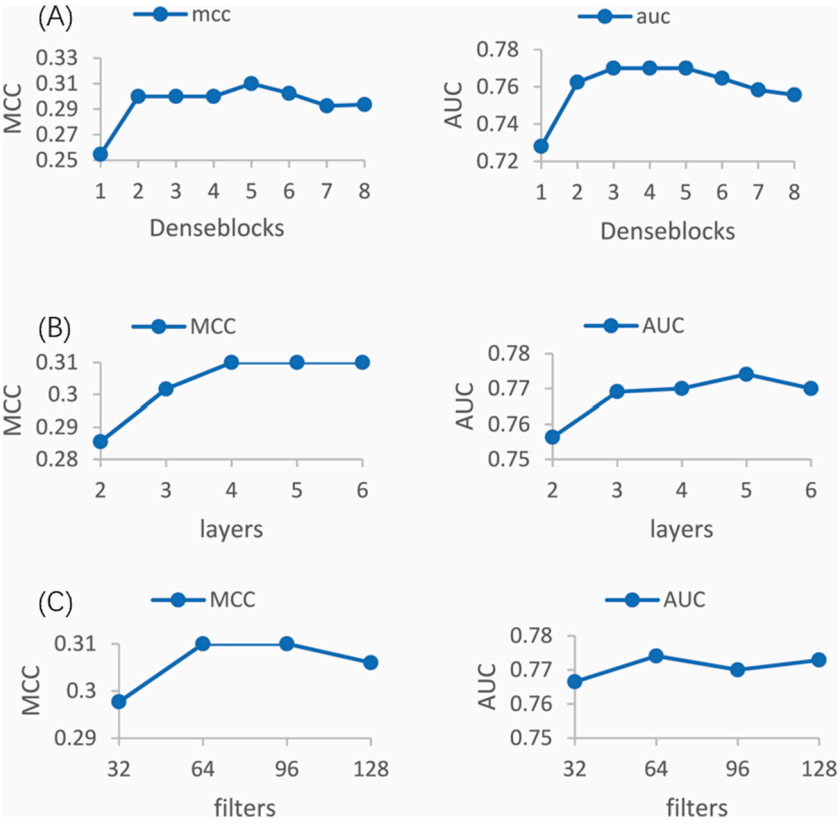


Fig. 4. The results of parameter selection of the model SNO-DCA (A) Selection of the number of dense blocks. (B) Selection of the number of layers. (C) Selection of the number of filters.

Table 2
The 10-flod cross-validation performance with different model structure.

CNN	✓	✓	✓
+ DenseNets		✓	✓
+ECA-Net			✓
SN	0.32	0.80	0.80
SP	0.80	0.59	0.62
ACC	0.40	0.63	0.65
MCC	0.09	0.30	0.31
AUC	0.60	0.76	0.77

Table 3
Comparison with other methods on the training data set.

Predictors	Sn	Sp	Acc	MCC	AUC
DeepNitro	NA	NA	NA	NA	0.71
SNO-DCA	**0.80**	0.62	**0.65**	**0.31**	**0.77**

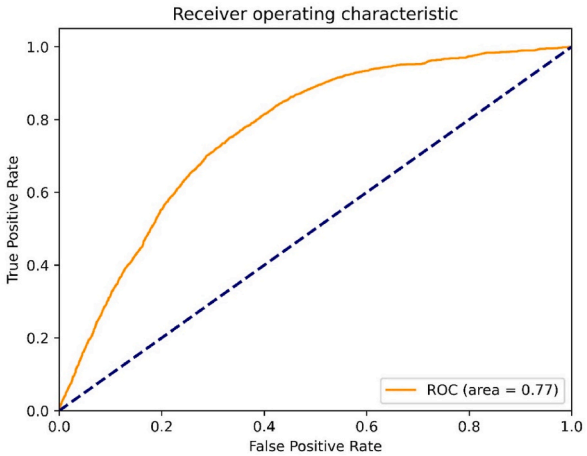


Fig. 5. Ten-fold cross-validated ROC curve.

3.4. Comparison of independent datasets with existing models

To evaluate the performance of SNO-DCA and ensure the good generalization ability of the model, we have selected the following methods for predicting SNO sites for comparison with our model, taking into account the fact that some models use different training sets and do not provide independent prediction tools. DeepNitro is based on an eight-layer neural network, PreSNO is a linear regression model based on SVM and RF, and RecSNO uses protein sequences as input and is a predictor based on embedded layers and BILSTM. The results are shown in Table 4. The prediction results of the deep learning predictor SNO-DCA constructed in this paper are Sn = 0.82; Sp = 0.69; Acc = 0.70; MCC = 0.32; AUC = 0.82. The ROC curves are shown in Fig. 6.

In Table 4, we can find that SNO-DCA has the highest Sn, MCC, and AUC; however, it has relatively low Sp due to the antagonistic formula of Sn and Sp. Similar to the finding of Ref [30], a significant increase in Sn usually lead to a decrease in Sp and thus Acc. Both PreSNO and RecSNO were constructed on a balanced dataset after downsampling the original dataset trained, in which the negative

Table 4
Comparison with other methods on the same test data set.

Predictors	Sn	Sp	Acc	MCC	AUC
GPS-SNO	0.28	0.74	0.69	0.01	0.52
ISNOPseAAC	0.29	0.76	0.71	0.03	NA
SNOSite	0.67	0.45	0.47	0.07	NA
DeepNitro	0.58	0.76	0.73	0.22	0.73
PreSNO	0.60	0.77	0.75	0.25	0.76
RecSNO	0.77	0.71	0.71	0.30	0.80
SNO-DCA	**0.82**	0.69	0.70	**0.32**	**0.81**

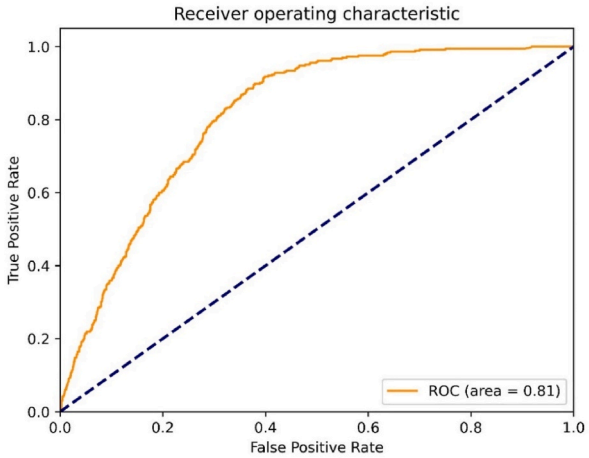


Fig. 6. ROC curve of the independent test set.

samples of the dataset are not sufficient. However, models trained on unbalanced training sets are more relevant. In the biological field, the real datasets are often unbalanced and it is especially important to identify as many SNO sites as possible, i.e., the values of Sn and MCC are more important than the other parameters. In contrast, the SNO-DCA predictor constructed in this paper solves the problem of unbalanced data by class weights and does not require the construction of balanced data sets with high Sn and MCC by undersampling, which effectively overcomes the disadvantage of the inadequacy for negative samples brought by the training under balanced data, avoids the lack of information to some extent. Therefore, our SNO-DCA model is more relevant and practical for predicting SNO sites.

3.5. Server online prediction website

To facilitate the utilization of our prediction model and provide experimental help to scientific researchers, we have constructed a user-friendly web server website https://sno.cangmang.xyz/SNO-DCA/, whose homepage is shown in Fig. 7.

Step1 Enter the URL in your browser and click Search to open the SNO-DCA sever homepage as shown in Figure. Click the Help button in the upper right corner or the More info … Button in the middle of Home to jump to the User's Guide, where one can view the background and a brief introduction about the SNO-DCA prediction model, and learn about the model structure and advantages; the user can see the sequence input format, and click example to jump to the Example interface, which shows the correct format and the incorrect format of the input sequences.

Step2 Visitors can enter a single protein sequence for query in the center box of the homepage and click the Submit button to make a prediction, and the predicted result will be highlighted in red, indicating SNO sites. As shown in Fig. 8.

Step3 You can also pass the file in FASTA format to the bottom left box of the homepage, enter your username and email address in the bottom right box, and click Submit for batch prediction, and the final prediction will be emailed to the recipient.

This platform simply encodes the original sequence of proteins to reduce manual feature selection intervention. A dense convolutional network is introduced to extract the high-level features, and an efficient channel attention mechanism is added to weight different features and learn important features. The efficient channel attention mechanism is added to weight different features to learn important features. Finally, the information is fused by the fully connected layer and inputted into the softmax layer for site prediction, and the SNO sites are identified by the prediction probability. The platform gives full effect to the deep learning models to explore the potential information of SNO sites, extract important features and improve the accuracy of SNO sites prediction.

4. Conclusions

In this study, we proposed SNO-DCA, a deep learning model for SNO sites prediction based on DenseNets and an efficient channel attention module under unbalanced data. The model performs a simple feature representation of protein sequence information, extracts high-level features by dense convolutional blocks, and adds an efficient attention module to further enhance the feature information. The results of both ten-fold cross-validation and independent tests show that SNO-DCA is a powerful prediction tool for identifying SNO sites. We developed a web online server to provide convenience to researchers. Both the training and test sets in this study are non-equilibrium datasets, which meet the practical prediction conditions. Another advantage in this study is that only protein sequences are simply encoded, which greatly reduces the intervention problem of manual feature selection. Meanwhile, the developed deep learning algorithm architecture is used to dig the potential information of the sequences in depth, which can give full play to the deep learning model.

In the follow-up work for this study, we will try not to encode the protein sequences in advance, but to embed the original sequences into the deep learning model to achieve end-to-end prediction, directly mining the hidden information of the samples and improving

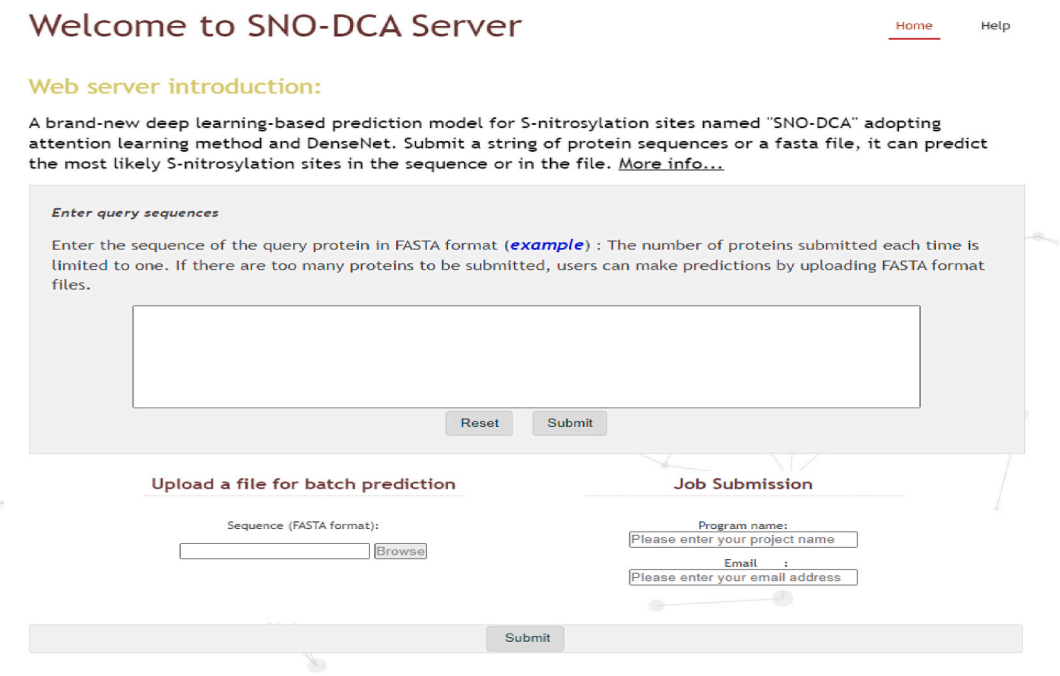


Fig. 7. Screenshot of the homepage of the web server SNO-DCA.

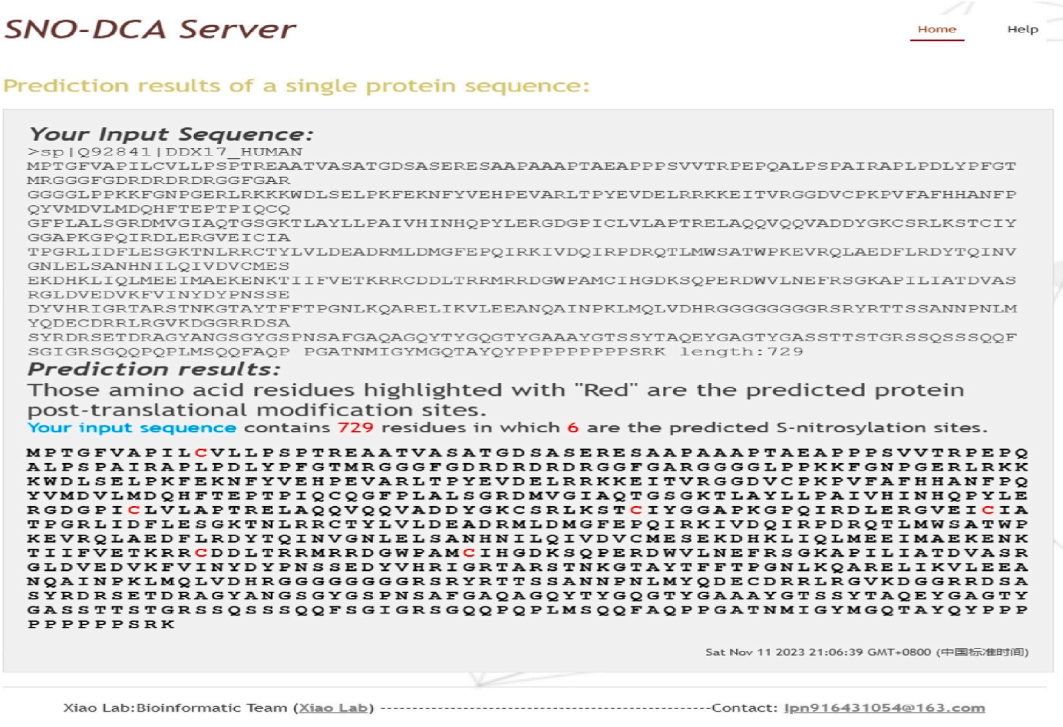


Fig. 8. Screenshot of the predicted results of the web server SNO-DCA.

the prediction accuracy. In addition, we will also explore more possibilities of deep learning algorithms for protein locus prediction, such as capsule network for prediction of small sample data, improved long and short-term memory network combined with improved CNN, etc. At present, the problem of data imbalance is still a major difficulty in the field of bioinformatics. Though the traditional methods of balancing data may have some certain value in the data information processing, we should transform our focus into the

imbalanced data with employing the improved/enhanced deep learning algorithm architecture to adapt to the actual protein prediction circumstance.

Data availability statement

Data will be made available on request. SNO-DCA model and datasets can be available at https://github.com/peanono/SNO-DCA.

Additional information

No additional information is available for this paper.

CRediT authorship contribution statement

Jianhua Jia: Writing – review & editing. **Peinuo Lv:** Writing – original draft. **Xin Wei:** Data curation. **Wangren Qiu:** Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was partially supported by the National Nature Science Foundation of China (Nos. 61761023 and 62162032), the Natural Science Foundation of Jiangxi Province, China (Nos. 20202BABL202004), the Scientific Research Plan of the Department of Education of Jiangxi Province (GJJ212419 and GJJ2201004). These funders had no role in the study design, data collection and analysis, decision to publish or preparation of manuscript.

Abbreviations

SNO	*S*-nitrosylation
BSA	biotin switch assay
ICAT	isotope-coded affinity tag
SVM	Support Vector Machine
MDD	maximal dependence decomposition
PSAAP	position-specific amino acid Propensity
CRF	conditional random field
RF	random forest
NLP	natural language processing
DenseNets	densely connected convolutional networks
ECA-Net	Efficient Channel Attention Module
CNN	Convolutional Neural Network
ResNet	Residual Networks
GAP	global average pooling
Sn	sensitivity
Sp	specificity
Acc	accuracy
MCC	Mathews correlation coefficient
ROC	receiver operating characteristic
AUC	area under the ROC curve

References

[1] Matthew, W., Foster, et al. Protein S-nitrosylation in health and disease: a current perspective - ScienceDirect [J]. Trends Mol. Med., 15(9): 391-404.
[2] X. Zhang, C. B, Regulation of cellular function by protein sulfhydryl nitrosylation [J], Chin. J. Pathophysiol. 27 (11) (2011) 2237–2240.
[3] I.V. Turko, F.J.P.R. Murad, Protein nitration in cardiovascular diseases [J], Pharmacol. Rev. 54 (4) (2002) 619.
[4] Z. Wang, Protein S-nitrosylation and cancer [J], Cancer Lett. 320 (2) (2012) 123–129.
[5] M. Piroddi, A. Palmese, F. Pilolli, et al., Plasma nitroproteome of kidney disease patients [J], Amino Acids 40 (2) (2011) 653–657.
[6] Heneka T S W S B-a S S G K B T. Quantitative proteomics of synaptosome S-nitrosylation in Alzheimer's disease [J], J. Neurochem. 152 (6) (2020) 710–726.
[7] S.R. Jaffrey, S.H. Snyder, The biotin switch method for the detection of S-nitrosylated proteins [J], Sci. STKE : Signal Transduct. Knowl. Environ. 2001 (86) (2001) pl1.

[8] C. Lindermayr, G. Saalbach, J. Durner, Proteomic identification of S-nitrosylated proteins in Arabidopsis [J], Plant Physiol. 137 (3) (2005) 921–930.
[9] C. Wu, T. Liu, W. Chen, et al., Redox regulatory mechanism of transnitrosylation by thioredoxin [J], Mol. Cell. Proteomics : MCP 9 (10) (2010) 2262–2275.
[10] G. Hao, B. Derakhshan, L. Shi, et al., SNOSID, a proteomic method for identification of cysteine S-nitrosylation sites in complex protein mixtures, Proc. Natl. Acad. Sci. U.S.A. 103 (4) (2006) 1012–1017.
[11] Y. Xue, Z. Liu, X. Gao, et al., GPS-SNO: computational prediction of protein S-nitrosylation sites with a modified GPS algorithm [J], PLoS One 5 (6) (2010), e11290.
[12] T.Y. Lee, Y.J. Chen, T.C. Lu, et al., SNOSite: exploiting maximal dependence decomposition to identify cysteine S-nitrosylation with substrate site specificity [J], PLoS One 6 (7) (2011), e21849.
[13] Y. Xu, J. Ding, L.Y. Wu, et al., iSNO-PseAAC: predict cysteine S-nitrosylation sites in proteins by incorporating position specific amino acid propensity into pseudo amino acid composition [J], PLoS One 8 (2) (2013), e55844.
[14] Y. Xie, X. Luo, Y. Li, et al., DeepNitro: prediction of protein nitration and nitrosylation sites by deep learning [J], Dev. Reprod. Biol. 16 (4) (2018) 294–306.
[15] M.M. Hasan, B. Manavalan, M.S. Khatun, et al., Prediction of S-nitrosylation sites by integrating support vector machines and random forest [J], Molecular omics 15 (6) (2019) 451–458.
[16] A. Siraj, T. Chantsalnyam, H. Tayara, et al., RecSNO: prediction of protein S-nitrosylation sites using a recurrent neural network [J], IEEE Access 9 (2021) 6674–6682.
[17] G. Huang, Z. Liu, V. Laurens, et al., Densely Connected Convolutional Networks, proceedings of the IEEE Computer Society, F, 2016 [C].
[18] Q. Wang, B. Wu, P. Zhu, et al., ECA-net: efficient Channel Attention for deep convolutional neural networks, in: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), F, 2020 [C].
[19] Beifang LiminNiu, et al., CD-HIT: accelerated for clustering the next-generation sequencing data [J], Bioinformatics 28 (23) (2012) 3150–3152.
[20] D. Wang, Y. Liang, D. Xu, Capsule network for protein post-translational modification site prediction [J], Bioinformatics 35 (14) (2018) 2386–2394.
[21] K. He, X. Zhang, S. Ren, et al., Deep Residual Learning for Image Recognition; Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), F 27-30 June 2016, 2016 [C].
[22] Wei L, Hu J, Li F, et al. Comparative analysis and prediction of quorum-sensing peptides using feature representation learning and machine learning algorithms. LID - 10.1093/bib/bby107 [doi] [J]. (1477-4054 (Electronic)).
[23] L. Wei, P. Xing, J. Zeng, et al., Improved prediction of protein–protein interactions using novel negative samples, features, and an ensemble classifier [J], Artif. Intell. Med. 83 (2017) 67–74.
[24] Wei L, Zhou C, Chen H, et al. ACPred-FL: a sequence-based predictor using effective feature representation to improve the prediction of anti-cancer peptides [J]. (1367-4811 (Electronic)).
[25] Su R, Liu X, Xiao G, et al. Meta-GDBP: a high-level stacked regression model to improve anticancer drug response prediction [J]. (1477-4054 (Electronic)).
[26] L. Wei, W. He, A. Malik, et al., Computational prediction and interpretation of cell-specific replication origin sites from multiple eukaryotes by exploiting stacking framework [J], Briefings Bioinf. 22 (4) (2021).
[27] L. Wei, J. Tang, Q. Zou, Local-DPP: an improved DNA-binding protein prediction method by exploring local evolutionary information [J], Inf. Sci. 384 (2017) 135–144.
[28] D. Eisenberg, E.M. Marcotte, I. Xenarios, et al., Protein function in the post-genomic era [J], Nature 405 (6788) (2000) 823–826.
[29] D. Chicco, G. Jurman, The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation [J], BMC Genom. 21 (1) (2020) 6.
[30] H. He, E.A. Garcia, Learning from imbalanced data [J], IEEE Trans. Knowl. Data Eng. 21 (9) (2009) 1263–1284.