Plant Molecular Biology (2021) 105:601–610
https://doi.org/10.1007/s11103-020-01112-w

Computational identification of ubiquitination sites in *Arabidopsis thaliana* using convolutional neural networks

Xiaofeng Wang[1] · Renxiang Yan[2] · Yong-Zi Chen[3] · Yongji Wang[4]

Received: 7 July 2020 / Accepted: 27 December 2020 / Published online: 1 February 2021

Abstract

Key message **We developed two CNNs for predicting ubiquitination sites in *Arabidopsis thaliana*, demonstrated their competitive performance, analyzed amino acid physicochemical properties and the CNN structures, and predicted ubiquitination sites in *Arabidopsis*.**

Abstract As an important posttranslational protein modification, ubiquitination plays critical roles in plant physiology, including plant growth and development, biotic and abiotic stress, metabolism, and so on. A lot of ubiquitination site prediction models have been developed for human, mouse and yeast. However, there are few models to predict ubiquitination sites for the plant *Arabidopsis thaliana*. Based on this context, we proposed two convolutional neural network (CNN) based models for predicting ubiquitination sites in *A. thaliana*. The two models reach AUC (area under the ROC curve) values of 0.924 and 0.913 respectively in five-fold cross-validation, and 0.921 and 0.914 respectively in independent test, which outperform other models and demonstrate the competitive edge of them. We in-depth analyze the amino acid physicochemical properties in the neighboring sequence regions of the ubiquitination sites, and study the influence of the CNN structure to the prediction performance. Potential ubiquitination sites in the global *Arbidopsis* proteome are predicted using the two CNN models. To facilitate the community, the source code, training and test dataset, predicted ubiquitination sites in the *Arbidopsis* proteome are available at GitHub (http://github.com/nongdaxiaofeng/CNNAthUbi) for interest users.

Keywords Ubiquitination site · Prediction · Convolutional neural network · *Arabidopsis thaliana*

Introduction

Ubiquitin is a small 76-residue protein found in all eukaryotic cells. Ubiquitination is the addition of ubiquitin to lysine residues of substrate proteins. The whole process involves ubiquitin molecules, substrate proteins, enzyme systems (such as ubiquitin-activating enzymes E1s, ubiquitin-binding enzymes E2s, ubiquitin ligases E3s and deubiquitinating enzymes DUBs), and proteasomes. They together constitute the ubiquitin-proteasome system. Ubiquitination consists of three main steps. Firstly, the E1 enzyme adheres to the cysteine residue in the tail of ubiquitin to activate ubiquitin. Then the E1 enzyme transfers the activated ubiquitin molecule to the E2 enzyme. Finally, the E2 enzyme and some different E3 enzymes jointly recognize the target protein and bind ubiquitin to it. DUBs oppose the role of ubiquination, which remove ubiquitin from substrate proteins. Ubiquitination affects proteins by marking them for degradation, coordinating their cellular location, regulating their activity, and modulating

Supplementary Information The online version of this article (https://doi.org/10.1007/s11103-020-01112-w) contains supplementary material, which is available to authorized users.

✉ Xiaofeng Wang
wangxf@sxnu.edu.cn

✉ Renxiang Yan
yanrenxiang@fzu.edu.cn

[1] College of Mathematics and Computer Sciences, Shanxi Normal University, Linfen 041004, China

[2] School of Biological Sciences and Engineering, Fujian Key Laboratory of Marine Enzyme Engineering, Fuzhou University, Fuzhou 350002, China

[3] Laboratory of Tumor Cell Biology, Key Laboratory of Cancer Prevention and Therapy, Tianjin's Clinical Research Center for Cancer, National Clinical Research Center for Cancer, Tianjin Medical University Cancer Institute and Hospital, Tianjin Medical University, Tianjin 300060, China

[4] College of Life Sciences, Shanxi Normal University, Linfen 041000, China

protein-protein interactions (Glickman and Ciechanover 2002; Schnell and Hicke 2003). Therefore, ubiquitination is highly interconnected with a wide variety of cellular processes, including signal transduction, endocytic trafficking, DNA transcription and repair, cell division, differentiation, apoptosis, viral infection, immune response, and so on (Herrmann et al. 2007; Wagner et al. 2011).

Protein ubiquitination plays critical roles in multiple plant developmental stages. Many experimental studies have already proved that ubiquitination is involved in plant growth and development, biotic and abiotic stress, metabolism and so on. It was found that E3 ubiquitin ligase complex (SCF(FBL17)) participates in the regulation of plant germline proliferation by degrading cell cycle inhibitors (Kim et al. 2008). Direct ubiquitination of pattern recognition receptor FLS2 attenuates plant innate immunity (Lu et al. 2011). Light and E3 ubiquitin ligase COP1/SPA can regulate the protein stability of MYB transcription factors PAP1 and PAP2 through ubiquitin-proteasome pathway, and further control anthocyanin levels in *A. thaliana* (Maier et al. 2013).

In the study of ubiquitination, the identification of substrates and their corresponding modified sites is an important issue. At present, there are two main experimental methods for the identification of ubiquitination sites, site mutation and mass spectrometry. Although ubiquitination sites can be correctly determined by experimental methods, it is a time-consuming and laborious process, and is often restricted by the difficulty in obtaining affinity reagents such as specific antibodies and appropriate catalytic conditions. Fortunately, bioinformatics methods provide a more cost- and time-efficient approach which can be used for proteome-wide annotation and hypothesis-driven experimental design.

In fact, there have been some effective tools to predict ubiquitination sites (Cai and Jiang 2016; Chen et al. 2011; Feng et al. 2013; Fu et al. 2019). However, these existing tools vary in the selection of machine learning methods and training features. This also suggests further improvements are still possible. According to a previous research (Chen et al. 2015), ubiquitination sites of different species have their own characteristics, so specific prediction tools should be established for different species. Most existing ubiquitination site prediction methods focus on human, mouse and yeast. As one of the model organisms for plant biology, *A. thaliana* is the first plant to have its entire genome sequenced, but there are few models to predict ubiquitination sites for it. In 2016, ubiquitin combined fractional diagonal chromatography was implemented for proteome-wide ubiquitination site mapping on *A. thaliana* cell cultures, and 3009 sites on 1607 proteins were identified (Walton et al. 2016). Based on the data, the first *A. thaliana* ubiquitination site prediction model AraUbiSite was established, which used the amino acid type, amino acid composition, and *k*-spaced amino acid pair frequency in the protein sequences as features, and a support-vector machine (SVM) as training method.(Chen et al. 2019).

In order to improve the prediction performance, here we propose two convolutional neural network (CNN) based models to identify ubiquitination sites in *A. thaliana* proteins. In five-fold cross-validation, the proposed models reach AUC values of 0.924 and 0.913 respectively. In independent test, the models obtain AUC values of 0.921 and 0.914 respectively. It is demonstrated that the prediction performance of the two models outperforms those of AraUbiSite and some other machine learning models. We analyzed the physicochemical properties of amino acids near ubiquitination sites in *A. thaliana* protein sequences, and found the difference of amino acid properties between ubiquitinated and non-ubiquitinated sequences. We apply the two CNN models to an *Arabidopsis* proteome-wide prediction of ubiquitination sites. The obtained information, knowledge and data should be useful to the biological community. The predicted ubiquitination sites in the *Arabidopsis* proteome and the source code are publicly available at http://github.com/nongdaxiaofeng/CNNAthUbi for interest researchers.

Method

Benchmark dataset

Previously, a model called AraUbiSite was built to predict ubiquitination sites in *A. thaliana* (Chen et al. 2019). To make a fair and direct comparison, the AraUbiSite datasets are used to train and test our methods in this work. The ubiquitination sites in the datasets are experimentally determined using ubiquitin combined fractional diagonal chromatography, and the non-ubiquitination sites are in fact non-validated lysine residues (Walton et al. 2016). The datasets contain a training dataset and a test dataset. The training dataset includes 2043 ubiquitination sites and 6130 non-ubiquitination sites. The test dataset is consisted of 511 ubiquitination sites and 1533 non-ubiquitination sites. The ubiquitination sites and non-ubiquitination sites in the datasets are called positive samples and negative samples, respectively. The prediction procedure was a binary classification problem (i.e., classify a residue into ubiquitination or non-ubiquitination site). To build the prediction model, peptides of 41 amino acids centering the potential sites in the datasets were extracted. If the site is positioned at the very beginning or end of a protein sequence, which results the peptide shorter than 41, the character X is used to fill the termini of the peptide. In such a way, all peptides are of the same sizes. In addition, the identity of any two peptides in the datasets is less than 40%. Low sequence identity can avoid overfitting problem and make the prediction model effective in the real application.

Input features

Input features are a very critical factor to obtain accurate and reliable models in the machine learning methods. In biology, whether a lysine residue in a protein sequence is a ubiquitination site or not is closely related to the amino acids around it. Therefore, input features are constructed using these neighboring residues. We make use of CNN to train the model, which requires the input to retain the amino acid order in the sequence. Binary encoding and amino acid physicochemical properties are used to encode each amino acid in the sequence, which are briefly described as below.

Binary encoding

For the binary encoding, 'X' is also considered as an amino acid. So there are 21 types of amino acids together. Each amino acid is encoded as a 0–1 vector of length 21. For the *i*th type of amino acid, the *i*th position of the vector is 1, and the rest is 0. Because there are 40 neighboring amino acids centering the site, the encoding for each sample is a 40×21 matrix.

Amino acid properties

A previous study (Tung and Ho 2008) selected a set of 31 informative physicochemical properties (supplementary file 1) from a large set of 531 amino acid physicochemical properties in the AAindex database for predicting ubiquitination sites. Most selected properties represent the hydrophobicity of amino acids. There are also properties representing the volumes, propensity to form secondary structures, and occurring frequencies of amino acids. We used the 31 properties to encode each amino acid in the peptide after normalizing the values of each property to [0, 1] using the formula $x–x_{min}/(x_{max}\text{-}x_{min})$, where x_{min} and x_{max} are the minimum and maximum respectively for the property. X in the peptide is encoded as a zero vector of length 31. Then each sequence is encoded by a 40×31 matrix.

Structures of CNN models

Artificial neural networks (ANNs) are computing systems modeling the biological neural networks of the human brain (Zhang et al. 1998). An ANN is composed of connected units called artificial neurons. The fully connected neural network is one of the most widely used and rapidly developing ANNs. It adopts a structure of unidirectional multilayer. Each neuron in one layer receives signals from neurons of the former layer, feeds the weighted sum of them to the activation function, and sends the result to all neurons in the next layer. The first layer is called input layer, the last layer is called output layer, and middle layers are called hidden layers.

Convolutional neural networks (CNNs) are regularized versions of fully connected neural networks with convolution calculation (Krizhevsky et al. 2017). We developed two CNN models, which used the binary encoding and amino acid properties as inputs respectively. For convenience, we named the two CNN models CNN_Binary and CNN_Property respectively. The Keras package (Lee and Song 2019) with a tensorflow backend (Rampasek and Goldenberg 2016) is employed to implement the CNNs. Both of the two CNNs include an input layer, a convolutional layer, a max-pooling layer, a flatten layer, a dropout layer, a fully connected layer and an output layer (Fig. 1). The details of the layers are described as below.

In the CNN models, the convolutional layer receives the encoding matrix of width 40 from the input layer, and obtains the feature map through using n_1 filters of width w

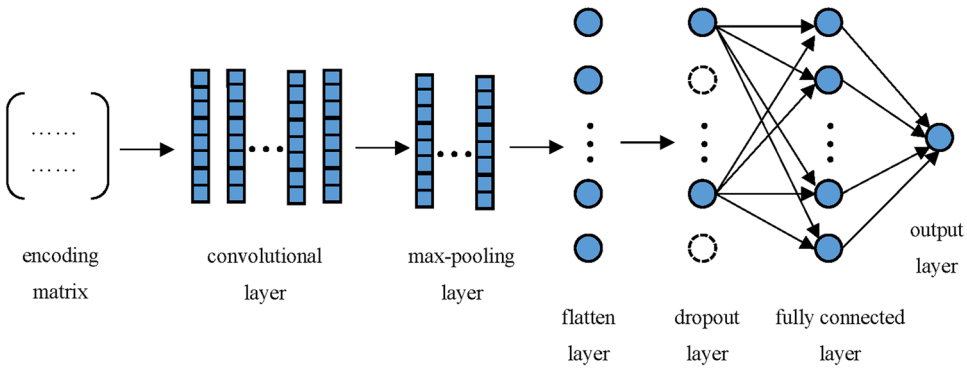

Fig. 1 The structure of the CNN model. The encoding matrix is the input layer. The convolutional layer does the convolution, captures local features for classification, and obtains a feature map. The max-pooling layer samples important features from the feature map and halves the width of it. The flatten layer transforms the feature map matrix to a vector. The dropout layer deleted a certain ratio of neurons to avoid overfitting. The fully connected layer does effective integration of the local features. The output layer outputs the prediction score

to do the convolution. Each filter is like a scanner to scan the amino acid sequence which are represented by the encoding matrix from left to right, and captures local features for the classification. In addition, zero-padding is used before doing the convolution, to make the feature map have the same width dimension as the input. The convolutional layer sends the feature map with width of 40 plus depth of n_1 to the max-pooling layer. The max-pooling layer separates the feature map into grids of length 2 along the width dimension, picks the neuron with the maximum value in each grid, and discards the rest. Such an operation keeps only the neurons that contribute the most in each grid and halves the width of the feature map to 20. The flatten layer flattens the 2D feature map to 1D. The dropout layer drops half of the neurons and their connections, which speeds up the training and prevents over-fitting. A fully connected layer of n_2 neurons follows the flatten layer. It assembles the local features created by the convolutional layer, and produces global features covering the entire sequence. The fully connected layer sends signals to the output layer, which has one neuron representing the probability for a sample to be ubiquitination site.

Rectified linear unit (ReLU) activation function is used in the convolutional layer and fully connected hidden layer for nonlinear transformation. The function is defined as: ReLU $(x) = \max(0, x)$. ReLU function can avoid the vanishing gradient problem and reduce the training time. The logistic function is used as the activation function of the output layer. The function is defined as $1/(1+\exp(-x))$. It generates the predicted value in the range of 0 to 1, which is able to denote the probability of the sample to be ubiquitination site (Chu 2020).

Training of the models

In the benchmark datasets, the number of positive samples is approximately one third of the number of negative samples. In order to tackle the imbalance problem, we repeat each positive sample three times when training. In general, a balanced training dataset can make a trained neural network robust.

We use the binary cross-entropy (Ramos et al. 2018) as a loss function to train the two CNNs. L1 regularization (Tibshirani 1996) is used to prevent overfitting, which adds a cost to the loss function. The cost is the sum of the absolute values of the connection weights plus a regularization coefficient λ. L1 regularization makes the weights of redundant neurons to be zero, which performs feature selection as well as regularization. We adopt RMSProp optimizer (Nhu et al. 2020) to optimize the two CNNs, which has shown excellent adaption of learning rate in various applications.

There are several hyper parameters in the two CNNs which should be carefully tuned. We use a simple grid search scheme (Dowsing 1970) to optimize hyper parameters that make the cross-validation perform best. The determined hyper parameter values are displayed in Table 1.

Performance evaluation

Both five-fold cross-validation and independent test are used to assess the performance of the model. In five-fold cross-validation, the training dataset is randomly divided into five subsets with nearly equal sizes. Of the five subsets, one single subset is retained as the validation set and the other four subsets are integrated as one set to train the model. This process is repeated five times with every subset used once as the validation set. In independent test, the training dataset is used to train the model, and the prediction result of the test dataset is used to test the model.

We use sensitivity, specificity, accuracy and area under the receiver operating characteristic (ROC) curve (Fawcett 2006) as metrics to evaluate the prediction performance of different models. As in most studies, we use 0.5 as the threshold to calculate the sensitivity, specificity, accuracy values. The formulas for sensitivity and specificity are defined as follows:

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

where TP, FP, TN and FN represent the numbers of true positives, false positives, true negatives and false negatives respectively.

Because the number of negative samples is about three times the number of positive samples, the usually used accuracy metric does not reflect the prediction performance well (Song et al. 2014). For example, if all samples are predicted as negative samples, the accuracy will be 75 %. For this reason, we define the accuracy as follows (Brodersen et al. 2010):

Table 1 Hyper parameters of the two CNN models

Model	n_1[a]	n_2[b]	η[c]	λ[d]	w[e]
CNN_Binary	16	64	0.001	0.0001	3
CNN_Property	32	512	0.001	0.0001	3

[a] n_1 is the number of filters in the convolutional layer

[b] n_2 is the number of neurons in the fully connected layer

[c] η is the learning rate of the optimizer

[d] λ is the coefficient of the L1 regularization

[e] w is the width of the filter

$$Accuracy = \frac{3 * TP + TN}{3 * (TP + FN) + TN + FP}$$

The ROC curve is a graphical plot showing the trade-off between the true positive rate and the false positive rate at every possible threshold. The true positive rate is equal to sensitivity, and the false positive rate is 1-specificity. In general, the closer the ROC curve is to the upper left corner, the better the prediction result is. Therefore, the area under ROC curve (AUC) is often used to measure the prediction performance of the model. At the same time, AUC also indicates the probability that the prediction score of a randomly selected positive sample is greater than that of a randomly selected negative sample.

Results

In order to efficiently and conveniently identify potential ubiquitination sites in *A. thaliana*, we develop two CNN based models. In this section, we first analyze the physicochemical properties of amino acids near ubiquitination sites in *A. thaliana*. Then we show the prediction performance of the models and study the effect of the CNN structure on the prediction performance. Finally, we make prediction of *A. thaliana* proteome-wide ubiquitination sites.

Analysis of amino acid properties

In the paper of AraUbiSite, the position-specific residue composition surrounding the ubiquitination sites was analyzed using the two-sample logo, where Arg, Lys and Glu are enriched near ubiquitination sites, while Ser and Leu are depleted. As a complement to it, we analyze physicochemical properties of residues surrounding ubiquitination sites in *A. thaliana*.

A previous study categorized the amino acid indices of AAindex database into eight clusters, and eight high-quality amino acid indices in each cluster were extracted (Saha et al. 2012). We denoted the eight indices as HQI1 to HQI8 (Supplementary File 1), and compared the mean values of them between the ubiquitination and non-ubiquitination peptides, which are displayed as Fig. 2. HQI1 denotes electric properties. It is observed that the residues near ubiquitination sites carry more electric charge than those near non-ubiquitination sites. HQI2 represents amino acid hydrophobicity. Ubiquitination peptides are less hydrophobic than non-ubiquitination peptides. HQI3 denotes alpha helix and turn propensities, and HQI7 denotes beta-strand propensities. In general, residues in the ubiquitination peptides tend to form alpha helices and turns, while residues in the non-ubiquitination peptides tend to form beta strands. HQI4 represents the amino acid volumes. The volumes of residues in some positions that are closest to the ubiquitination sites are smaller than those of the non-ubiquitination sites. The volumes of residues in other positions are larger than those of the non-ubiquitination sites. HQI5 represents transmembrane residue propensities. Residues in the ubiquitination peptides are disfavored to be localized in the transmembrane regions. HQI6 represents the amino acid compositions of intracellular proteins. Positions closest to the ubiquitination sites prefer more frequently occurring residues (such as Ala, Gly and Glu). HQI8 represents the relative partition energies of residues. Residues in the ubiquitination peptides have a stronger tendency to contact with other residues than those in the non-ubiquitination peptides.

Prediction performance

Prediction performance of CNN models in five-fold cross-validation and independent test

We use five-fold cross-validation and independent test to assess different models. AraUbiSite is the first ubiquitination site prediction tool specifically for *A. thaliana* (Chen et al. 2019). It uses binary encoding, amino acid composition (AAC) and composition of *k*-spaced amino acid pairs (CKSAAP) as input features, and an SVM as training method to build the model. Because AraUbiSite provides the prediction scores in five-fold cross-validation and independent test in its webserver, and uses the same dataset as our methods, it is feasible to directly compare our methods with it. Figure 3 shows the ROC curves of CNN_binary, CNNProperty and AraUbiSite in five-fold cross-validation (left panel) and independent test (right panel). From the ROC curves, we can see that CNN_Binary and CNN_Property perform much better than AraUbiSite, and CNN_Binary performs a little better than CNN_Property. This may suggest that binary encoding can better represent the original information of the sequence, and is more conducive to the convolution layer extracting local features for prediction.

The other metrics for performance of the three methods in five-fold cross-validation and independent test are listed in Tables 2 and 3 respectively. The sensitivity values of CNN_Binary and CNN_Property are larger than that of AraUbiSite, but their specificity values are smaller than that of AraUbiSite. This is because CNN_Binary and CNN_Property predict more ubiquitination sites, which simultaneously increases the numbers of true positives and false positives. The accuracy of CNN_Binary and CNN_Property is much larger than that of AraUbiSite. This demonstrates that CNN_Binary and CNN_Property perform better than AraUbiSite. AUC can better reflect the performance of different methods. Both in five-fold cross-validation and independent test, the AUC values of CNN_Binary and

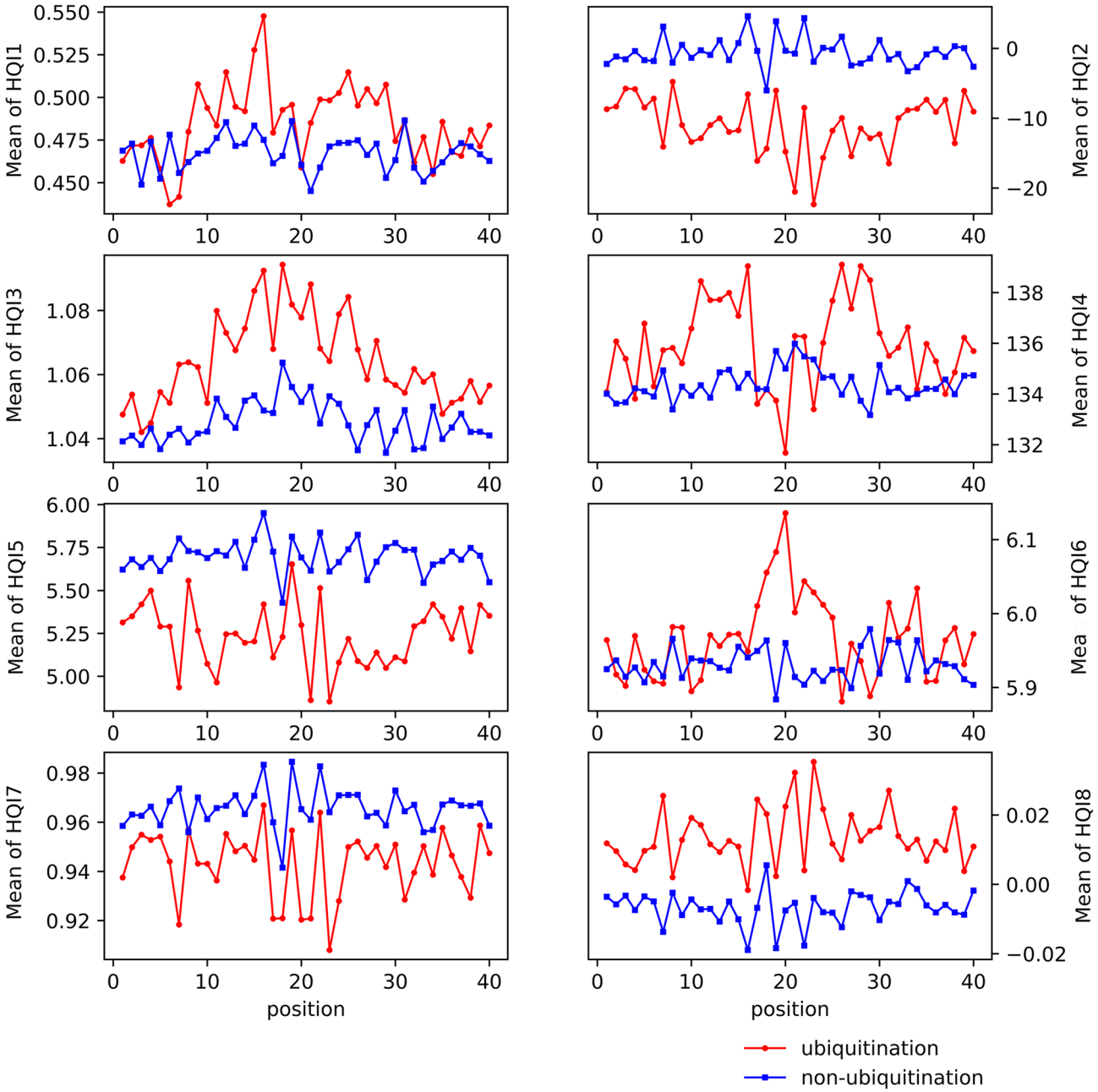

Fig. 2 Comparison of eight high-quality amino acid indices (HQI) between ubiquitination and non-ubiquitination peptides. The eight indices from HIQ1 to HIQ8 represent electric charge properties, hydrophobicity, alpha helix and turn propensities, volume, transmembrane residue propensities, amino acid composition, beta strand propensities, and relative partition energies, respectively. Red dots represent ubiquitination peptides, while blue dots denote non-ubiquitination peptides. Positions from 1 to 20 are the left positions of the ubiquitination site, and positions from 21 to 40 are the right positions of the ubiquitination sites

CNN_Property are larger than that of AraUbiSite, further demonstrating the competitive edge of the two CNN models.

Sensitivity values at high specificity

In *A. thaliana* proteome, the amount of ubiquitination sites is much less than that of non-ubiquitination sites. Therefore, experimental scientists will pay more attention to predicted ubiquitination sites with high scores to avoid the experimental trial-and-error cost. Table 4 displays the sensitivity values of the three models at specificity of 95 % (false positive rate of 5 %) in five-fold cross-validation. The sensitivity of CNN_Binary, CNN_Property and AraUbiSite are 48.83 %, 46.54 % and 7.7 % respectively. This suggests that when 5 %

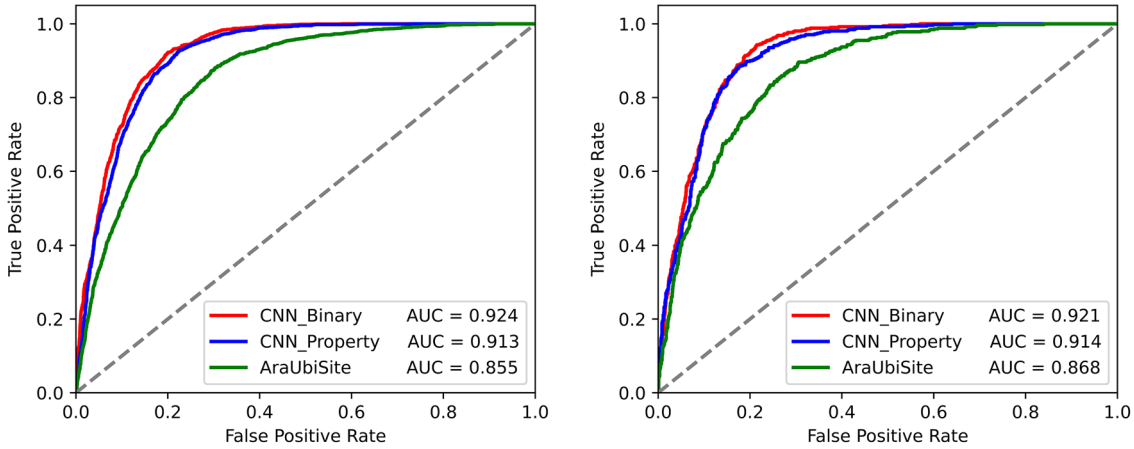

Fig. 3 ROC curves of different methods. **a** ROC curves in five-fold cross-validation. **b** ROC curves in independent test (right panel). AUC is area under the ROC curve

Table 2 Prediction performance in five-fold cross-validation

Method	Sensitivity	Specificity	Accuracy	AUC
CNN_Binary	88.05	82.71	85.38	0.924
CNN_Property	84.92	83.60	84.26	0.913
AraUbiSite	53.33	91.32	69.19	0.855

Table 3 Prediction performance in independent test

Method	Sensitivity	Specificity	Accuracy	AUC
CNN_Binary	89.24	81.67	85.36	0.921
CNN_Property	88.65	82.06	85.45	0.914
AraUbiSite	51.27	91.38	71.33	0.868

Table 4 Sensitivity values at specificity of 95 % for different methods

Method	CNN_Binary (%)	CNN_Property (%)	AraUbiSite (%)
sensitivity	48.83	46.54	7.70

non-ubiquitination sites are wrongly predicted as ubiquitination sites, the correctly predicted ubiquitination sites by CNN_Binary, CNN_Property and AraUbiSite account for 48.83 %, 46.54 % and 7.7 % of the true ubiquitination sites respectively.

Comparison of predicted ubiquitination sites by CNN_Binary, CNN_Property and AraUbiSite

Because biologists are mainly concerned with high-confidence predicted ubiquitination sites, to analyze the overlap of the predictions of CNN_Binary, CNN_Property and AraUbiSite, we plot the Venn diagrams based on the top 100, 300, 500 and 1000 predicted ubiquitination sites in five-fold cross-validation for the three methods, which are displayed in Fig. 4. From the figure, we can see that CNN_Binary and CNN_Property have more overlap predictions. The overlap predictions between AraUbiSite and the other two methods are less. This may be because both CNN_Binary and CNN_Property are CNN models, while AraUbisite is an SVM model. From the figure, we can also see that with the increase of the predicted ubiquitination site, the ratio of the overlap predictions for each model increases, too.

Comparison with non-convolutional neural network and random forest models

We use binary encoding, physicochemical properties, AAC, and CKSAAP as inputs of non-convolutional neural network (NCNN) and random forest (RF) models, to test their prediction performance. Unlike the input of the CNN model, these features are fed to the models in the form of vectors. The details of the features and models are described in Supplementary File 2. Table 5 shows their AUC values in 5-fold cross-validation, which are lower than those of the two CNN models. This proves that CNN is more competitive for predicting ubiquitination sites based on sequence information.

The effect of the CNN structure on the prediction performance

In the two CNN models, the convolutional layer and max-pooling layer are responsible for capturing important local properties, while the fully connected hidden layer is responsible for effectively combining the local properties.

We remove the convolutional layer and the maximum pooling layer from the two CNNs to study their importance. After adjusting the hyper parameters, the maximum

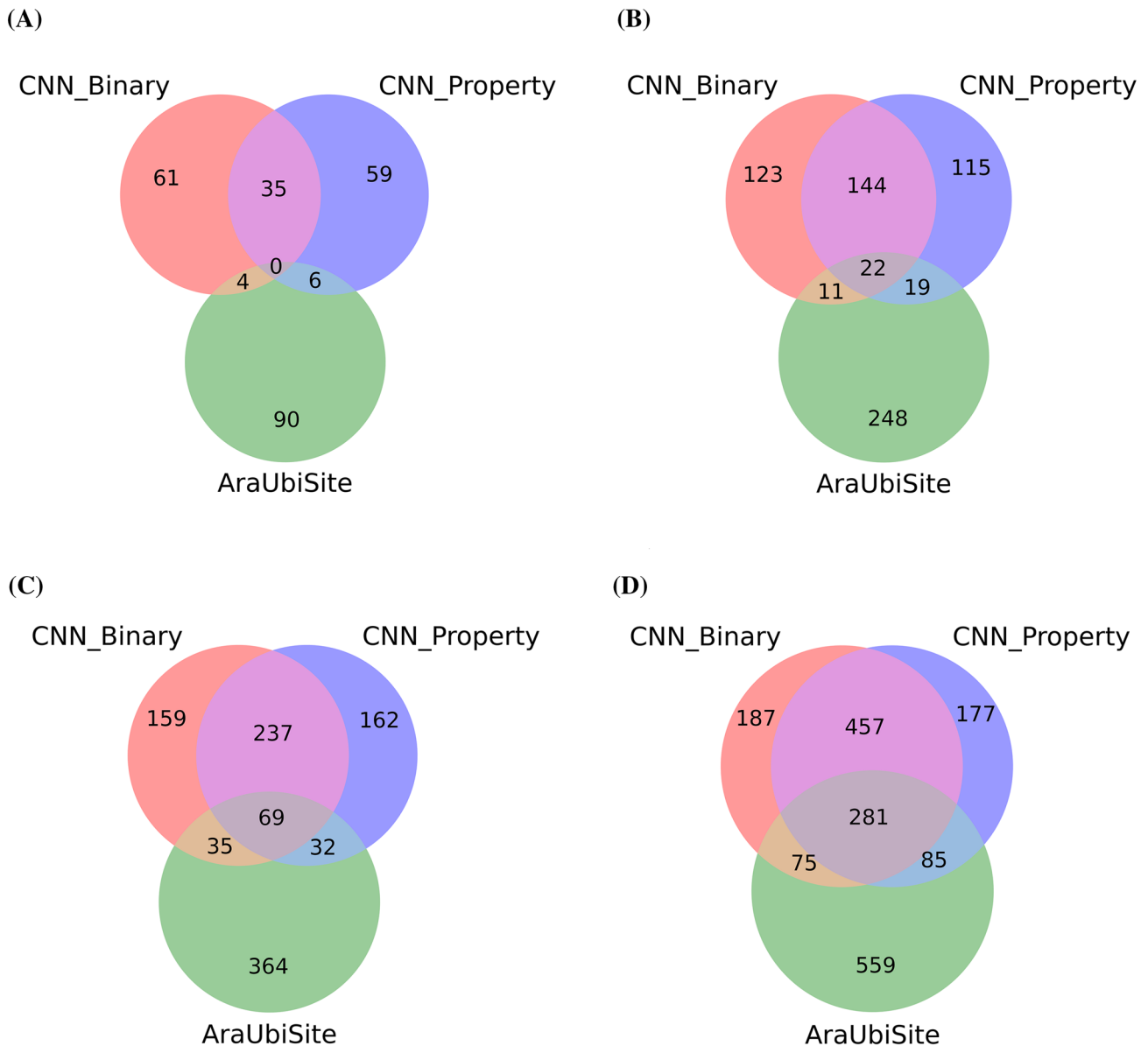

Fig. 4 Venn diagrams of the top predicted ubiquitination sites. Panels **a**, **b**, **c**, **d** respectively plot Venn diagrams of the top 100, 300, 500 and 1000 predicted ubiquitination sites between CNN_Binary, CNN_Property and AraUbiSite in five-fold cross-validation

Table 5 AUC values of NCNN and RF in 5-fold cross-validation

Feature/Method	Binary encoding	Physicochemical property	AAC	CKSAAP
NCNN	0.862	0.789	0.814	0.822
RF	0.818	0.811	0.809	0.827

AUC values in five-fold cross-validation are 0.862 and 0.789 for the binary and property encodings respectively, which are lower than those of the two original CNN models.

We also remove the fully connected hidden layers from the two CNN models to study their importance. Under the optimal hyper parameter combinations, the AUC values in five-fold cross-validation are 0.859 and 0.847 for the binary and property encodings, respectively. Compared with the original models, the performance also declines.

The above results indicate that the fully connected hidden layer, convolutional layer and maximum pooling layer in the CNN models are very important for the prediction performance.

Prediction of ubiquitination sites in *A. thaliana* proteome

In view of the excellent prediction performance of the two CNN models, we use them to predict the potential ubiquitination sites in *A. thaliana* proteome to help experimental scientists. Finally, CNN_Binary and CNN_Property respectively predict 224,195 and 329,678 potential ubiquitination sites in *A. thaliana* proteome. We save the predicted ubiquitination sites by CNN_Binary and CNN_Property into two tables with protein IDs, site positions, 40-residue peptides and prediction scores as columns, which are available at GitHub (http://github.com/nongdaxiaofeng/CNNAthUbi). Two lists of prediction thresholds, true positive rates (sensitivity), and false positive rates in five-fold cross-validation for the two CNN models are also available at GitHub for users to better make use of the predicted ubiquitination sites.

Discussion

CNN_Binary and CNN_Property use binary encoding and amino acid physicochemical properties respectively to achieve better prediction performance than other models. This is mainly because the convolutional kernel of CNN can scan the sequences represented by binary encodings or amino acid properties to find useful local patterns for prediction. At the same time, the fully connected layer in the network can learn the global pattern. According to the theory of deep learning (LeCun et al. 2015), more useful features can be learned by increasing the number of layers of neural network, so as to improve the accuracy of prediction. But when we increased the layer numbers, the performance of the two CNN models was not improved, which may be because the task of ubiquitination site prediction is not complex, and only through one or two layers can we extract effective features. We also tried to average the prediction scores of CNN_Binary and CNN_Property, but can't obtain better results.

Sequence and structure analysis of human and yeast proteins with ubiquitination sites were performed by a previous study (Kumar and Vellaichamy 2019). In the study, the ubiquitination site positions at 0. Negative and positive integers represented left and right positions of the ubiquitination site in the sequence. In their analysis, R is most preferred from position − 13 to − 7 and from position 7 to 14, but is least preferred in close proximity to the ubiquitination site in human. In the sequence analysis of AraUbiSite, R is the most frequently occurring amino acid in almost all the positions from − 20 to 20, which is different from that of human. At the same time, eight models which are trained using ubiquitination sites from human, mouse and yeast, were used to predict ubiquitination sites of *A. thaliana*, resulting poor performance equivalent to random prediction (Chen et al. 2015). These results suggest that the ubiquitination sites of different species have their own characteristics, and it's necessary to establish species-specific prediction model.

In this paper, we adopt CNNs to establish two ubiquitination site prediction models for *A. thaliana*, and prove that they are significantly better than other models. There are other types of neural networks that are also very suitable for processing sequences, such as recurrent neural networks (Vaferi et al. 2015). Perhaps they can be used to build prediction models with good performance. Integrating different types of neural networks to construct a complex neural network may also improve the prediction performance. This probably is a direction to build good or better models in the future. The performance of the model is also related to the quantity and quality of training data. Generally, a larger training set will learn a more accurate prediction model. The negative samples in the data set are lysine residues which are not experimentally confirmed to be ubiquitination sites, but some of them may be true ubiquitination sites. This can reduce the prediction performance of the model. With the development of experimental technologies, more and more ubiquitination sites in *A. thaliana* will be identified, and models with better performance can be established.

Conclusions

In this paper, we present two CNN models which use binary encoding and amino acid physicochemical properties as inputs respectively for predicting ubiquitination sites in *A. thaliana*. It is demonstrated that the two models perform much better than the previously reported SVM-based model AraUbiSite, and some other machine learning models. Through the analysis of the physicochemical properties of residues surrounding the ubiquitination site, we found that compared with residues near non-ubiquitination sites, residues near ubiquitination sites carry more electric charge, are less hydrophobic, prefer to form alpha helices and turns, and tend to contact with other residues. We study the structure of the two CNN models, and prove that both the convolutional layer and the fully connected hidden layer are indispensable in the models for improving the prediction performance. The source code of CNN_Binary and CNN_Property, training and test datasets, and predicted ubiquitination sites in *Arabidopsis* are freely available at GitHub (http://github.com/nongdaxiaofeng/CNNAthUbi) for interest researchers. It is believed that with more and more ubiquitination sites in *A. thaliana* identified experimentally, so as to expand the size of the training set, and the development of machine learning technologies, more excellent prediction tools will emerge in the future.

Acknowledgements This work was supported by the National Natural Science Foundation of China (41801027).

Author contributions XW devised the method and drafted the paper. RY modified the paper. YZC and Y Wang tested the source code.

References

Brodersen KH, Ong CS, Stephan KE, Buhmann JM The Balanced accuracy and its posterior distribution. In: 2010 20th International Conference on Pattern Recognition, 23–26 Aug (2010) pp. 3121–3124. https://doi.org/10.1109/ICPR.2010.764

Cai B, Jiang X (2016) Computational methods for ubiquitination site prediction using physicochemical properties of protein sequences. BMC Bioinform 17:116. https://doi.org/10.1186/s12859-016-0959-z

Chen Z, Chen Y-Z, Wang X-F, Wang C, Yan R-X, Zhang Z (2011) Prediction of ubiquitination sites by using the composition of k-spaced amino acid pairs. PLoS ONE 6:e22930. https://doi.org/10.1371/journal.pone.0022930

Chen Z, Zhou Y, Zhang Z, Song J (2015) Towards more accurate prediction of ubiquitination sites: a comprehensive review of current methods, tools and features. Brief Bioinform 16:640–657. https://doi.org/10.1093/bib/bbu031

Chen J, Zhao J, Yang S, Chen Z, Zhang Z (2019) Prediction of protein ubiquitination sites in *Arabidopsis thaliana*. Curr Bioinform 14:614–620. https://doi.org/10.2174/1574893614666190311141647

Chu KH (2020) Exponential and logistic functions: the two faces of the Bohart-Adams model. J Hazard Mater 389:122025. https://doi.org/10.1016/j.jhazmat.2020.122025

Dowsing RD (1970) Use of grid search techniques to extend the use of a least squares program for analysis of electron spin resonance spectra. J Comput Phys 6:326–328. https://doi.org/10.1016/0021-9991(70)90030-6

Fawcett T (2006) An introduction to ROC analysis. Pattern Recogn Lett 27:861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Feng K, Huang T, Feng K, Liu X (2013) Using WPNNA classifier in ubiquitination site prediction based on hybrid features. Protein Pept Lett 20:318–323. https://doi.org/10.2174/0929866511320030010

Fu HL, Yang YX, Wang XB, Wang H, Xu Y (2019) DeepUbi: a deep learning framework for prediction of ubiquitination sites in proteins. BMC Bioinform 20:1–10. https://doi.org/10.1186/s12859-019-2677-9

Glickman MH, Ciechanover A (2002) The ubiquitin-proteasome proteolytic pathway: destruction for the sake of construction. Physiol Rev 82:373–428. https://doi.org/10.1152/physrev.00027.2001

Herrmann J, Lerman LO, Lerman A (2007) Ubiquitin and ubiquitin-like proteins in protein regulation. Circ Res 100:1276–1291. https://doi.org/10.1161/01.RES.0000264500.11888.f0

Kim HJ, Oh SA, Brownfield L et al (2008) Control of plant germline proliferation by SCF(FBL17) degradation of cell cycle inhibitors. Nature 455:1134–1137. https://doi.org/10.1038/nature07289

Krizhevsky A, Sutskever I, Hinton GE (2017) ImageNet classification with deep convolutional neural networks. Commun ACM 60:84–90. https://doi.org/10.1145/3065386

Kumar VS, Vellaichamy A (2019) Sequence and structure-based characterization of ubiquitination sites in human and yeast proteins using Chou's sample formulation. Proteins: Struct Funct Bioinf 87:646–657. https://doi.org/10.1002/prot.25689

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521:436–444. https://doi.org/10.1038/nature14539

Lee H, Song J (2019) Introduction to convolutional neural network using Keras; an understanding from a statistician. Commun Stat Appl Methods 26:591–610. https://doi.org/10.29220/csam.2019.26.6.591

Lu D, Lin W, Gao X et al (2011) Direct ubiquitination of pattern recognition receptor FLS2 attenuates plant innate immunity. Science 332:1439–1442. https://doi.org/10.1126/science.1204903

Maier A, Schrader A, Kokkelink L et al (2013) Light and the E3 ubiquitin ligase COP1/SPA control the protein stability of the MYB transcription factors PAP1 and PAP2 involved in anthocyanin accumulation in Arabidopsis. Plant J 74:638–651. https://doi.org/10.1111/tpj.12153

Nhu V-H, Hoang N-D, Nguyen H et al (2020) Effectiveness assessment of Keras based deep learning with different robust optimization algorithms for shallow landslide susceptibility mapping at tropical area. CATENA 188:104458. https://doi.org/10.1016/j.catena.2020.104458

Ramos D, Franco-Pedroso J, Lozano-Diez A, Gonzalez-Rodriguez J (2018) Deconstructing cross-entropy for probabilistic binary classifiers. Entropy 20:208. https://doi.org/10.3390/e20030208

Rampasek L, Goldenberg A (2016) TensorFlow: biology's gateway to deep learning? Cell Syst 2:12–14. https://doi.org/10.1016/j.cels.2016.01.009

Saha I, Maulik U, Bandyopadhyay S, Plewczynski D (2012) Fuzzy clustering of physicochemical and biochemical properties of amino acids. Amino Acids 43:583–594. https://doi.org/10.1007/s00726-011-1106-9

Schnell JD, Hicke L (2003) Non-traditional functions of ubiquitin and ubiquitin-binding proteins. J Biol Chem 278:35857–35860. https://doi.org/10.1074/jbc.R300018200

Song L, Li D, Zeng X, Wu Y, Guo L, Zou Q (2014) nDNA-prot: identification of DNA-binding proteins based on unbalanced classification. BMC Bioinform 15:298. https://doi.org/10.1186/1471-2105-15-298

Tibshirani R (1996) Regression shrinkage and selection via the Lasso. J Roy Stat Soc Ser B 58:267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Tung C-W, Ho S-Y (2008) Computational identification of ubiquitylation sites from protein sequences. BMC Bioinform 9:310. https://doi.org/10.1186/1471-2105-9-310

Vaferi B, Eslamloueyan R, Ayatollahi S (2015) Application of recurrent networks to classification of oil reservoir models in well-testing analysis. Energy Sour Part A 37:174–180. https://doi.org/10.1080/15567036.2011.582610

Wagner SA, Beli P, Weinert BT, Nielsen ML, Cox J, Mann M, Choudhary C (2011) A proteome-wide, quantitative survey of in vivo ubiquitylation sites reveals widespread regulatory roles. Mol Cell Proteomics 10:M111013284. https://doi.org/10.1074/mcp.M111.013284

Walton A, Stes E, Cybulski N et al (2016) It's time for some "site"-seeing: novel tools to monitor the ubiquitin landscape in *Arabidopsis thaliana*. Plant Cell 28:6–16. https://doi.org/10.1105/tpc.15.00878

Zhang GQ, Patuwo BE, Hu MY (1998) Forecasting with artificial neural networks: the state of the art. Int J Forecast 14:35–62. https://doi.org/10.1016/s0169-2070(97)00044-7