Plant Methods

METHODOLOGY

Open Access

PlantNh-Kcr: a deep learning model for predicting non-histone crotonylation sites in plants

Yanming Jiang[1], Renxiang Yan[2,3] and Xiaofeng Wang[1*]

Abstract

Background Lysine crotonylation (Kcr) is a crucial protein post-translational modification found in histone and non-histone proteins. It plays a pivotal role in regulating diverse biological processes in both animals and plants, including gene transcription and replication, cell metabolism and differentiation, as well as photosynthesis. Despite the significance of Kcr, detection of Kcr sites through biological experiments is often time-consuming, expensive, and only a fraction of crotonylated peptides can be identified. This reality highlights the need for efficient and rapid prediction of Kcr sites through computational methods. Currently, several machine learning models exist for predicting Kcr sites in humans, yet models tailored for plants are rare. Furthermore, no downloadable Kcr site predictors or datasets have been developed specifically for plants. To address this gap, it is imperative to integrate existing Kcr sites detected in plant experiments and establish a dedicated computational model for plants.

Results Most plant Kcr sites are located on non-histones. In this study, we collected non-histone Kcr sites from five plants, including wheat, tabacum, rice, peanut, and papaya. We then conducted a comprehensive analysis of the amino acid distribution surrounding these sites. To develop a predictive model for plant non-histone Kcr sites, we combined a convolutional neural network (CNN), a bidirectional long short-term memory network (BiLSTM), and attention mechanism to build a deep learning model called PlantNh-Kcr. On both five-fold cross-validation and independent tests, PlantNh-Kcr outperformed multiple conventional machine learning models and other deep learning models. Furthermore, we conducted an analysis of species-specific effect on the PlantNh-Kcr model and found that a general model trained using data from multiple species outperforms species-specific models.

Conclusion PlantNh-Kcr represents a valuable tool for predicting plant non-histone Kcr sites. We expect that this model will aid in addressing key challenges and tasks in the study of plant crotonylation sites.

Keywords Crotonylation, Convolutional neural network, Bidirectional long short-term memory, Attention mechanism, Focal loss

*Correspondence:
Xiaofeng Wang
wangxf@sxnu.edu.cn
[1] College of Mathematics and Computer Sciences, Shanxi Normal University, Taiyuan 030031, China
[2] The Key Laboratory of Marine Enzyme Engineering of Fujian Province, Fuzhou University, Fuzhou 350002, China
[3] College of Biological Science and Engineering, Fuzhou University, Fuzhou 350002, China

Introduction

Post-translational modifications (PTMs) [1] of proteins involve the addition or removal of chemical groups to amino acid residues, thereby modifying protein properties and expanding functional diversity. PTMs play crucial roles in various biological processes and metabolic pathways. Among the PTMs, lysine crotonylation (Kcr) is a novel and significant modification that has been widely detected in both animals and plants. The Kcr

modification was initially discovered in the histones of human somatic cells and mouse germ cells, and Kcr enrichment on sex chromosomes has been identified as a key indicator in male germ cell differentiation control [2]. In mice, it was shown that increased histone crotonylation level might have a positive effect on acute kidney injury [3]. In addition, histone Kcr sites are involved in many other biological processes, including organism development [2], DNA damage response [4], and gene transcription and expression [5]. Kcr is not limited to histones but is also abundant in non-histones [6–9]. Previous studies have clearly revealed that crotonylation of non-histone proteins is associated with various metabolic pathways and participates in protein expression and multiple cell signaling cascades [8].

In plants, global identification and functional analysis of lysine crotonylation have been conducted in species such as tabacum [10], papaya [11], rice [12], peanut [13], and wheat [14, 15]. The Kcr modification is involved in regulating various metabolic pathways in plants, including photosynthesis, oxidative phosphorylation, and carbon metabolism [14]. Additionally, Kcr is involved in gene transcription regulation [12] and adaptation to adverse conditions in plants [16]. Notably, Kcr is related to cold stress tolerance in plants [17], exhibiting a positive regulatory effect on wheat's freezing tolerance [14].

The current experimental methods for detecting Kcr sites include high-performance liquid chromatography fractionation, stable isotope labelling of amino acids in cell culture, immunological affinity enrichment, and high-resolution liquid chromatography-tandem mass spectrometry [18]. While biological experiments are the most reliable means to identify Kcr sites, the experiments are often time-consuming, labor-intensive, and costly. In addition, mass spectrometry platforms can only identify a subset of crotonylated peptides due to factors such as protein abundance, protein hydrolysis and digestion [19]. Therefore, computational models for conveniently and rapidly predicting Kcr sites are desirable, which have been developed in the past few years.

Early models for Kcr site prediction were limited by the small training datasets with fewer than 200 Kcr sites, and all sites were limited to histones. These models employed conventional machine learning methods such as support vector machines [20], random forest (RF) [21], LightGBM [22–24], etc. The input features used by these models include composition of amino acids and amino acid pairs, amino acid properties, etc. The representative models include CrotPred [25], CKSAAP_CrotSite [26], iKcr-PseEns [27], iCrotoK-PseAAC [28], LightGBM-CroSite [29], etc.

With the advancement of mass spectrometry technology, the global Kcr sites in the proteome of several species have been detected that enabled the utilization of significantly larger training datasets. At the same time, the deep learning frameworks [30] have reached a level of maturity that resulted in the common employment of deep learning methods in establishing predictive models for Kcr sites. Among these models, some predict Kcr sites on a mixture of histones and non-histones, such as Deep-Kcr [31], BERT-Kcr [32], DeepCap-Kcr [33], Adapt-Kcr [34], and ATCLSTM-Kcr [19]. Others are tailored to predict Kcr sites on non-histones, such as nhKcr [35], DeepKcrot [36], iKcr_CNN [37], and CapsNh-Kcr [38]. The primary input features of these models consist of binary encoding and embedding vectors. A majority of these models utilize convolutional neural networks (CNNs) [19, 31, 33–38] and long short-term memory networks (LSTM) [19, 32–34, 36] as integral components of their structure. Notably, some models integrate self-attention mechanism to enhance their predictive capabilities [19, 32, 34]. Additionally, a few models stand out due to their unique network architecture. For instance, DeepCap-Kcr and CapsNh-Kcr employ capsule networks, a distinctive approach in deep learning architecture. The deep learning models have resulted in significantly improved performance compared to the earlier conventional machine learning models.

Among the four models for predicting Kcr sites on non-histones, nhKcr, iKcr_CNN, and CapsNh-Kcr are limited to predicting human Kcr sites. In contrast, DeepKcrot predict Kcr sites in four species, including humans, rice, tabacum, and papaya. However, it is important to note that due to the current unavailability of DeepKcrot's web server, accessing its datasets and models for further research purposes can be challenging. In addition, recent experimental studies have detected Kcr sites in some other plants, emphasizing the need for a computational model that is specifically tailored for plants. To address this gap, it is essential to integrate existing Kcr site data detected from plants and establish a specialized computational model dedicated to plants.

In this study, we compiled a comprehensive dataset of non-histone Kcr sites from five plant species including rice, tabacum, papaya, peanut, and wheat, and built a reliable training and test dataset. Then we utilized the binary encoding (BE) as input features and employed a combination of a convolutional neural network (CNN) [39], a bidirectional long short-term memory (BiLSTM) network [40] and multi-head self-attention (MHSA) [41] to construct a novel deep learning model called PlantNh-Kcr. This model was specifically designed to predict Kcr sites on non-histones in plants. We validated our model through rigorous comparisons with conventional machine learning methods and other state-of-the-art deep learning

models. On five-fold cross-validation and independent test, PlantNh-Kcr consistently demonstrated superior performance. Furthermore, it excelled in predicting Kcr sites across individual plant species, highlighting its remarkable versatility and generalizability. We believe that the development of this plant-specific prediction model offers valuable insights for the biological community and will drive further advancement in plant biology.

Materials and methods

Benchmark dataset

To train and test the model, we carefully curated a training dataset and a test dataset. This process was meticulously designed and is depicted in Fig. 1. We first collected non-histone Kcr sites from five plant species. These sites numbered 5692 from wheat [14, 15], 1258 from rice [12], 2028 from tabacum [10], 6603

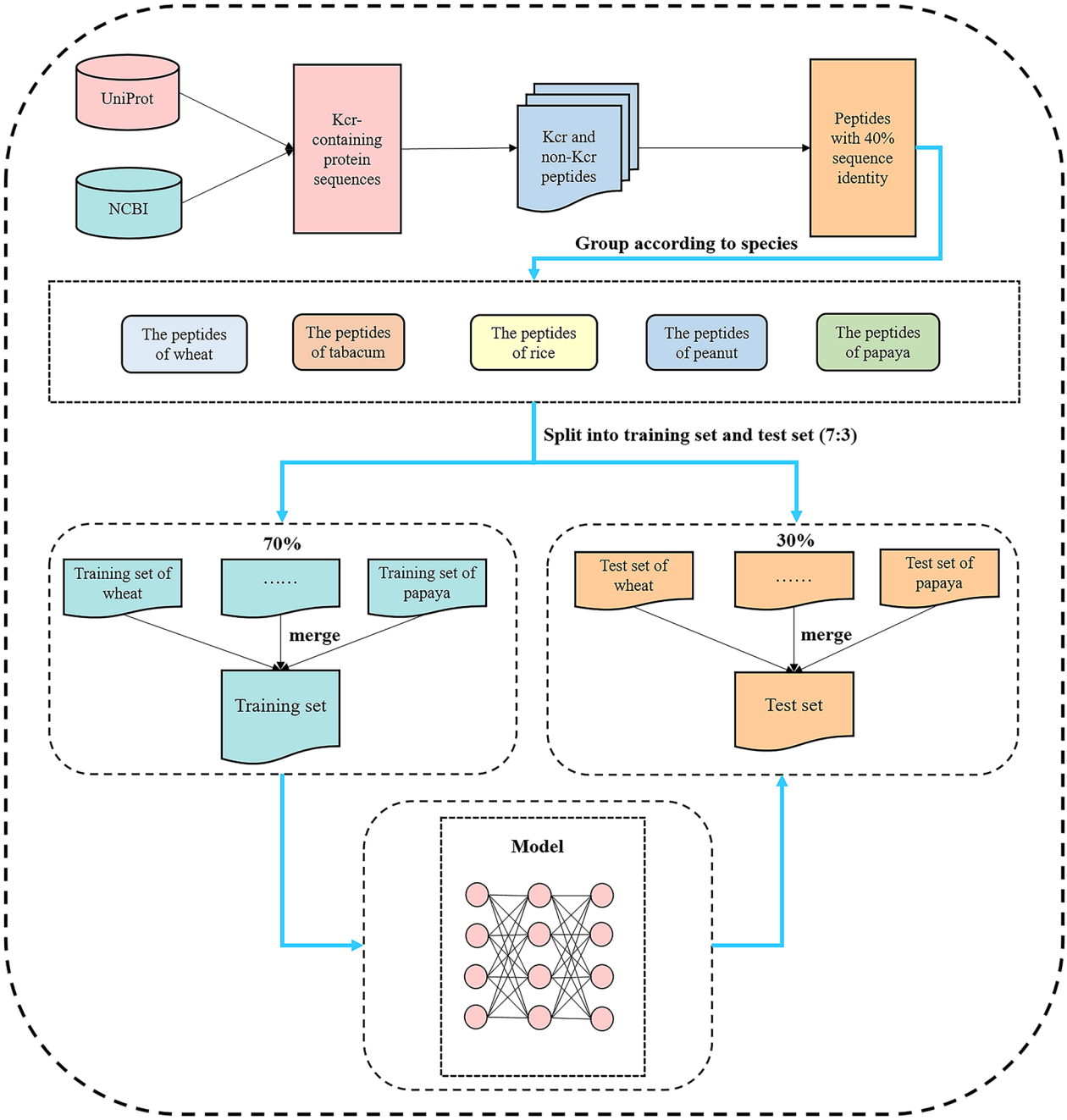

Fig. 1 The flowchart of dataset preparation

from peanut [13], and 5332 from papaya [11]. We then retrieved the corresponding protein sequences from the UniProt [42] and NCBI [43] databases for each species. Subsequently, we extracted peptides of length 29 from these protein sequences, with the K (Lysine) residue positioned at the center and 14 residues upstream and downstream respectively. If a peptide had fewer than 14 residues on one side, we replaced the missing residues with X. Peptides where the central K residues represented Kcr sites were designated as positive samples; otherwise, they were designated as negative samples. To eliminate redundancy and potential false negatives, we used the CD-HIT program [44] with a sequence identity threshold of 40%. Finally, we obtained 12,352 positive and 46,389 negative samples. To evaluate the performance of the model on test samples of each species, we separated the samples based on their species. For each species, the samples were randomly divided into two subsets in a 7:3 ratio while retaining the proportion of positive and negative samples during the partitioning process. The larger sets for each species were merged to form the training dataset and the smaller sets for each species were merged to form the test dataset. The training dataset totaled 41,114 samples, with 8644 positive and 32,470 negative samples. The test dataset included 17,627 samples, with 3708 positive 13,919 negative samples. The numbers of samples for each species in the training and test datasets are listed in Table 1.

Peptide encoding

To build a predictive model for non-histone Kcr sites, it is necessary to transform peptide samples into numerical vectors as input features for the model. PlantNh-Kcr employs binary encoding as its input features. We conducted a comparative analysis of PlantNh-Kcr's predictive performance against conventional machine learning models and other deep learning models. The conventional machine learning models utilize various input features, including amino acid composition (AAC), enhanced group amino acid composition (EGAAC), BE, AAindex encoding, and BLOSUM62 encoding. Other deep learning models employ features including BE, word embedding (WE) encoding, AAindex encoding, and BLOSUM62 encoding. The following provides a detailed description of these encoding methods:

AAC: In bioinformatics, AAC is a commonly used encoding method, which calculates the frequencies of each amino acid in a peptide. In this study, X is also considered as an amino acid. So the peptide is encoded as a 21-dimensional vector, where each dimension corresponds to the frequency of one of the 21 amino acids present in the peptide.

EGAAC: The EGAAC encoding divides amino acids into five groups based on their physicochemical properties, i.e. aliphatic group (GAVLMI), aromatic group (FYW), positively charged group (KRH), negatively charged group (DE), and no charge group (STCPNQ). A peptide is encoded as a five-dimensional vector, where each dimension represents the proportion of one of the five groups of amino acids within the peptide.

BE: BE is a common technique used to convert amino acid sequences into numerical representations suitable for model training. For this encoding method, each amino acid is encoded as a 21-dimensional binary vector. This vector has one component set to 1 to indicate the type of amino acid, while all other components are set to 0. Finally, a peptide of length 29, is encoded as a matrix or a vector of size 29×21.

WE encoding: WE is a technique that has gained popularity in the field of natural language processing. It assigns words to vectors in a high-dimensional space, ensuring that semantically similar words are positioned close to each other. This technique has also been effectively applied to sequence encoding in bioinformatics [45, 46]. In this work, the vocabulary size is set to 21, representing the number of amino acid types. The peptide of length 29 is treated as a sentence, with each amino acid residue mapped to a unique word ID. Subsequently, WE is used to translate these IDs into vectors. Finally, the peptide is encoded as a matrix of size 29×10, where 10 is the dimension of the vector space.

AAindex encoding: AAindex [47] is a public database that curates a range of physicochemical and biochemical properties of amino acids. This database serves as a valuable resource for various bioinformatics studies, including protein structure prediction, sequence alignment, protein function annotation, and more. Previously, the model nhKcr selected 29 indices from AAindex that were most relevant to the prediction task to encode the peptide [35]. In this study, we used the same 29 indices to encode the peptide. Consequently, the peptide of length 29 is encoded as a matrix or a vector of size 29×29.

Table 1 The numbers of positive and negative samples for each species in the training and test datasets

Species	Training set		Test set	
	Positive	Negative	Positive	Negative
Wheat	2484	7585	1066	3252
Tabacum	820	2524	352	1082
Rice	662	3486	284	1495
Peanut	2452	10,675	1051	4575
Papaya	2226	8200	955	3515
Total samples	8644	32,470	3708	13,919

BLOSUM62 encoding: BLOSUM62 (BLOck Substitution Matrix 62) [48], is a widely used substitution matrix in protein sequence alignment. The matrix assigns scores to pairs of amino acids based on their substitution frequency during evolution. Higher scores indicate more frequent substitutions, while lower scores indicate rare substitutions. In this study, we used the rows of the BLUSUM62 matrix to encode amino acids in the peptide. Consequently, the peptide of length 29 is encoded as a matrix or a vector of size 29×21.

The structures of the plantNh-Kcr model

The structure of PlantNh-Kcr was determined as Fig. 2 after evaluating various encoding methods and model architectures. The model accepts a 29×21 matrix derived from binary encoding as input. This matrix feeds into two distinct layers. The first is a convolutional layer that is followed by two additional convolutional layers. The second layer is a BiLSTM layer that is succeeded by a MHSA layer. The outputs of the third convolutional layer and the MHSA layer are merged and flattened into a vector. The flatten layer is followed by a linear layer and an output layer. All the layers are described in detail below.

Input layer: The layer receives a 29×21 matrix as input.

Convolutional layers: The first convolutional layer has 21 input channels and 32 output channels, with a kernel size of 5 and a stride of 1. The second convolutional layer has 32 input channels and 32 output channels, and the third one has 32 input channels and 29 output channels. Both the latter two layers have a kernel size of 5 and a stride of 2. The outputs of each layer are activated using the ReLU function [49]. During training, to prevent overfitting, 30% of the output data from the three convolution layers are dropped respectively.

BiLSTM layer and MHSA layer: The input size of the BiLSTM layer is 21, and the output size is 128. The MHSA layer has an input size of 128 and eight attention heads. To prevent overfitting, dropout operations with ratios of 0.9 and 0.5 are applied after the BiLSTM layer and MHSA layer, respectively.

Flatten layer: The flatten layer flattens the concatenated outputs of the third convolutional layer and MHSA layer, resulting in a 3944-dimensional vector.

Linear layer: The input size of the linear layer is 3944 and the output size is 128. The output is activated using the ReLU function.

Output layer: The output layer has an input size of 128 and output size of 2. The two-dimensional output vector represents the probabilities of a sample being positive and negative, respectively.

Focal loss

In this study, the training dataset has significantly more negative samples than positive samples, which would lead to a bias towards the negative samples during model training. To address this issue, we employed focal loss [50] as the loss function for the model. Focal loss reshapes the traditional cross-entropy loss function by introducing a modulating factor $(1 - p_t)^{\gamma}$. The mathematical formulas for focal loss are as follows:

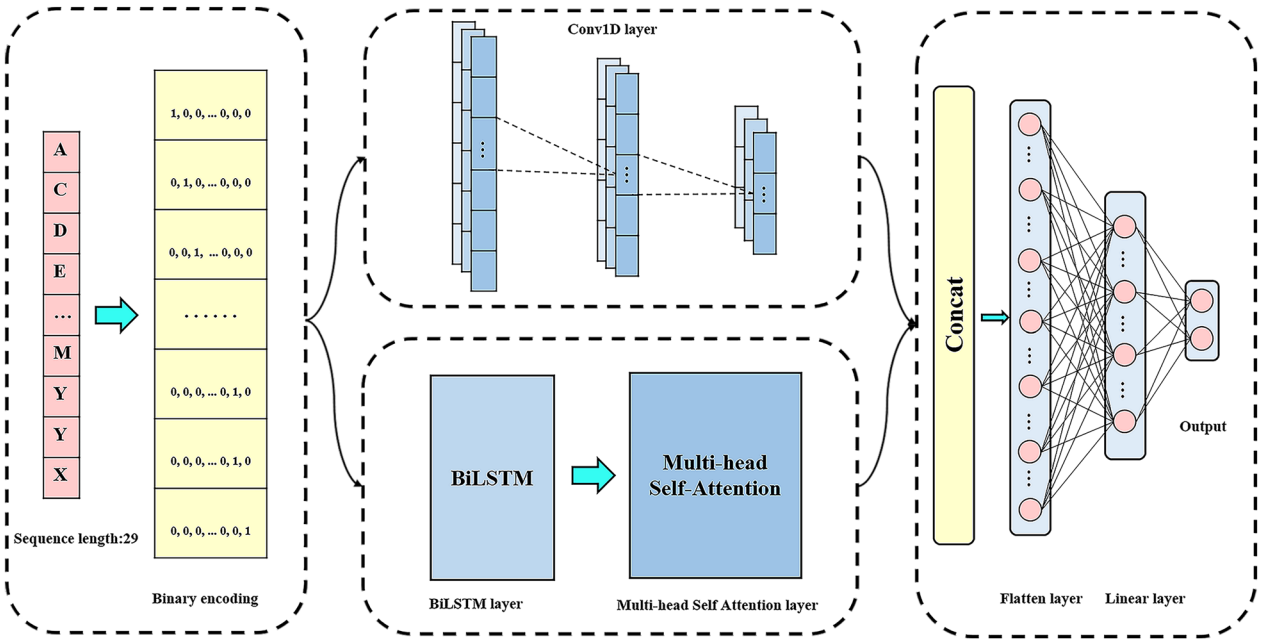

Fig. 2 The architecture of the PlantNh-Kcr model

$$p_t = \begin{cases} p & if\ y = 1 \\ 1 - p & otherwise. \end{cases} \quad 0 \le p \le 1 \tag{1}$$

$$\alpha_t = \begin{cases} \alpha & if\ y = 1 \\ 1 - \alpha & otherwise. \end{cases} \quad 0 \le \alpha \le 1 \tag{2}$$

$$\mathrm{FL}(p_t) = -\alpha_t (1 - p_t)^{\gamma} log(p_t) \quad 0 \le \alpha_t \le 1, \gamma \ge 0 \tag{3}$$

where p represents the probability for the sample to be class 1 (1 represents positive samples, while 0 represents negative samples). α_t represents the balanced weight factor for the sample. The modulating factor $(1 - p_t)^{\gamma}$ is used to adjust the weight of easy samples and hard samples, and γ indicates a tunable focusing parameter.

Optimization of the model

The PlantNh-Kcr model was constructed and trained in a Python 3.9 and Pytorch 1.13.1 environment. Focal loss [50], with α set to 0.7 and γ to 1, was employed as the loss function. The model optimization was achieved using the Adam algorithm [51] with a learning rate of 0.001. The batch size of the input data during training was set to 256, and the number of training epochs was set to 50. To determine the optimal hyperparameters for the model, grid search was employed.

Model evaluation

In bioinformatics, the evaluation of classification models often involves cross-validation and independent test to assess their generalization capabilities. Similar to previous studies [33, 35, 37, 38] that predicted Kcr sites, we employed five-fold cross-validation and independent tests to evaluate PlantNh-Kcr and other models. For this purpose, we prepared the training dataset and the test dataset. For five-fold cross-validation, the training dataset was evenly divided into five folds. Four folds were used to learn the model, while the remaining one was used to validate its performance. This process was repeated five times, ensuring that each fold was used once for validation. For independent test, the training dataset was used to build the model, and the test dataset was used to access its performance.

In bioinformatics, commonly used evaluation metrics for binary classification models include sensitivity (Sn), specificity (Sp), accuracy (ACC), F1-score, Matthews correlation coefficient (MCC), and area under the receiver operating characteristic (ROC) curve (AUC) [35, 38, 52, 53]. The mathematical formulas for Sn, Sp, ACC, and MCC are as follows:

$$\mathrm{Sn} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{4}$$

$$\mathrm{Sp} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{5}$$

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{6}$$

$$\mathrm{F1-score} = \frac{2 \times \mathrm{TP}}{2 \times \mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{7}$$

$$\mathrm{MCC} = \frac{(\mathrm{TP} \times \mathrm{TN}) - (\mathrm{FP} \times \mathrm{FN})}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}} \tag{8}$$

In the above equations, TP, FP, TN, and FN represent the numbers of true positives, false positives, true negatives, and false negatives, respectively. Sn indicates the ability of the model to identify positive samples, with higher values indicating more accurate predictions for positive samples. Sp reflects the ability of the model to identify negative samples, with higher values indicating more accurate predictions for negative samples. F1-score provides a comprehensive measure of the model's performance in identifying positive samples, through balancing the counts of true positives, false positives, and false negatives. A higher F1-score indicates better performance. MCC considers both Sn and Sp, and ranges from − 1 to 1. A higher MCC value indicates better performance of the model. The ROC curve offers a graphical representation of the relationship between the true positive rate (TPR) and false positive rate (FPR) at different thresholds. TPR corresponds to Sn, while FPR equals one minus Sp. AUC represents the probability of a model ranking positive samples above negative samples. AUC is regarded as the most important metric in the evaluation of many bioinformatics models. The closer the ROC curve approaches the upper left corner, the closer the AUC value approaches 1, indicating a better classification performance of the model. In this study, samples with a predicted probability of being positive greater than 0.5 are classified as positive samples. The evaluation metrics of Sn, Sp, ACC, F1-score, and MCC are computed based on the fixed threshold of 0.5. We primarily use the ROC curve and its corresponding AUC value to compare the performance of different models. The ROC curve effectively visualizes the trade-off between Sn and Sp across various thresholds. This

allows to compare models by considering their respective sensitivities at the same specificities, thus providing a more comprehensive evaluation.

To ensure the robustness of our model's performance, we conducted rigorous tests. For five-fold cross-validation, we calculated the mean and standard deviation of the metric values obtained from each fold. For independent test, we conducted 10 independent tests with different random seeds, and calculated the mean and standard deviation of the evaluated metrics.

Results

Conservation analysis of non-histone Kcr sites in plants

Kcr is a post-translational modification that plays a crucial role in various cellular processes. It has been observed that the evolution of Kcr sites exhibits conservation, which suggests that these sites have functional significance [35]. To further investigate the conservation of plant non-histone Kcr sites, we used the pLogo tool [54] to generate a sequence logo (Fig. 3A) using the merged training and test dataset. Significant disparities

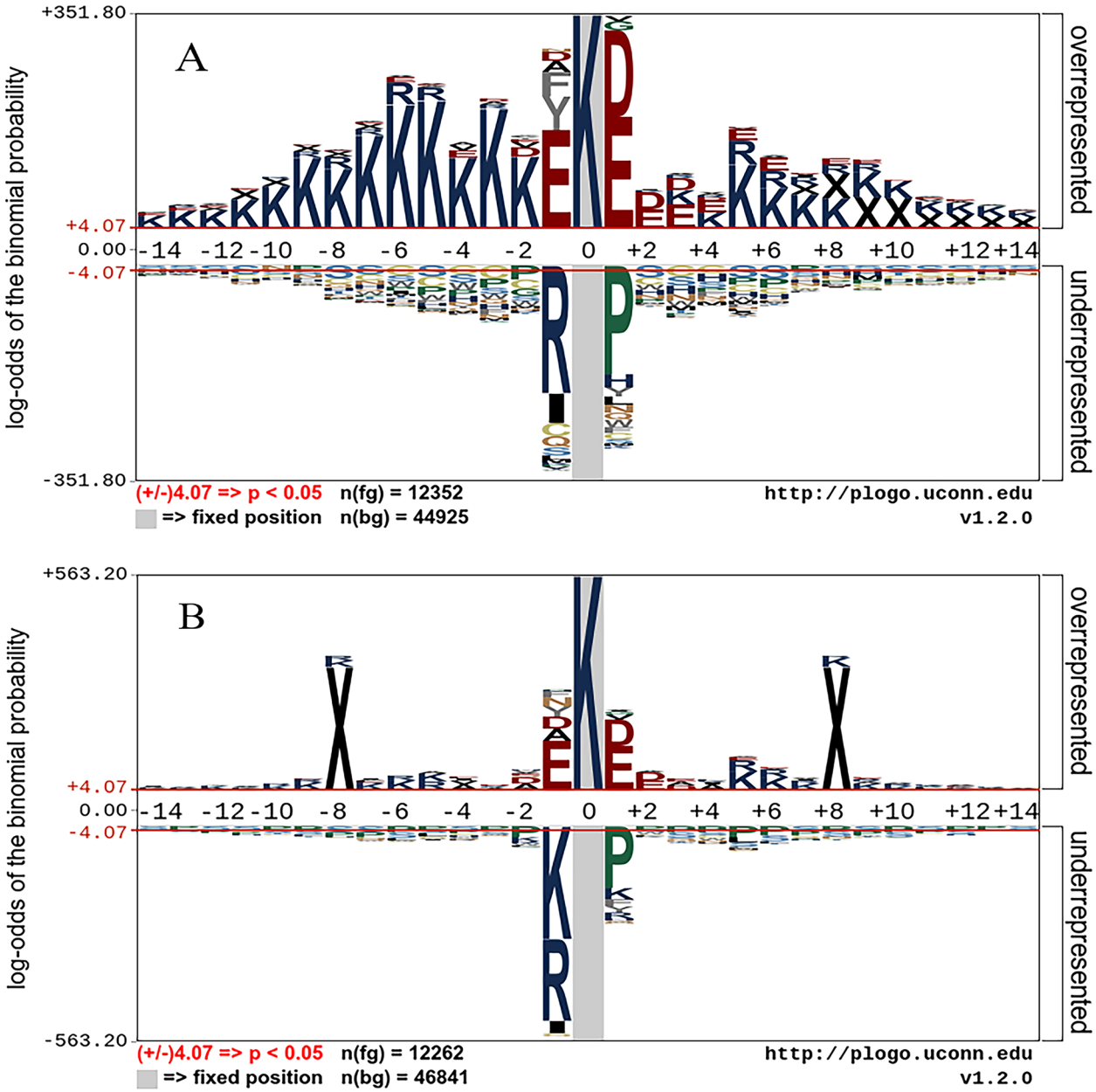

Fig. 3 Sequence logo of Kcr sites on non-histone proteins. **A** Sequence logo for plants; **B** Sequence logo for humans

can be observed in the distribution of some amino acids such as K, D, E, R, and P, surrounding Kcr sites and non-Kcr sites. Notably, residues D and E are overrepresented at positions -1 and + 1. Residue K is prevalent at numerous positions, and it is more overrepresented on the left side of Kcr sites. Residues R and P are significantly underrepresented at positions − 1 and + 1, respectively. Previous research has identified specific motifs that are enriched around Kcr sites, including "EkxxxxxK", "EkxxxK", and "KxxxEK", where x denotes any amino acid [35, 38]. These findings are consistent with our sequence logo, which indicates an overrepresentation of amino acid residues K and E in the vicinity of Kcr sites in plants. To further contextualize our findings, we also generated a sequence logo (Fig. 3B) for Kcr sites on non-histone proteins in humans. The resulting logo was based on the training dataset of nhKcr, a model developed for predicting Kcr sites on human non-histones [35]. Notably, the amino acid distribution observed around human Kcr sites exhibits similarity to that in plants; however, there were slight differences. For example, K is underrepresented at positions − 1 and + 1 for the human Kcr sites, which is not observed in plants. This observation highlights the need for developing a predictive model dedicated to plants.

Performance of PlantNh-Kcr on five-fold cross-validation and independent tests

To evaluate the performance of PlantNh-Kcr, we conducted a five-fold cross-validation and 10 independent tests. For cross-validation, the ROC curves for each fold were tightly clustered in the top left corner of the plot (Fig. 4A), indicating that the model has strong discriminatory power. The average AUC value are 0.891, which is significantly better than random prediction. The average values for Sn, Sp, ACC, F1-score and MCC are 0.821, 0.810, 0.812, 0.648 and 0.551 respectively (Table 2). For independent tests, PlantNh-Kcr also demonstrats strong performance, with the average AUC value of 0.899 (Fig. 4B) and the average values for Sn, Sp, ACC, F1-score, and MCC are 0.811, 0.833, 0.828, 0.665 and 0.572, respectively (Table 2).

To visualize the discriminatory power of PlantNh-Kcr, we utilized the training dataset to train a model and subsequently fed the samples in the test dataset to it. Then we used t-SNE [55] for dimensionality reduction and visualization of the input data in the input layer, the output data of the flatten layer, and the output data of the linear layer. The results are presented in Fig. 5. In this figure, the red and light blue dots represent Kcr and non-Kcr sites, respectively. It is evident from the input layer that Kcr and non-Kcr sites are intermingled. However, following

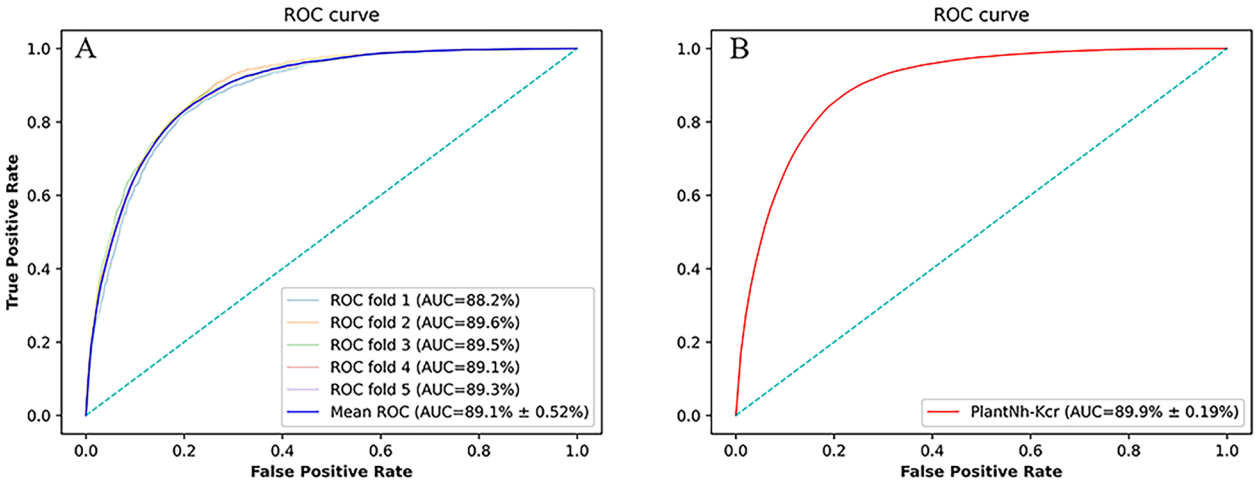

Fig. 4 ROC curves of the PlantNh-Kcr model on five-fold cross-validation and independent tests. **A** The ROC curves on five-fold cross-validation; **B** The ROC curve on independent tests

Table 2 Metric values of the PlantNh-Kcr model on five-fold cross-validation and independent tests

Evaluation methods	Sn (%)	Sp (%)	ACC (%)	F1–score (%)	MCC (%)	AUC (%)
Cross-validation	82.1 ± 2.36	81.0 ± 1.91	81.2 ± 1.04	64.8 ± 0.33	55.1 ± 0.54	89.1 ± 0.54
Independent test	81.1 ± 3.23	83.3 ± 2.09	82.8 ± 0.99	66.5 ± 0.50	57.2 ± 0.50	89.9 ± 0.19

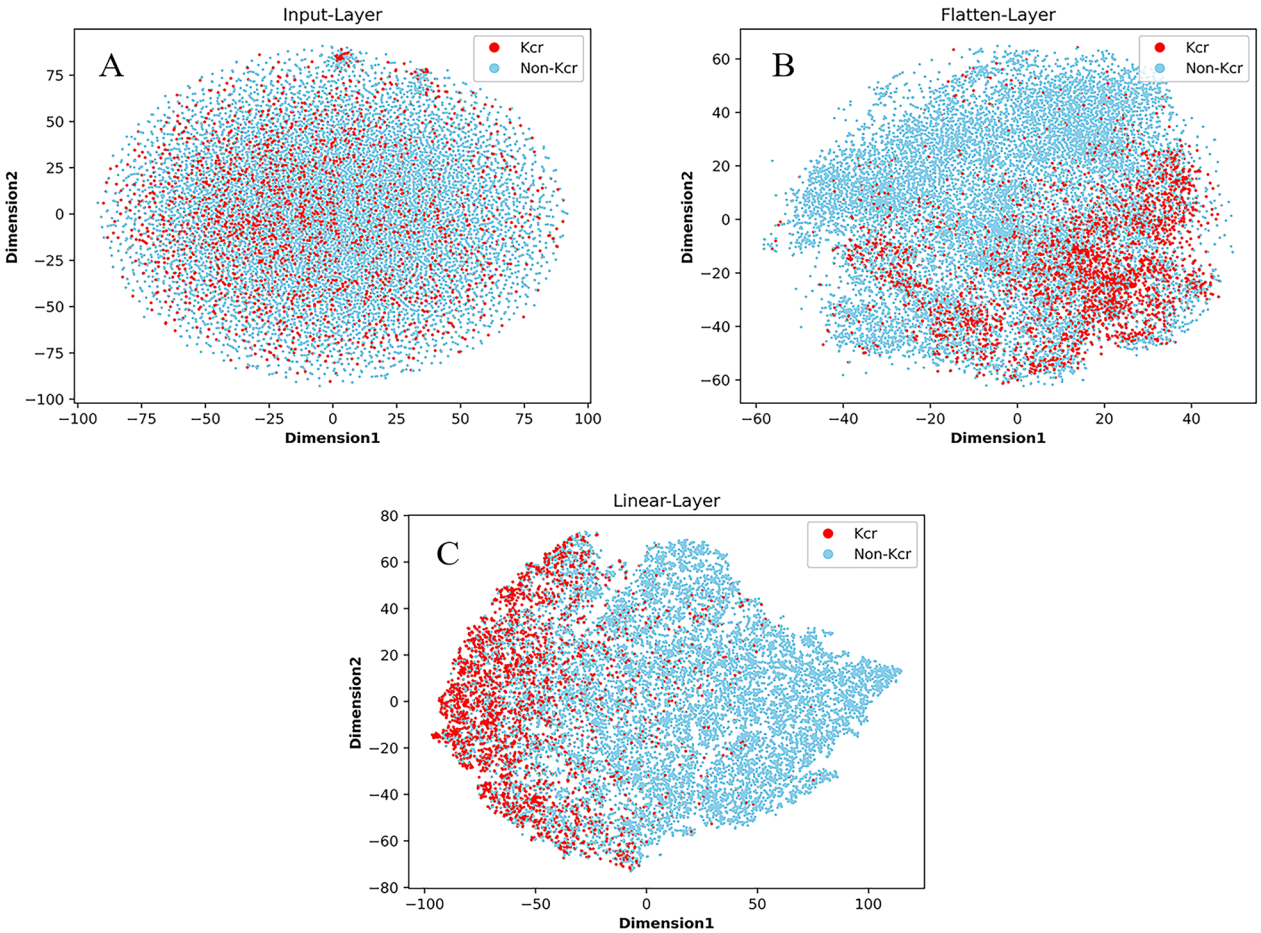

Fig. 5 T-SNE visualization of test samples in PlantNh-Kcr layers. **A** The input layer; **B** The flatten layer; **C** The linear layer

the processing involving three convolutional layers, a BiLSTM layer, and a MHSA layer, most Kcr and non-Kcr sites are separated. This observation underscores the discriminatory capability of these layers in effectively distinguishing between Kcr and non-Kcr sites. Subsequently, after the linear layer, Kcr sites are predominantly clustered in the left region, forming a distinct boundary from non-Kcr sites. This outcome highlights the model's prowess in accurately classifying Kcr sites.

Comparison with conventional machine learning models and deep learning models

To further demonstrate the superior performance and robustness of PlantNh-Kcr, we conducted a comparative analysis with several well-established conventional machine learning models and deep learning models. The detailed information about these models is provided in Additional file 1.

In this study, we utilized three conventional machine learning methods including RF [21], AdaBoost [56], and LightGBM [24]. Additionally, five commonly used encodings were employed, including BE, ACC, EGAAC, AAindex, and BLOUSUM62. Each encoding was fed into each conventional machine learning model for training. The specific results for each combination are summarized in Tables 3, 4. For both five-fold cross-validation and independent tests, RF, AdaBoost, and LightGBM get the largest AUC values, when BE, AAindex, and BLOSUM62 encodings were used as input features, respectively. Among the three models, the LightGBM model emerged as the leader with average AUC values of 0.869 and 0.881 on five-fold cross-validation and independent tests, respectively. However, it lagged behind PlantNh-Kcr in terms of performance. To visually compare the models, Fig. 6 presents the ROC curves on independent tests. Notably, The ROC curve of PlantNh-Kcr is above those of other models. This observation indicates that at the same FPR, PlantNh-Kcr exhibits the highest TPR, indicating its superior ability to correctly identify positive samples compared to other models when predicting the same number of false positives.

Table 3 Metric values of different models on five-fold cross-validation

Classifiers	Encodings	Sn (%)	Sp (%)	ACC (%)	F1_score (%)	MCC (%)	AUC (%)
RF	[a]**BE**	**69.4 ± 1.12**	**71.3 ± 0.94**	**70.9 ± 0.76**	**50.1 ± 0.99**	**34.4 ± 1.29**	**77.5 ± 0.77**
	AAC	70.5 ± 1.89	64.8 ± 0.99	66.0 ± 0.55	44.6 ± 0.78	29.1 ± 1.02	74.5 ± 0.63
	EGAAC	70.9 ± 1.47	59.3 ± 1.44	61.8 ± 0.87	43.8 ± 0.82	24.7 ± 0.67	71.1 ± 0.30
	AAindex	74.1 ± 0.83	60.9 ± 1.41	63.7 ± 0.98	46.2 ± 0.58	28.6 ± 0.78	75.2 ± 0.56
	BLOSUM62	70.1 ± 1.04	66.8 ± 0.53	67.5 ± 0.30	47.6 ± 0.44	30.6 ± 0.60	75.5 ± 0.50
AdaBoost	BE	24.3 ± 1.28	95.4 ± 0.17	80.5 ± 0.51	34.4 ± 1.60	28.6 ± 1.82	79.0 ± 0.60
	AAC	17.6 ± 0.79	95.3 ± 0.28	79.0 ± 0.32	26.0 ± 0.76	20.1 ± 0.52	75.8 ± 0.63
	EGAAC	2.40 ± 0.30	99.4 ± 0.17	79.0 ± 0.28	4.70 ± 0.54	7.60 ± 1.05	71.5 ± 0.66
	AAindex	**25.4 ± 0.86**	**95.3 ± 0.36**	**80.6 ± 0.28**	**35.6 ± 1.09**	**29.5 ± 1.43**	**79.4 ± 0.49**
	BLOSUM62	24.2 ± 0.79	95.6 ± 0.30	80.6 ± 0.26	34.4 ± 0.80	28.8 ± 0.77	78.9 ± 0.55
LightGBM	BE	68.6 ± 1.30	85.0 ± 0.23	81.5 ± 0.47	60.9 ± 1.00	49.5 ± 1.32	85.6 ± 0.52
	AAC	67.3 ± 1.03	73.5 ± 0.74	72.2 ± 0.51	50.4 ± 0.66	34.8 ± 0.77	78.0 ± 0.81
	EGAAC	70.3 ± 0.48	59.4 ± 0.85	61.7 ± 0.67	43.6 ± 0.39	24.3 ± 0.75	70.5 ± 0.62
	AAindex	65.1 ± 1.13	88.0 ± 0.31	83.2 ± 0.32	61.9 ± 0.81	51.3 ± 0.99	86.6 ± 0.23
	BLOSUM62	**66.8 ± 0.64**	**87.2 ± 0.60**	**83.0 ± 0.50**	**62.3 ± 0.93**	**51.7 ± 1.17**	**86.9 ± 0.54**
LSTM	**BE**	**78.3 ± 2.23**	**81.7 ± 2.08**	**81.0 ± 1.20**	**63.4 ± 0.86**	**53.0 ± 0.99**	**88.2 ± 0.12**
	WE	74.8 ± 0.79	81.8 ± 0.47	80.4 ± 0.24	61.6 ± 0.37	50.3 ± 0.36	86.5 ± 0.37
	AAindex	76.6 ± 4.05	83.1 ± 2.18	81.8 ± 0.97	63.8 ± 0.94	53.5 ± 1.15	88.0 ± 0.59
	BLOSUM62	72.3 ± 3.92	84.7 ± 2.36	82.1 ± 1.18	63.0 ± 1.26	52.3 ± 1.50	87.6 ± 0.43
BiLSTM	**BE**	**76.9 ± 3.03**	**82.1 ± 2.71**	**81.0 ± 1.50**	**63.1 ± 1.13**	**52.6 ± 1.22**	**88.0 ± 0.25**
	WE	74.4 ± 2.67	81.9 ± 130	80.3 ± 0.50	61.4 ± 0.55	50.1 ± 0.62	86.6 ± 0.54
	AAindex	77.6 ± 2.28	81.7 ± 1.25	80.9 ± 0.56	63.0 ± 0.55	52.4 ± 0.65	88.1 ± 0.30
	BLOSUM62	81.1 ± 3.10	78.4 ± 2.95	79.0 ± 1.70	62.0 ± 0.86	51.3 ± 1.01	87.7 ± 0.31
CNN	**BE**	**80.9 ± 1.78**	**81.5 ± 0.92**	**81.0 ± 0.50**	**64.2 ± 0.89**	**54.1 ± 1.04**	**88.8 ± 0.34**
	WE	80.3 ± 2.59	81.6 ± 1.95	81.4 ± 1.13	64.4 ± 1.25	54.4 ± 1.43	88.6 ± 0.47
	AAindex	78.3 ± 4.65	82.6 ± 2.94	81.7 ± 1.40	64.3 ± 0.87	54.2 ± 0.90	88.5 ± 0.64
	BLOSUM62	77.3 ± 4.29	83.0 ± 2.75	81.9 ± 1.31	64.2 ± 0.82	54.0 ± 0.96	88.6 ± 0.32
PlantNh-Kcr	**BE**	**82.1 ± 2.36**	**81.0 ± 1.91**	**81.2 ± 1.04**	**64.8 ± 0.33**	**55.1 ± 0.54**	**89.1 ± 0.54**

[a] Bold indicates the best performance for the classifier

Three networks including LSTM network, BiLSTM network, and CNN were used to compare with PlantNh-Kcr. The inputs for these networks encompassed BE, WE, AAindex, and BLOSUM62 encodings. The specific metric values for each network are detailed in Tables 3, 4. Interestingly, all three networks perform best when using BE as input. On five-fold cross-validation, the maximum average AUC values achieved by the LSTM, BiLSTM, and CNN networks are 0.882, 0.880 and 0.888, respectively. Similarly, on independent tests, the maximum average AUC values of these networks are 0.890, 0.889, and 0.896, respectively. However, the performance of the three networks is still inferior to PlantNh-Kcr.

Comparison with existing models for Kcr site prediction on non-histones

To further demonstrate the performance of our model, we conducted a comparative analysis with four other models: nhKcr, iKcr_CNN, CapsNh-Kcr, and DeepKcrot, all designed to predict Kcr sites on non-histones. nhKcr, iKcr_CNN and CapsNh-Kcr predict Kcr sites in human. The nhKcr model integrated BE, AAindex encoding and BLOSUM62 encoding as input features and employed a CNN architecture. The iKcr_CNN model employed a CNN architecture and utilized a focal loss function for optimization. CapsNh-Kcr employed a CNN-based capsule network strategy. DeepKcrot predicted Kcr sites in four species including human, rice, papaya and tabacum. It utilized CNN with WE encoding as input features.

For nhKcr, iKcr_CNN and CapsNh-Kcr, we downloaded their source codes. For DeepKcrot, we rewrote its code due to the unavailability of its web server. We applied focal loss to nhKcr and DeepKcrot because they didn't address the data imbalance issue in their original source codes. We then trained the four models using the training dataset and evaluated their performance on the test set. The prediction performance of the four

Table 4 Metric values of different models on independent test

Classifiers	Encodings	Sn (%)	Sp (%)	ACC (%)	F1_score (%)	MCC (%)	AUC (%)
RF	**BE**	**69.6 ± 0.30**	**70.7 ± 0.41**	**70.5 ± 0.27**	**49.8 ± 0.15**	**33.9 ± 0.22**	**77.4 ± 0.04**
	AAC	70.5 ± 0.32	65.1 ± 0.20	66.2 ± 0.15	46.8 ± 0.18	29.4 ± 0.28	74.6 ± 0.06
	EGAAC	70.5 ± 0.71	60.9 ± 0.54	62.9 ± 0.29	44.4 ± 0.10	25.6 ± 0.17	71.1 ± 0.01
	AAindex	74.5 ± 0.40	60.0 ± 043	63.0 ± 0.27	45.9 ± 0.11	28.2 ± 0.17	75.2 ± 0.10
	BLOSUM62	70.0 ± 0.42	66.4 ± 0.40	67.1 ± 0.27	47.2 ± 0.20	30.1 ± 0.30	75.4 ± 0.11
AdaBoost	BE	23.5 ± 0.00	95.4 ± 0.00	80.3 ± 0.00	33.8 ± 0.00	27.8 ± 0.00	78.9 ± 0.00
	AAC	18.0 ± 0.00	95.6 ± 0.00	79.3 ± 0.00	26.8 ± 0.00	23.1 ± 0.00	76.2 ± 0.00
	EGAAC	2.60 ± 0.00	99.4 ± 0.00	79.0 ± 0.00	5.00 ± 0.00	7.90 ± 0.00	71.5 ± 0.00
	AAindex	**25.8 ± 0.00**	**95.2 ± 0.00**	**80.6 ± 0.00**	**35.0 ± 0.00**	**29.5 ± 0.00**	**79.5 ± 0.00**
	BLOSUM62	23.1 ± 0.00	95.3 ± 0.00	80.1 ± 0.00	32.9 ± 0.00	26.9 ± 0.00	79.0 ± 0.00
LightGBM	BE	71.9 ± 0.00	84.0 ± 0.00	81.4 ± 0.00	62.0 ± 0.00	50.8 ± 0.00	86.6 ± 0.00
	AAC	69.0 ± 0.00	72.2 ± 0.00	71.5 ± 0.00	50.5 ± 0.00	34.9 ± 0.00	78.4 ± 0.00
	EGAAC	72.4 ± 0.00	59.0 ± 0.00	61.9 ± 0.00	44.4 ± 0.00	25.7 ± 0.00	71.1 ± 0.00
	AAindex	69.4 ± 0.00	86.9 ± 0.00	83.2 ± 0.00	63.5 ± 0.00	53.0 ± 0.00	87.6 ± 0.00
	BLOSUM62	**71.7 ± 0.00**	**86.2 ± 0.00**	**83.2 ± 0.00**	**64.2 ± 0.00**	**53.8 ± 0.00**	**88.1 ± 0.00**
LSTM	**BE**	**79.7 ± 5.22**	**82.2 ± 3.29**	**81.7 ± 1.54**	**64.7 ± 0.66**	**54.9 ± 0.60**	**89.0 ± 0.14**
	WE	75.2 ± 1.82	83.4 ± 1.32	81.7 ± 0.62	63.3 ± 0.40	52.7 ± 0.51	87.5 ± 0.21
	AAindex	79.0 ± 4.48	82.5 ± 3.21	81.7 ± 1.61	64.6 ± 0.77	54.6 ± 0.72	88.7 ± 0.32
	BLOSUM62	75.5 ± 4.72	83.8 ± 2.88	82.0 ± 1.33	63.9 ± 0.68	53.6 ± 0.82	88.5 ± 0.43
BiLSTM	**BE**	**75.8 ± 3.49**	**84.3 ± 1.77**	**82.5 ± 0.71**	**64.6 ± 0.50**	**54.5 ± 0.67**	**88.9 ± 0.25**
	WE	77.5 ± 2.79	81.0 ± 0.20	80.3 ± 1.03	62.3 ± 0.60	51.5 ± 0.73	87.4 ± 0.28
	AAindex	79.3 ± 3.45	82.2 ± 2.59	81.6 ± 1.37	64.5 ± 0.88	54.5 ± 0.96	88.9 ± 0.26
	BLOSUM62	75.5 ± 7.16	83.5 ± 4.16	81.8 ± 1.81	63.7 ± 0.58	53.5 ± 0.50	88.7 ± 0.13
CNN	**BE**	**82.1 ± 1.08**	**82.2 ± 0.80**	**82.1 ± 0.43**	**66.0 ± 0.33**	**56.5 ± 0.38**	**89.6 ± 0.07**
	WE	79.0 ± 2.13	83.4 ± 1.33	82.4 ± 0.54	65.4 ± 0.48	55.6 ± 0.63	89.1 ± 0.16
	AAindex	82.1 ± 3.27	81.2 ± 2.06	81.4 ± 0.96	65.0 ± 0.39	55.3 ± 0.38	89.1 ± 0.14
	BLOSUM62	79.2 ± 4.92	83.1 ± 2.88	82.3 ± 1.26	65.3 ± 0.43	55.6 ± 0.38	89.4 ± 0.11
PlantNh-Kcr	**BE**	**81.1 ± 3.23**	**83.3 ± 2.09**	**82.8 ± 0.99**	**66.5 ± 0.50**	**57.2 ± 0.50**	**89.9 ± 0.19**

[a] Bold indicates the best performance for the classifier

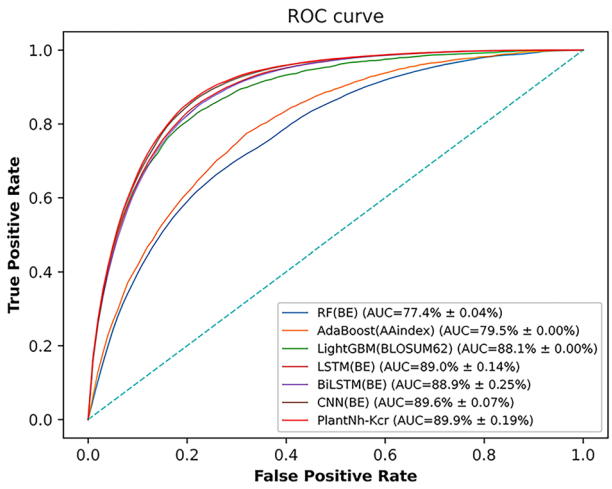

Fig. 6 ROC curves of different models on independent tests

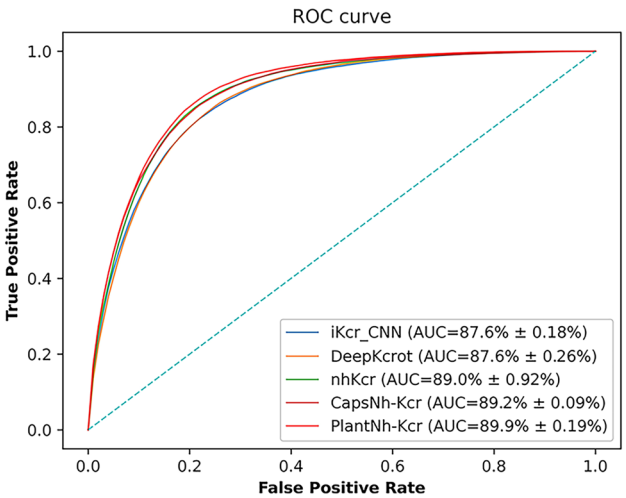

Fig. 7 ROC curves of PlantNh-Kcr and the other four models

models is shown in Fig. 7 and Table 5. The average AUC values are 0.876, 0.876, 0.890, and 0.892, respectively, which are lower than that of PlantNh-Kcr. This again underscores the superior performance of PlantNh-Kcr.

Ablation study

To assess the effect of each component in the PlantNh-Kcr model on prediction performance, we conducted an ablation study. In this study, we removed the linear layer, CNN, MHSA, and BiLSTM + MHSA individually from the model and evaluated the prediction performance on independent tests. The results are summarized in Table 6. Removing the linear layer and CNN individually resulted in a decrease of 1.1% and 1.2% in AUC values, respectively. This suggests that these two components have a certain impact on the overall performance of the model. On the other hand, removing MHSA and BiLSTM + MHSA individually resulted in a decrease of 0.5% and 0.3% in AUC values, respectively, indicating that these components have a smaller impact on performance compared to the linear layer and CNN. Overall, our results demonstrate that each component in the PlantNh-Kcr model contributes to its prediction performance. Removing any module from the model will result in a decrease in performance, indicating that each module is essential for achieving optimal performance.

The performance of PlantNh-Kcr on independent tests for each plant

In this study, we collected non-histone Kcr sites from different types of plants. Given the potential species-specific impact on these sites, it's necessary to assess the generalizability of our predictive model across diverse plant species. Therefore, we studied the performance of our model for each species on independent tests. Table 7 details the evaluation metrics for each species, which are further visualized in Fig. 8 as a bar chart.

The results indicate that the prediction performance of the model varies slightly across different species. Notably, peanuts and papaya exhibit particularly strong performance, with average AUC values of 0.930 and 0.914, respectively. The model also demonstrates good performance for tabacum and rice, with average AUC values of

Table 5 Metric values of PlantNh-Kcr and the other four models

Models	Sn (%)	Sp (%)	ACC (%)	F1_score (%)	MCC (%)	AUC (%)
iKcr_CNN	77.2 ± 1.07	82.0 ± 0.79	81.0 ± 0.45	63.1 ± 0.39	52.5 ± 0.51	87.6 ± 0.18
DeepKcrot	82.8 ± 1.53	77.9 ± 1.46	78.9 ± 0.85	62.3 ± 0.56	51.9 ± 0.59	87.6 ± 0.26
nhKcr	87.6 ± 2.42	77.3 ± 3.26	79.4 ± 3.26	64.3 ± 1.61	55.1 ± 1.63	89.0 ± 0.92
CapsNh-Kcr	76.4 ± 3.59	84.3 ± 1.59	83.1 ± 0.71	65.5 ± 0.47	55.6 ± 0.58	89.2 ± 0.09
PlantNh-Kcr	81.1 ± 3.23	83.3 ± 2.09	82.8 ± 0.99	66.5 ± 0.50	57.2 ± 0.50	89.9 ± 0.19

Table 6 Prediction performance of models in the ablation study

Model	Sn (%)	Sp (%)	ACC (%)	F1_score (%)	MCC (%)	AUC (%)
Removing linear layer	90.0 ± 1.19	70.7 ± 2.24	74.8 ± 1.56	60.1 ± 1.27	50.2 ± 1.46	88.8 ± 0.32
Removing CNN	78.0 ± 4.48	82.6 ± 3.00	81.6 ± 1.50	64.1 ± 0.92	53.9 ± 1.09	88.7 ± 0.44
Removing MHSA	79.2 ± 2.33	84.5 ± 1.45	83.4 ± 0.70	66.7 ± 0.51	57.3 ± 0.64	89.4 ± 0.44
Removing BiLSTM + MHSA	82.1 ± 1.08	82.2 ± 0.80	82.1 ± 0.43	66.0 ± 0.33	56.5 ± 0.38	89.6 ± 0.07
PlantNh-Kcr	81.1 ± 3.23	83.3 ± 2.09	82.8 ± 0.99	66.5 ± 0.50	57.2 ± 0.50	89.9 ± 0.19

Table 7 Performance of PlantNh-Kcr for different plants

Species	Sn (%)	Sp (%)	ACC (%)	F1-score (%)	MCC (%)	AUC (%)
Wheat	73.4 ± 3.94	81.9 ± 2.19	79.8 ± 0.76	64.2 ± 0.80	51.3 ± 0.93	85.8 ± 0.32
Tabacum	79.0 ± 4.16	84.5 ± 1.98	83.1 ± 0.81	69.7 ± 1.22	59.1 ± 1.67	89.6 ± 0.46
Rice	87.2 ± 2.18	78.2 ± 2.32	79.6 ± 1.68	57.8 ± 1.65	51.3 ± 1.69	89.0 ± 0.49
Peanut	87.9 ± 2.54	84.0 ± 2.17	84.7 ± 1.31	68.3 ± 1.29	61.5 ± 1.21	93.0 ± 0.21
Papaya	81.1 ± 3.58	85.5 ± 1.94	84.6 ± 0.81	69.2 ± 0.57	60.4 ± 0.75	91.4 ± 0.30

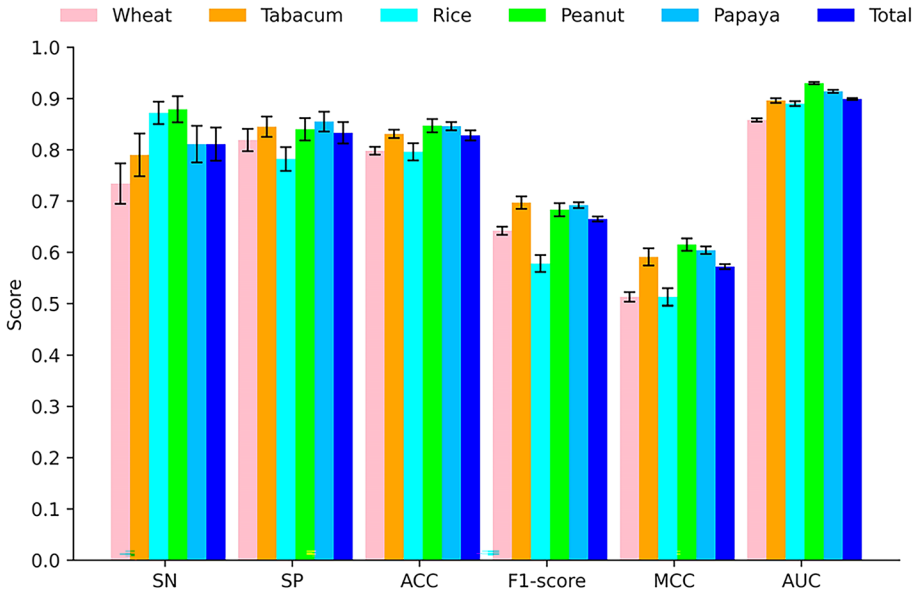

Fig. 8 Metric values on independent tests for different plants

0.896 and 0.890, respectively. However, wheat exhibits slightly lower performance compared to other species, with an average AUC value of 0.858. This may be attributed to species-specific characteristics.

To study the performance of species-specific models, we developed individual models for each plant using samples from the corresponding species in the training dataset. We then evaluated these models using samples from the corresponding species in the test set. The performance of these models on five metrics are shown in Table 8. Notably, the peanut-specific and papaya-specific models exhibit the best performance, with average AUC values of 0.920 and 0.902, respectively. In contrast, the species-specific models for rice, tabacum, and wheat exhibit relatively poorer performance. This can be attributed to the smaller training set size for rice and tabacum and potential species-specific characteristics affecting crotonylation patterns in wheat. When compared with the general model's performance in Table 7, the species-specific models underperform. This finding underscores the advantage of integrating data from diverse species to train a general predictive model for plant non-histone Kcr sites.

Discussion

The PlantNh-Kcr exhibits superior performance. However, there are still some issues that need to be considered.

First, our model PlantNh-Kcr contains three convolutional layers, which can effectively capture local patterns in protein sequences. Careful consideration must be given to the kernel size and the step size, as well as the number of convolution kernels. Too few or too many convolution kernels can lead to information loss or overfitting, respectively, which can impact model performance. Furthermore, when utilizing the convolutional layer to process long protein sequences, there is a risk of losing global contextual information. This can be a limiting factor in the predictive accuracy of the model. To address this issue, stacking multiple convolutional layers and effectively integrating their outputs can compensate for the loss of global context. By doing so, the model can

Table 8 Performance of species-specific models on independent tests

Species	Sn (%)	Sp (%)	ACC (%)	F1-score (%)	MCC (%)	AUC (%)
Wheat	73.3 ± 5.12	77.6 ± 4.26	76.5 ± 1.95	60.7 ± 0.53	46.2 ± 0.86	83.1 ± 0.36
Tabacum	72.4 ± 2.28	77.1 ± 1.06	76.0 ± 0.43	59.7 ± 0.77	44.7 ± 1.06	82.7 ± 0.77
Rice	66.2 ± 5.53	83.2 ± 3.46	80.4 ± 2.07	52.8 ± 1.11	43.2 ± 1.31	83.6 ± 0.80
Peanut	83.7 ± 2.55	84.8 ± 1.58	84.6 ± 0.84	67.1 ± 0.65	59.6 ± 0.62	92.0 ± 0.17
Papaya	84.4 ± 2.90	80.5 ± 2.26	81.4 ± 1.20	66.0 ± 0.85	56.6 ± 0.93	90.2 ± 0.19

achieve a more comprehensive understanding of the protein sequences, ultimately leading to improved predictive performance.

Second, multiple encodings were described in the paper, such as BE, WE encoding, AAindex encoding, and BLOSUM62 encoding. The PlantNh-Kcr model only utilize BE as input features. We have attempted to integrate multiple encodings as input features of the model, but failed to improve the performance. This may be because these features have poor complementarity.

Third, there are far more negative samples than positive samples in our training set. This imbalance can significantly influence model training, biasing it towards the negative samples. To address this issue, three methods were employed: up-sampling the positive samples, down-sampling the negative samples, and utilizing the focal loss function. Among these methods, the focal loss function presented the best prediction performance, and improved the ability of the model to correctly predict positive samples. We believe that dataset imbalance remains a potential problem that needs to be addressed in bioinformatics.

Conclusion

In this study, we compiled a large dataset of non-histone Kcr sites from five different plant species. Using this dataset, we developed a deep learning model called PlantNh-Kcr to predict non-histone Kcr sites in plants. The model's architecture integrates CNN, LSTM, and attention mechanism, utilizing BE as its primary input features. Notably, the model exhibits satisfactory performance on both five-fold cross-validation and independent tests, outperforming several other models. In addition, there are minor variations in prediction performance across different plant species, a general predictive model demonstrates superior performance compared to species-specific models. We believe that the PlantNh-Kcr model offers a valuable contribution to addressing challenges and advancing the study of plant Kcr sites. We also believe that as more Kcr sites are experimentally determined and as deep learning techniques continue to develop, we will see the emergence of more high-performance models for predicting Kcr sites.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13007-024-01157-8.

Additional file 1. Detailed information about the conventional machine learning models.

Acknowledgements
We are very grateful to the experimental scientists for their publicly available data of Kcr sites.

Author contributions
J.Y. devised the method, drafted and revised the manuscript. Y.R. analyzed the data and revised the manuscript. W.X. supervised the study and revised the manuscript.

Funding
This work was supported by the Start-up fund of Shanxi Normal University (83358).

Availability of data and materials
The source code and datasets are publicly available at https://github.com/jiangyanming-individual/PlantNh-Kcr.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
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.

Received: 6 December 2023 Accepted: 7 February 2024
Published online: 15 February 2024

References

1. Bao W, Yang B, Chen B. 2-hydr_ensemble: lysine 2-hydroxyisobutyrylation identification with ensemble method. Chemometr Intell Laboratory Syst. 2021. https://doi.org/10.1016/j.chemolab.2021.104351.
2. Tan M, Luo H, Lee S, et al. Identification of 67 Histone marks and histone lysine crotonylation as a new type of histone modification. Cell. 2011;146:1016–28. https://doi.org/10.1016/j.cell.2011.08.008.
3. Ruiz-Andres O, Sanchez-Niño MD, Cannata-Ortiz P, et al. Histone lysine-crotonylation in acute kidney injury. Dis Model Mech. 2016. https://doi.org/10.1242/dmm.024455.
4. Abu-Zhayia ER, Machour FE, Ayoub N. HDAC-dependent decrease in histone crotonylation during DNA damage. J Mol Cell Biol. 2019;11:804–6. https://doi.org/10.1093/jmcb/mjz019.
5. Montellier E, Rousseaux S, Zhao Y, et al. Histone crotonylation specifically marks the haploid male germ cell gene expression program: post-meiotic male-specific gene expression. BioEssays. 2012;34:187–93. https://doi.org/10.1002/bies.201100141.
6. Wu Q, Li W, Wang C, et al. Ultradeep lysine crotonylome reveals the crotonylation enhancement on both histones and nonhistone proteins by SAHA treatment. J Proteome Res. 2017;16:3664–71. https://doi.org/10.1021/acs.jproteome.7b00380.
7. Wei W, Mao A, Tang B, et al. Large-scale identification of protein crotonylation reveals its role in multiple cellular functions. J Proteome Res. 2017;16:1743–52. https://doi.org/10.1021/acs.jproteome.7b00012.
8. Xu W, Wan J, Zhan J, et al. Global profiling of crotonylation on non-histone proteins. Cell Res. 2017;27:946–9. https://doi.org/10.1038/cr.2017.60.
9. Hou JY, Zhou L, Li JL, et al. Emerging roles of non-histone protein crotonylation in biomedicine. Cell Biosci. 2021;11:101. https://doi.org/10.1186/s13578-021-00616-2.
10. Sun H, Liu X, Li F, et al. First comprehensive proteome analysis of lysine crotonylation in seedling leaves of Nicotiana tabacum. Sci Rep. 2017;7:3013. https://doi.org/10.1038/s41598-017-03369-6.

11. Liu K, Yuan C, Li H, et al. A qualitative proteome-wide lysine crotonylation profiling of papaya (Carica papaya L.). Sci Rep. 2018;8:8230. https://doi.org/10.1038/s41598-018-26676-y.
12. Liu S, Xue C, Fang Y, et al. Global involvement of lysine crotonylation in protein modification and transcription regulation in rice. Mol Cell Proteomics. 2018;17:1922–36. https://doi.org/10.1074/mcp.RA118.000640.
13. Xu M, Luo J, Li Y, et al. First comprehensive proteomics analysis of lysine crotonylation in leaves of peanut (Arachis hypogaea L.). Proteomics. 2021;21:e2000156. https://doi.org/10.1002/pmic.202000156.
14. Zhang N, Wang S, Zhao S, et al. Global crotonylatome and GWAS revealed a TaSRT1-TaPGK model regulating wheat cold tolerance through mediating pyruvate. Sci Adv. 2023;9:eadg1012. https://doi.org/10.1126/sciadv.adg1012.
15. Zhu D, Liu J, Duan W, et al. Analysis of the chloroplast crotonylome of wheat seedling leaves reveals the roles of crotonylated proteins involved in salt-stress responses. J Exp Bot. 2023;74:2067–82. https://doi.org/10.1093/jxb/erad006.
16. Lu Y, Xu Q, Liu Y, et al. Dynamics and functional interplay of histone lysine butyrylation, crotonylation, and acetylation in rice under starvation and submergence. Genome Biol. 2018;19:144. https://doi.org/10.1186/s13059-018-1533-y.
17. Lin P, Bai HR, He L, et al. Proteome-wide and lysine crotonylation profiling reveals the importance of crotonylation in chrysanthemum (Dendranthema grandiforum) under low-temperature. BMC Genomics. 2021;22:51. https://doi.org/10.1186/s12864-020-07365-5.
18. Yu H, Bu C, Liu Y, et al. Global crotonylome reveals CDYL-regulated RPA1 crotonylation in homologous recombination-mediated DNA repair. Sci Adv. 2020;6:eaay4697. https://doi.org/10.1126/sciadv.aay4697.
19. Yang YH, Wu SF, Kong J, et al. Using ATCLSTM-Kcr to predict and generate the human lysine crotonylation database. J Proteomics. 2023;281: 104905. https://doi.org/10.1016/j.jprot.2023.104905.
20. Joachims T. Making large-scale SVM learning practical. Technical report, 1998.
21. Breiman LJMI. Random forests. Mach Learn. 2001;45:5–32.
22. Bao W, Cui Q, Chen B, et al. Phage_UniR_LGBM: phage virion proteins classification with UniRep features and lightGBM model. Comput Math Methods Med. 2022;2022:9470683. https://doi.org/10.1155/2022/9470683.
23. Bao W, Gu Y, Chen B, et al. Golgi_DF: golgi proteins classification with deep forest. Front Neurosci. 2023;17:1197824. https://doi.org/10.3389/fnins.2023.1197824.
24. Ke G, Meng Q, Finley T, et al. Lightgbm: a highly efficient gradient boosting decision tree. Front Neurosci. 2017. https://doi.org/10.3389/fnins.2023.1197824.
25. Huang G, Zeng W. A discrete hidden Markov model for detecting histone crotonyllysine sites. J Mol Graph Modell. 2016;75:717–30.
26. Ju Z, He JJ. Prediction of lysine crotonylation sites by incorporating the composition of k-spaced amino acid pairs into Chou's general PseAAC. J Mol Graph Model. 2017;77:200–4. https://doi.org/10.1016/j.jmgm.2017.08.020.
27. Qiu WR, Sun BQ, Xiao X, et al. iKcr-PseEns: Identify lysine crotonylation sites in histone proteins with pseudo components and ensemble classifier. Genomics. 2018;110:239–46. https://doi.org/10.1016/j.ygeno.2017.10.008.
28. Malebary SJ, Rehman MSU, Khan YD. iCrotoK-PseAAC: Identify lysine crotonylation sites by blending position relative statistical features according to the Chou's 5-step rule. PLoS ONE. 2019;14: e0223993. https://doi.org/10.1371/journal.pone.0223993.
29. Liu Y, Yu Z, Chen C, et al. Prediction of protein crotonylation sites through LightGBM classifier based on SMOTE and elastic net. Anal Biochem. 2020;609: 113903. https://doi.org/10.1016/j.ab.2020.113903.
30. Meng R, Yin S, Sun J, et al. scAAGA: Single cell data analysis framework using asymmetric autoencoder with gene attention. Comput Biol Med. 2023;165: 107414. https://doi.org/10.1016/j.compbiomed.2023.107414.
31. Lv H, Dao F-Y, Guan Z-X, et al. Deep-Kcr: accurate detection of lysine crotonylation sites using deep learning method. Briefings Bioinf. 2021. https://doi.org/10.1093/bib/bbaa255.
32. Qiao Y, Zhu X, Gong H, et al. BERT-Kcr: prediction of lysine crotonylation sites by a transfer learning method with pre-trained BERT models. Bioinformatics. 2022;38:648–54. https://doi.org/10.1093/bioinformatics/btab712.
33. Khanal J, Tayara H, Zou Q, et al. DeepCap-Kcr: accurate identification and investigation of protein lysine crotonylation sites based on capsule network. Brief Bioinform. 2022. https://doi.org/10.1093/bib/bbab492.
34. Li Z, Fang J, Wang S, et al. Adapt-Kcr: a novel deep learning framework for accurate prediction of lysine crotonylation sites based on learning embedding features and attention architecture. Brief Bioinform. 2022. https://doi.org/10.1093/bib/bbac037.
35. Chen YZ, Wang ZZ, Wang Y, et al. nhKcr: a new bioinformatics tool for predicting crotonylation sites on human nonhistone proteins based on deep learning. Brief Bioinform. 2021. https://doi.org/10.1093/bib/bbab146.
36. Wei X, Sha Y, Zhao Y, et al. DeepKcrot: a deep-learning architecture for general and species-specific lysine crotonylation site prediction. IEEE Access. 2021;9:49504–13. https://doi.org/10.1109/access.2021.3068413.
37. Dou L, Zhang Z, Xu L, et al. iKcr_CNN: A novel computational tool for imbalance classification of human nonhistone crotonylation sites based on convolutional neural networks with focal loss. Comput Struct Biotechnol J. 2022;20:3268–79. https://doi.org/10.1016/j.csbj.2022.06.032.
38. Khanal J, Kandel J, Tayara H, et al. CapsNh-Kcr: Capsule network-based prediction of lysine crotonylation sites in human non-histone proteins. Comput Struct Biotechnol J. 2023;21:120–7. https://doi.org/10.1016/j.csbj.2022.11.056.
39. Kim YJapa. Convolutional neural networks for sentence classification 2014.
40. Schuster M, Paliwal KK. Bidirectional recurrent neural networks. IEEE Trans Signal Process. 1997;45:2673–81. https://doi.org/10.1109/78.650093.
41. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. Nucleic Acids Res. 2017. https://doi.org/10.1093/nar/gkr1048.
42. Dimmer EC, Huntley RP, Alam-Faruque Y, et al. The UniProt-GO annotation database in 2011. Nucleic Acids Res. 2012;40:D565-570. https://doi.org/10.1093/nar/gkr1048.
43. Benson DA, Karsch-Mizrachi I, Lipman DJ, et al. GenBank. Nucleic Acids Res. 2009;37:D26-31. https://doi.org/10.1093/nar/gkn723.
44. Huang Y, Niu B, Gao Y, et al. CD-HIT Suite: a web server for clustering and comparing biological sequences. Bioinformatics. 2010;26:680–2. https://doi.org/10.1093/bioinformatics/btq003.
45. Yang KK, Wu Z, Bedbrook CN, et al. Learned protein embeddings for machine learning. Bioinformatics. 2018;34:2642–8. https://doi.org/10.1093/bioinformatics/bty178.
46. Lyu X, Li S, Jiang C, et al. DeepCSO: a deep-learning network approach to predicting cysteine S-Sulphenylation sites. Front Cell Dev Biol. 2020;8:594587. https://doi.org/10.3389/fcell.2020.594587.
47. Kawashima S, Ogata H, Kanehisa M. AAindex: amino acid index database. Nucleic Acids Res. 1999;27:368–9. https://doi.org/10.1093/nar/27.1.368.
48. Henikoff S, Henikoff JG. Amino acid substitution matrices from protein blocks. Proc Natl Acad Sci U S A. 1992;89:10915–9. https://doi.org/10.1073/pnas.89.22.10915.
49. Nair V, Hinton GE. Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th international conference on machine learning (ICML-10). 2010, p. 807–814.
50. Lin T-Y, Goyal P, Girshick R et al. Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 2980–2988.
51. Kingma DP, Ba JJapa. Adam: A method for stochastic optimization 2014.
52. Khanal J, Lim DY, Tayara H, et al. i6mA-stack: A stacking ensemble-based computational prediction of DNA N6-methyladenine (6mA) sites in the Rosaceae genome. Genomics. 2021;113:582–92. https://doi.org/10.1016/j.ygeno.2020.09.054.
53. Jia C, He W. EnhancerPred: a predictor for discovering enhancers based on the combination and selection of multiple features. Sci Rep. 2016;6:38741. https://doi.org/10.1038/srep38741.
54. O'Shea JP, Chou MF, Quader SA, et al. pLogo: a probabilistic approach to visualizing sequence motifs. Nat Methods. 2013;10:1211–2. https://doi.org/10.1038/nmeth.2646.
55. Van der Maaten L, Hinton GJJomlr. Visualizing data using t-SNE 2008;9.
56. Freund Y, Schapire REJJoc, sciences s. A decision-theoretic generalization of on-line learning and an application to boosting 1997;55:119-139.

Publisher's Note