Predicting protein amidation sites by orchestrating amino acid sequence features

Shuqiu Zhao¹,², Hua Yu¹,² and Xiujun Gong¹,²,*

¹School of Computer Science and Technology, Tianjin University, Nankai, Tianjin, China, 30072
²Tianjin Key Laboratory of Cognitive Computing and Application, Nankai, Tianjin, China, 30072
* Corresponding author: gongxj@tju.edu.cn

Abstract. Amidation is the fourth major category of post-translational modifications, which plays an important role in physiological and pathological processes. Identifying amidation sites can help us understanding the amidation and recognizing the original reason of many kinds of diseases. But the traditional experimental methods for predicting amidation sites are often time-consuming and expensive. In this study, we propose a computational method for predicting amidation sites by orchestrating amino acid sequence features. Three kinds of feature extraction methods are used to build a feature vector enabling to capture not only the physicochemical properties but also position related information of the amino acids. An extremely randomized trees algorithm is applied to choose the optimal features to remove redundancy and dependence among components of the feature vector by a supervised fashion. Finally the support vector machine classifier is used to label the amidation sites. When tested on an independent data set, it shows that the proposed method performs better than all the previous ones with the prediction accuracy of 0.962 at the Matthew’s correlation coefficient of 0.89 and area under curve of 0.964.

1. Introduction
Post translational modification (PTM) can affect the function and structure of a protein and play an important role in many physiological and pathological processes, such as immune response [1], cancer diseases [2, 3], neural diseases [4] and so on. Amidation is a type of PTMs, which can happen at the C-terminus of a peptide as known as C-terminal amidation. Amidation is crucial to the activation of many neuropeptides and hormones, such as receptor recognition and signal transduction [5, 6], and it also associates with a variety of diseases such as hypertension [7], cancer [8], neural dysfunction [9], and sleep apnea [10]. Therefore it is very necessary to identify amidation sites. Because traditional experimental methods are often time consuming and costly, identifying amidation using computational methods is a practical alternative.

In the past several years, a number of methods have emerged to predict PTMs sites, and some of them have achieved very good performance. For instance, the PseAAC algorithm proposed by Chou et al. [11] is widely used to represent sequence order information, such as in cysteine S-nitrosylation sites predictor [12]. Lv et al. [13] proposed a predictor named CarSPred using the high quality indices 8 (HQI8) feature to predict carbonylation sites of human proteins. The composition of k-spaced amino acid pairs (CKSAAP) method was used to predict Ubiquitination sites by Chen et al. [14]. Xu et al. [15] developed SUMO_LDA based on position-specific amino acid propensity (PSAAP) and modification of composition of k-space amino acid pair (MCKSAAP) feature extraction methods to predict sumoylation sites. Those methods are also applicable to the amidation sites prediction. Cui et
al. [16] proposed an amidation sites predictor based on nearest neighbor algorithm. Wang et al. [17] developed PrAS, which combined four representative features to identify the amidation sites. Both of them achieved great performance.

In this paper, we aim at building a more accurate predictor by orchestrating amino acid sequence features making full use of position related information and physicochemical information. The position related features including PSAAP and PKSAAP can more correctly describe the distribution difference of the amino acids at each position between amidation samples and non-amidation samples, and HQI8 feature can capture the physicochemical properties of the amino acids. An extremely randomized trees algorithm is applied to choose the optimal features to remove redundancy and dependence among components of the feature vector by a supervised fashion. Finally the SVM classifier is used to label the amidation sites.

2. Materials and Methods

2.1. Dataset

Building an experimentally validated dataset is a critical step for the success construction of a classifier in identifying amidation sites computationally. In this paper, the dataset used to train the classifier is derived from Wang's dataset [17], which can be obtained from the UniProt database. There are 209 protein sequences in the dataset. 139 sequences are served as training set, and 70 are used as the independent test set. There are 332 amidation sites and 1202 non-amidation sites in the training set, 165 amidation sites and 632 non-amidation sites in the independent test set. A protein sequence fragment \( P \) that contains 21 amino acid residues can be expressed as:

\[
P = R_1 R_2 \ldots R_{11} \ldots R_{20} R_{21}
\]

where \( R_i \) denotes one of the 20 natural amino acids or the pseudo amino acid 'X'. \( R_{11} \) is the potential amidation site, and \( R_{12} \) is a Glycine residue.

2.2. Feature Extraction

In order to predict amidation sites using a computational method, we need to translate sequence fragments into feature vectors which are recognizable and learnable by a classifier, like Support Vector Machine (SVM). Three feature extraction methods are used to build the feature vector including HQI8, PSAAP and PKSAAP features for representing the sequence fragments.

2.2.1. HQI8 features. The AAindex database [18] contains a variety of physicochemical properties of amino acids. Because very large size of features in a classification problem may cause over-fitting [19], thus, Sahara et al. [20] use the fuzzy clustering method to cluster these properties, and each of the achieved clusters is represented by their central properties, namely, high-quality indices (HQI8). These amino acid indices' accession numbers in AAindex database are BLAM930101, BIOV880101, MAXF760101, TSAJ990101, NAKH920108, CEDJ970104, LIFS790101 and MIYS990104.

We use HQI8 to do feature extraction on the sequence fragments, and each sequence fragment with the length of \( L \), will produce an \( L \times 8 \)-dimensional feature vector. Without considering the amino acid \( R_{12} \), each sequence fragment in our work will have a 160-dimensional feature vector, and when we encounter pseudo-amino acid 'X', all the property values are set to 0.

2.2.2. PSAAP features. According to the annotated situations of the center amino acid \( R_{11} \), the training set can be divided into the amidated positive dataset and the non-amidated negative dataset. The basic idea of PSAAP is to obtain the frequency difference of 21 amino acids at each position between the positive training set and negative training set, and this method has been successfully applied to the prediction of many types of PTMs sites [21].

When the length of the fragment is \( L \), the PSAAP generates a \( 21 \times L \)-dimensional position-specific amino acid propensity matrix \( Z = Pos - Neg \), where \( Pos \) and \( Neg \) are defined as:
The 21×L-dimensional matrix $\text{Pos}$ is the occurrence frequency of 21 amino acids at each position of protein sequences contained in positive training dataset, and matrix $\text{Neg}$ has the same size as $\text{Pos}$, but with the numbers that counted from fragments of negative training dataset.

By directly querying the matrix $Z$, we can obtain the feature vector $T$ with the length of $L$, defined as equation (3). So the dimensions of PSAAP feature vector in our work will be 21.

$$T = (t_1, t_2, \cdots, t_i, \cdots, t_{L-1}, t_L)$$

where $t_i$ is defined as:

$$t_i = \begin{cases} 
\varepsilon_{1,i} & \text{when } R_i = A \\
\varepsilon_{2,i} & \text{when } R_i = C \\
\vdots & \\
\varepsilon_{21,i} & \text{when } R_i = X
\end{cases}$$

The pertinence value of events $x$ and $y$ is calculated as:

$$C_{x,y} = \frac{p(x,y)}{p(x)p(y)}$$

where $p(x)$ is the probability of occurrence of event $x$, $p(y)$ corresponding probability of event $y$, and $p(x,y)$ is the probability that event $x$ and event $y$ occur simultaneously.

We use $I_{l,k}$ to represent the pertinence difference of amino acid pair formed by $l$-th and $(l+k+1)$-th amino acids between the positive and negative training set, calculated as follows:

$$I_{l,k} = \frac{p_{b,(l,l+k+1)}^+ f_{i,l}^+ f_{j,l+k+1}^+ - n_{b,(l,l+k+1)}^- f_{i,l}^- f_{j,l+k+1}^-}{f_{i,l}^+ f_{j,l+k+1}^+}$$

where $f_{i,l}^+$ and $f_{i,l}^-$ are defined in equation (2), and represent the frequency of corresponding amino acids. The $p_{b,(l,l+k+1)}^+$ and $n_{b,(l,l+k+1)}^-$ are defined in equation (7), and $p_{b,(l,l+k+1)}^+$ represents the frequency of the amino acid pair $b$ formed by $l$-th and $(l+k+1)$-th amino acids in positive training set, $n_{b,(l,l+k+1)}^-$ represents corresponding frequency in negative training set.

We all know that 21 amino acids can form 441 amino acid pairs (AA, AC, ..., XX). So, when the length of the sequence fragment is $L$ and $k = 0$, we can get $p_{b,(l,l+k+1)}^+$ and $n_{b,(l,l+k+1)}^-$ from the 441×(L-1)-dimensional matrices $P_{\text{pair}}$ and $N_{\text{pair}}$ respectively, which defined as follows:
\[ P_{\text{pair}} = \begin{bmatrix} p_{1,(1,2)} & p_{1,(2,3)} & \ldots & p_{1,(L-1,L)} \\ p_{2,(1,2)} & p_{2,(2,3)} & \ldots & p_{2,(L-1,L)} \\ \vdots & \vdots & \ddots & \vdots \\ p_{441,(1,2)} & p_{441,(2,3)} & \ldots & p_{441,(L-1,L)} \end{bmatrix} \]

\[ N_{\text{pair}} = \begin{bmatrix} n_{1,(1,2)} & n_{1,(2,3)} & \ldots & n_{1,(L-1,L)} \\ n_{2,(1,2)} & n_{2,(2,3)} & \ldots & n_{2,(L-1,L)} \\ \vdots & \vdots & \ddots & \vdots \\ n_{441,(1,2)} & n_{441,(2,3)} & \ldots & n_{441,(L-1,L)} \end{bmatrix} \]

If the length of the sequence fragment is \( L \) and \( k = 0, 1, 2 \ldots l \), both \( P_{\text{pair}} \) and \( N_{\text{pair}} \) are \( 441 \times (L-1+L-2+\ldots+L-1) \)-dimensional matrices. And the feature vector \( T \) is:

\[ T_{k=0,1,\ldots,L-1} = (I_{1,0}, I_{2,0}, \ldots, I_{L-1,0}, I_{1,1}, I_{2,1}, \ldots, I_{L-1,1}, \ldots, I_{1,L-1}, I_{2,L-1}, \ldots, I_{L-1,L-1}) \]

In this work, we let \( k \) be 0, 1, \ldots, 19, and the PKSAAP obtains 210-dimensional feature vectors.

2.3. Feature Selection

We union all their features and normalize their values in order to achieve better performance. Finally, we obtain a 391-dimensional \((160+21+210)\) feature vector. The five-fold cross-validation and extremely randomized trees method [22] are used to select the optimal feature set.

The extremely randomized trees method is a tree-based ensemble algorithm. The basic idea of this method is splitting the samples in a dataset based on the information gain into a tree whose nodes denote the features, and calculate the importance of each feature using the samples routed to the corresponding node. For example, we have a sample set and a feature set. This algorithm randomly chooses \( k \) features from the feature set, and for each one of the \( k \) features, randomly chooses a feature value as the cut-point. Then the samples can be divided into two subsets, one with samples whose values on this feature are greater than the cut-point and one with others. Then it can calculate the information gain of such a classification. Among those \( k \) features, the one that has the max information gain is chosen as the root node, and the same way is used to split each subset. The split results of the two subsets will be the left sub-tree and right sub-tree. Iteratively, this algorithm can construct a tree whose internal nodes are the features. In this tree, each feature's importance can be represented by the proportion of the samples routed to the corresponding node. Construction of multiple trees can increase the stability of feature selection, and the average value of each feature's importance is its importance score.

The information gain used in above procedure can be defined according to a previous research [22], whose value can be calculated using the following equation:

\[ I_s = \frac{2M_{c,s}(P)}{H_s(P) + H_c(P)} \]  

in which \( P \) is the sample set, \( c \) is a classification in \( P \), \( s \) is a split of the feature, \( M_{c,s}(P) \) is the mutual information of the split outcome and the classification, \( H_s(P) \) is the entropy of the classification, and \( H_c(P) \) is the split entropy.

2.4. Evaluation Criterion

In this paper, we use the five-fold cross-validation to select the best-performed prediction model on the training set, then test the model using the independent test to further evaluate the effectiveness. Five metrics are used to evaluate the predictor, which are Sensitivity (Sn), Specificity (Sp), Accuracy (Acc), Mathews Correlation Coefficient (MCC), and Area under ROC (AUC). The area under the curve (AUC) is the proportion of the space below the receiver operating characteristic curve (ROC) to the whole space, and this value can directly illustrate the stability of performance of a classifier.

We choose support vector machines (SVM) to train the prediction model, which has been successfully applied on the prediction of PTMs sites [23, 24]. With proper parameters, SVM has good performance in small sample sets and unbalanced datasets. Both in our training set and independent test set, the ratio of the positive samples and the negative samples is 1:4.
3. Results and Discussion

3.1. Performance Analysis

We use the above feature extraction methods and their combinations to train five classifiers separately. All of them are adjusted to their optimal status. And the results are list in table 1.

From table 1, among the three individual feature extraction methods, HQI8 is the best. Between the two combined methods, PSAAP+HQI8+PKSAAP method is better than PSAAP+HQI8. We can see that PSAAP and HQI8 features are useful for predicting amidation sites and PKSAAP features can slightly improve the performance by including the information of amino acid pairs. So we choose the best one, PSAAP+HQI8+PKSAAP for the final prediction.

| Method        | Sn   | Sp   | Acc  | MCC  | AUC  |
|---------------|------|------|------|------|------|
| PKSAAP        | 68.5%| 94.6%| 87.7%| 0.66 | 0.928|
| PSAAP         | 86.1%| 96.4%| 94.2%| 0.82 | 0.955|
| HQI8          | 88.5%| 96.7%| 95.0%| 0.85 | 0.961|
| PSAAP + HQI8  | 92.1%| 97.0%| 96.0%| 0.88 | 0.961|
| PSAAP + HQI8 + PKSAAP | 92.7%| 97.2%| 96.2%| 0.89 | 0.964|

We also compare our method with existing method PrAS [17]. The PrAS achieves 81.2% on Sn, 94.9% on Sp, 92.1% on Acc, 0.76 on MCC, 0.96 on AUC. We can see that our method is better than PrAs with great improvement while we can reach to 92.7% on Sn, 97.2% on Sp, 96.2% on Acc, 0.89 on MCC and 0.964 on AUC. Specifically, the values of Sn, Sp, Acc, MCC and AUC have an improvement by 11.5%, 2.3%, 4.1%, 0.13 and 0.004. The method proposed by Cui et al. [16] also can predict amidation sites, but we do not compare our method with it because the data set they upload is not complete.

| Method       | Sn   | Sp   | Acc  | MCC  | AUC  |
|--------------|------|------|------|------|------|
| Naive Bayes  | 86.7%| 94.5%| 92.8%| 0.79 | 0.945|
| KNN          | 89.1%| 97.0%| 95.4%| 0.86 | 0.952|
| SVM          | 92.7%| 97.2%| 96.2%| 0.89 | 0.964|

Besides training the SVM classifier, KNN and Naive Bayes classifiers were also trained and tested on the independent test set with the 55-D optimal feature set. The performance of those classifiers are listed in table 2. We can see that SVM classifier is the best classifier on this prediction problem.

3.2. Feature Analysis

In the optimal feature set which is 55-dimensions, 21 dimensions are selected from PKSAAP, while HQI8 and PSAAP respectively contribute 22 and 12 dimensions. Although PSAAP features have the least size, more than half are selected when constructing the optimal feature set, indicating that these features are very suitable for the prediction of amidation sites.
Figure 1. Two Sample Logo of the positive and negative training datasets. The enrichment difference between the two datasets are more significant at position 13, 14, 11 and 10.

We filter out the top 10 features, and list them in descending order as 13-PSAAP, 14-PSAAP, 12-13-PKSAAP, 13-MIYS990104, 13-NAKH920108, 13-BIOV880101, 10-PSAAP, 13-TSAJ990101, 14-TSAJ990101 and 11-PSAAP. Two Sample Logo [25] can show the difference between the two datasets about the tendency of amino acids at different positions. For example, in Figure 1, the same information about PSAAP features is illustrated that amino acids at position 10, 11, 13 and 14 of the fragment have more significant distribution. And we can find out that the 13-th position of amidated fragments is enriched with amino acid K and R, the 14-th position and 10-th position are enriched with R and 11-th position is enriched with F, V and L.

Our method can achieve very good performance which benefits from those advantages: 1) HQI8 is obtained by clustering a variety of physical properties. 2) Amino acids at some positions have specific distribution pattern, on which PSAAP is suitable. 3) PKSAAP method can obtain the information about amino acid pairs, which can slightly improve the performance.

4. Conclusion

In this paper, we propose a computational method to predict the amidation sites. Our method utilizes three feature extraction methods to represent the sequence fragments and uses extremely randomized trees algorithm to construct the optimal feature set. In independent test set, our method has great performance and can reach to 92.7% on Sn, 97.2% on Sp, 96.2% on Acc and 0.89 on MCC. The AUC score of our method is 0.964. Compared with other existing predictors, our method is better than them. We hope that our method can help researchers to predict and validate amidation sites. Most significantly, our approach provides a new perspective to the amidation sites prediction and can inspire more related research works on fragment's position specificity.

5. Acknowledgments

This work was supported by the National Natural Science Foundation of China(61930007), the National High Technology Research and Development Program of China (863 Program)(2015BA3005) and the National 973 Program (2013CB32930X).

6. Reference

[1] Rudd P M, Elliott T, Cresswell P, Wilson I A and Dwek R A 2001 Science 291 2370-2376
[2] Moschos S J, Jukic D M, Athanassiou C, Bhargava R, Dacic S, Wang X, Kuan S F, Fayewicz S L, Galambos C, Acquafondata M et al. 2010 Human pathology 41 1286-1298
[3] Driscoll J J, Pelluru D, Lefkimiatis K, Fulciniti M, Prabhala R H, Greipp P R, Barlogie B, Tai Y T, Anderson K C, Shaughnessy J D et al. 2010 Blood 115 2827-2834
[4] Bennett E J, Bence N F, Jayakumar R and Kopito R R 2005 Molecular cell 17 351-365
[5] Eipper B A, Sto_ers D A and Mains R E 1992 Annual review of neuroscience 15 57-85
[6] Mueller G P and Driscoll W J 2008 Post-translational Modifications of Proteins: Tools for Functional Proteomics 68(11) 67-84
[7] Shimosawa T, Kanozawa K, Nagasawa R, Mitarai T, Isoda K, Takahashi K, Ando K, Tozawa Y, Nagase M, Sasaki N et al. 2000 Hypertension Research 23 167-171
[8] Dennison S R, Harris F, Bhatt T, Singh J and Phoenix D A 2009 Molecular and cellular biochemistry 332 43-50
[9] Bousquet-Moore D, Prohaska J R, Nilnni E A, Czyzyk T, Wetsel W C, Mains R E and Eipper B A 2010 Neurobiology of disease 37 130-140
[10] Kuyama H, Nakajima C, Nakazawa T, Nishimura O and Tsumasawa S 2009 Proteomics 9 4063-4070
[11] Chou K C 2011 Journal of theoretical biology 273 236-247
[12] Xu Y, Shao X J, Wu L Y, Deng N Y and Chou K C 2013 PeerJ 1 e171
[13] Lv H, Han J, Liu J, Zheng J, Liu R and Zhong D 2014 PloS one 9 e111478
[14] Chen Z, Chen Y Z, Wang X F, Wang C, Yan R X and Zhang Z 2011 PloS one 6 e22930
[15] Xu Y, Ding Y X, Deng N Y and Liu L M 2016 Gene 576 99-104
[16] Cui W, Niu S, Zheng L, Hu L, Huang T, Gu L, Feng K, Zhang N, Cai Y and Li Y 2013 Molecular genetics and genomics 288 391
[17] Wang T, Zheng W, Wuyun Q, Wu Z, Ruan J, Hu G and Gao J 2017 Computational biology and chemistry 66 57-62
[18] Kawashima S, Pokarowski P, Pokarowska M, Kolinski A, Katayama T and Kanehisa M 2007 Nucleic acids research 36 D202-D205
[19] Trost B and Kusalik A 2013 Bioinformatics 29 686-694
[20] Saha I, Maulik U, Bandyopadhyay S and Plewcynski D 2012 Amino Acids 43 583-594
[21] Xu Y, Ding J, Wu L Y and Chou K C 2013 PLoS One 8 e55844
[22] Geurts P, Ernst D and Wehenkel L 2006 Machine learning 63 3-42
[23] Zhou Y, Zhang N, Li B Q, Huang T, Cai Y D and Kong X Y 2015 Journal of Biomolecular Structure and Dynamics 33 2479-2490
[24] Du Y, Zhai Z, Li Y, Lu M, Cai T, Zhou B, Huang L, Wei T and Li T 2016 Journal of proteome research 15 4234-4244
[25] Vacic V, Iakoucheva L M and Radivojac P 2006 Bioinformatics 22 1536-1537