2023 26th International Conference on Computer and Information Technology (ICCIT)
13-15 December, Cox's Bazar, Bangladesh

Propionylation Characterization Ensembling Classifiers on Varied Features

Zannatul Ferdousee
Computer Science & Engineering
Rajshahi University of Engineering & Technology,
Rajshahi-6204, Bangladesh
methela1603060@gmail.com

S.M. Shovan
Computer Science & Engineering
Rajshahi University of Engineering & Technology,
Rajshahi-6204, Bangladesh
sm.shovan@gmail.com

Md. Sharifujjaman
Computer Science & Engineering
Rajshahi University of Engineering & Technology,
Rajshahi-6204, Bangladesh
sharifjaman2499@gmail.com

***Abstract*—A recently discovered post-translational modification (PTM) called propionylation where propionyl group is added in the lysine amino acid's side chains causing diversity and evolution. This alteration occurs in both eukaryotic and prokaryotic cellular environments under a variety of circumstances. It plays a significant part in metabolic functions, the control of biomolecules and several biological processes with consequences for a number of unhealthy disorders. One approach is to use mass spectrometry methods that distinguish propionylation PTMs in a laboratory setting. However, this procedure is characterized by a slow pace and significant cost inefficiency. The development of computational models has the potential to replace traditional laboratory techniques, increasing its availability to users of all skill levels worldwide. We developed a simplified efficient technique that combines positional and evolutionary data from classical machine learning with VGG-16 and RNN architectures applied to semantic embedding features taken from natural language processing. We were able to obtain accuracy, sensitivity along with specificity and MCC of 96.53 %, 97.27 %, 94.56 % and 0.53 % respectively which fared better than the most recent state-of-the-art models built.**

***Index Terms*—PTM, Propionylation, PLMD, CKSAAP, VGG-16, SVM, RNN, protvec.**

I. Introduction

Protein formation from DNA is a process known as post-translational modification. In the fundamental dogma, transcription and translation are two equally important processes. Transcription is associated using the conversion of double-helical DNA into single-helical mRNA and translation is associated with using the information carried by mRNA to create three-dimensional protein structures. The entire process is subject to varied modifications depending on the situation. Such as, Amino acids, the structural element of proteins, undergo post-translational changes to their side chains, whereas nucleotides are responsible for DNA and RNA [1]. Examples of post-translational changes that are important in controlling protein function and cellular activities include acetylation, methylation, and phosphorylation. Acetylation is the process of attaching an acetyl group to particular amino acid residues, which frequently improves the stability of proteins and gene expression. Methylation, on the other hand, involves the addition of a methyl group and has an impact on DNA repair, gene expression, and protein-protein interactions [2] One of the most important mechanisms for regulating protein activity, signal transduction, and cellular reactions is phosphorylation, which is the addition of a phosphate group.

A well-known laboratory technique for the characterization of a wide range of PTMs is mass spectrometry, which uses a mass spectrometer [3]. The use of this method has a number of downsides; as a manual procedure it necessitates complete expert engagement. However, it takes a long time and costs money for identifying, necessitating a substantial financial investment in laboratory setup and maintenance. As a outcome all of the aforementioned challenges might be eliminated by efficient and accurate computational forecasting systems.

A technique called PropPred [4] was created at the beginning of 2017 to forecast Propionylation sites using variables from the amino acid index database that are based on amino acid frequency such as combinations of the physicochemical characteristics, amino acid composition, and K-spaced amino acid pairings. For the classification process it didn't consider any evolutionary data. Contrarily PropSeek [5] made advantage of frequency-based, physicochemical-based and evolutionary information but since it used an undersampling strategy to balance the dataset, it lost important information and performed only moderately well. In 2020, Ahmed et al [6] employed only evolutionary data by comparing the query sequences to their special database after a few years. This method was different from that of PropSeek and the modification improved their performance. In 2021, Ang et al [7] provided a technique for computationally predicting propionylation sites that is based on transfer learning. Although their performance is comparable to that of the Ahmed et al. [6] research, they linked RNN and SVM to classify using a

979-8-3503-5901-5/23/$31.00

single hot encoding and ignored the other feature extraction techniques. In the most recent research M Zhang et al [8] outlines the global landscape of acylation in the Gram-positive model organism Bacillus and discusses potential metabolic and physiological roles for each type of acylation. For our suggested model The SMOTE oversampling technique was employed to prevent information loss while we extracted data from two domains, including amino acid frequency and evolutionary features to address the problem of data imbalance. These features were then used to feed three distinct, improved classifiers to determine the best result.Previous computer models, such PropPred [4] and PropSeek [5] have limits when it comes to taking into account a variety of attributes and striking a balance between dataset types. This means that in order to overcome these obstacles, new computational techniques must be developed.

The following are some contributions from our work.

1) In order to detect the unidentified propionylation PTM, we suggested a new ensembled classification technique combining both conventional ML and renowned deep neural network architectures.
2) Our classifiers are fed features from diversified domain including positional features, evolutionary features and word embedding technique for the context of protein domain, thus outperforms the existing models.

Our approach is motivated by the shortcomings of current laboratory techniques, in particular the resource- and time-intensive nature of mass spectrometry. To expedite the detection and characterisation of PTMs, precise and efficient computational models are required, while manual approaches require expert engagement. We summarized our work in this paper in four sections. The first section is *Introduction* in which we discussed what PTM is and why computational models are important. We also did the literature review in this section. The section two *Material and Methods* explains the dataset, feature extraction techniques, imbalance dataset management and classification techniques. *Result and Discussion* is the third section in which we depicted our achieved result with the help of different evaluation metrics and did the comparison with the existing models. The final section is *Conclusion* which is the summary of whole paper.

II. Material and methods

Figure 2 depicts the workflow of the proposed methodology. In this research work, at first the benchmark dataset was prepared. Next we we had evaluated three distinct approaches for extracting features from the prepared dataset. After that classification had been carried out by using SVM classifier, VGG16 and deep RNN architectures. Finally we had introduced the 'Tie Breaker Algorithm' to enhance the model's performance.

A. Benchmark dataset

The Protein Lysine Modification Database (PLMD) [9], from which the dataset was compiled, has information on twenty various post-translational modifications, including protonylation. The dataset of protonylation consists of 192 proteins with 413 propionylation (positive) and 3531 non-propionylation (negative) sites. The relevant residue for the propionylation site in this instance is the amino acid lysine (K) which is represented by the one-letter symbol and is present in the middle of the peptide sequence. The length of the window size is $2 \times \xi + 1$. In contrast, for the peptide sequence's downstream and upstream residues $\xi = 12$ and our window size is 25. X has been introduced as a dummy residue when necessary to ensure equal length. We employ the widely used technique CD-hit [10] to remove the superfluous homologous sample from the peptide sequence. CD-HIT is only being applied to negative sites rather than positive sites in order to maintain the sites ratio. Then the negative sequence decreased from 3531 to 1192 (0.4) with a 40% cutoff. After that we divided the total number of instances at random into 90% training instances and 10% test instances. $P_\xi(\odot)$ is used to symbolize each peptide which is given by Equation 1. [2]

$$P_\xi(\odot) = R_{-\xi}R_{-(\xi-1)}\dots R_{-2}R_{-1} \odot R_1R_2\dots R_{+(\xi-1)}R_{+\xi} \tag{1}$$

Eq.2 reflects the propionylated or positive samples that were obtained $P_\xi^+(\odot)$ and $P_\xi^-(\odot)$ reflects sites that are negative or not protonylated which results to 413 and 3531 samples respectively.

$$P_\xi(\odot) \in \begin{cases} P_\xi^+(\odot), & \text{Positive site} \\ P_\xi^-(\odot), & \text{Negative site} \end{cases} \tag{2}$$

B. Feature Extraction

1) Evolutionary features: We evaluated three distinct approaches including evolutionary which takes account of mutation-based data, to provide numerical values or features related to each of the peptide sequences. To create a Position Specific Scoring Matrix (PSSM), we used the NCBI-designed PSI-Blast [11] tool. PSI-Blast runs each peptide sequence against an existing database where in this case UniProt/SwissProt is reviewed and divides the database into two groups like in our base paper: i) homologous or sequences with an e-value less than the cutoff which is 0.001 ; and ii) non-homologous or sequences with an e-value greater than the cutoff. [2] It uses deep learning principles and multiple sequence alignment to determine score and determine how similar the sequences are. To create the PSSM matrix, which represents the tendency of mutation for each value, only homologous sequences are taken into account. The method is iterative where after each iteration the query sequence is applied once more to the most recent PSSM matrix until the required number of iterations, in this case three has been completed. For the next step, the final PSSM matrix is taken into consideration. Eq.3 is the representation of the query sequence for the window size $L = 25$.

$$P = R_1R_2R_3R_4R_5 \cdots\cdots\cdots\ R_L \tag{3}$$

The PSSM matrix having $L \times 20$ dimension represented in a transposed normalized (z-score) form in eq.4 where $E_{i\to j}$ determines the proclivity from i^{th} to j^{th} amino acids. The

range of i is from 1 to L and j is from 1 to 20. Each position's value can be either positives or negatives with varying magnitudes. The possibility of mutation increases as the value rises and vice versa.

$$M = \begin{bmatrix} \acute{E}_{1\to1} & \acute{E}_{2\to1} & \cdots \cdots & \acute{E}_{L\to1} \\ \acute{E}_{1\to2} & \acute{E}_{2\to2} & \cdots \cdots & \acute{E}_{L\to2} \\ \vdots & \vdots & & \vdots \\ \acute{E}_{1\to20} & \acute{E}_{2\to20} & \cdots \cdots & \acute{E}_{L\to20} \end{bmatrix} \quad (4)$$

This matrix is multiplied by the transposition of it resulting in a matrix with a dimension of 20×20. This matrix's distinctive feature is that its lower and upper triangular matrices are identical. The comparable feature vector with the dimension 1×210 has only been generated using the lower triangular matrix with the diagonal values in order to avoid using similar information.

2) Composition of K-spaced Amino Acid Pairs-CKSAAP: Protein or peptide sequences can be represented as numerical values using the CKSAAP[15] method which is based on the permutation of each of the twenty pairs of amino acids with a gap from 0 to K. Here K is the tuning parameter and in this case it is set to 5. The number of features will be $(K + 1) \times 400$ or 2400 in our example, because 400 features (= 20×20, where 20 is the number of amino acids) are generated for each phase.

3) Protvec word embedding [12]: The Protvec method of representing the protein sequence (peptide sequence in our case) into numerical values preserves the word relationships among the peptide sequences. To built up the dictionary of words, we split the peptide sequence into subsequence of length three, then find the word embedding as mentioned in the paper [12]. The word embedding represents a vector of length n for each of the words. For m words in peptide sequence, the scaling factor of $(244/n, 244/m)$ is then applied to match the dimension of VGG-16. For the channel match, we duplicated the matrix three times.

C. Imbalance Dataset Management

The dataset with unevenly distributed observations throughout the target classes, with one class label having a very high number of observations and the other having a very low number is referred to imbalanced data. How correctly are we truly predicting both the majority and minority class when dealing with an imbalanced dataset is the fundamental issue. In the dataset we used, the imbalanced ratio was 2.89. The negative and positive sites belong to the majority and minority classes respectively. Due to poor testing performance, the model had overfitting and underfitting issues for both random oversampling and undersampling respectively. SMOTE's synthetic data generation did not solve the generalization problem [13]. Cluster-centroid based undersampling on the evolutionary data (explained on the feature extraction section) is utilized to preserve the majority class's most spatial relationships without degrading the overall property, giving us the imbalance ratio of 1:1. [14].

D. Classification

1) Support Vector Machine: One of the most successful supervised learning classifiers is the support vector machine (SVM), which uses the support vectors (the data points needed to generate the hyperplane) to determine the best hyperplane by maximizing the margin's width [15] and for pattern recognition and data classification it is a discriminative classifier [16]. We fed evolutionary features of dimension 210 and amino acid positional based feature CKSAAP of dimension 2400 together by concatenating them and scaling the values between [0,1].

The input samples were transformed into a higher dimensional space by a kernel function based on the binary classification of samples that are leveled as positive or negative. Then, by using a separating hyperplane, a hyperplane is determined for discriminating between the two classes with a maximum margin and a minimum error. The radial basis function (RBF), which is defined as $(S_i, S_j) = \exp\left(-\gamma \|S_i - S_j\|^2\right)$, has been discussed in a number of earlier publications [17], [18] and is arguably the best option for SVM classifier learning. A gamma parameter controls the RBF kernel, while a cost parameter regulates the hyper-plane softness.In protein data, post-translational modifications frequently introduce complex patterns and non-linear interactions. These complexity are well captured by the RBF kernel, which enables the SVM to discover non-linear and nuanced correlations that may be important for locating propionylation sites.

The regularization tuning parameter, C is used to prevent overfitting and underfitting. Another parameter that affects the separation of the separating plane is the kernel function. By examining a wide range of values, we were able to determine that 64 and 0.001 were the optimal values for C and gamma (the tuning parameters for the RBF kernel) in our model [19]. With respect to area under the curve, the optimal hyperparameter value is selected.

2) VGG16: A 16-layer deep neural network makes up the VGG16 model which is a variant of the VGG model that consist of 13 convolution layers, each having ReLU activation and 3 fully connected layers. It consists of 13 convolutional layers arranged in 5 convolutional blocks with a maxpooling layer following next to each block. The 5 maxpooling layers after each block were able to accomplish the goal of producing a classification model that was more accurate while also taking less time to compute.

As VGG-16 uses an image as an input, one-hot encoding has been used to represent each amino acid in protein sequences as a numerical vector. The word embedding is represented into a matrix and then scaled up by interpolation to match the input size of (224x224x3) where the channels are duplicated for our case. However, we added one more layer at the end of output layer of VGG-16 of dimension 2 with softmax activation function that takes input from 1000 layer which is the last

layer of VGG-16. $\{0, 1\}$ and $\{1, 0\}$ are one hot representation of class positive and negative respectively.

Figure 1 shows the VGG16 network's architectural layout.

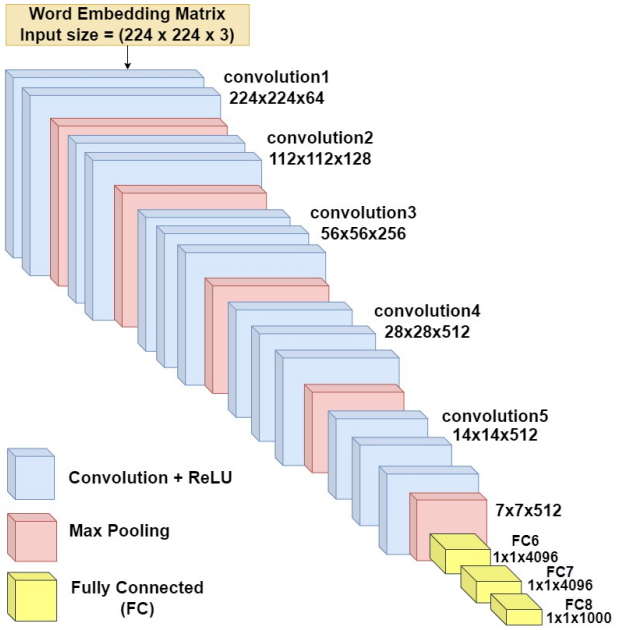

Fig. 1. Architecture of VGG16.

3) LSTM: The deep RNN model consisted of one embedding, two LSTM, one Gated Recurrent Unit (GRU), one dropout, one flatten, one fully connected and one output layer. Instead of word embedding into embedding vectors, the embedding layer converted integer amino acid character indices. The architecture has been taken from the paper [7].

In our experiment, merging RNN's networks performance was not helping with SVM and VGG-16, rather it was badly impacting. However, we experimented that when there is a case of tie (one says positive while other says negative) for VGG-16 (CNN) and SVM (conv), if we break the tie with the following algorithm using RNN, the performance get a slight boost.

Algorithm 1 Tie Breaker Algorithm

```
if (conv = TRUE) ∧ (CNN = TRUE) then
    return TRUE
else if (conv = FALSE) ∧ (CNN = FALSE) then
    return FALSE
else
    if RNN_proba ≥ 0.5 then
        return TRUE
    else
        return FALSE
    end if
end if
```

E. Experimental Setting

The optimal PSI-Blast parameters, the number of iterations (iter) and the similarity cutoff e-value (e-val), have been modified by trial and error. The ideal iter and e-val values were discovered to be 3 and 0.001, respectively. To achieve the best results, the classifier's parameters were adjusted using the 10-fold cross-validation model. In SVM, C and gamma (RBF) are set to 64 and 0.001, respectively. There was no depth constraint, and each node in the decision tree-based classifier received 10 trees.

F. Measuring Matrices

The following equations have been used to determine Accuracy (Acc), Sensitivity (Sn), Specificity (Sp) and Matthews correlation coefficient (Mcc) in order to evaluate the performance of the proposed model.

$$\text{Acc} = \frac{TP + TN}{TP + FP + TN + FN} \tag{5}$$

$$\text{Sn} = \frac{TP}{TP + FN} \tag{6}$$

$$\text{Sp} = \frac{TN}{TN + FP} \tag{7}$$

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

For the binary classification problem, here the labels TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative) correspond to the respective cells in the confusion matrix [1].

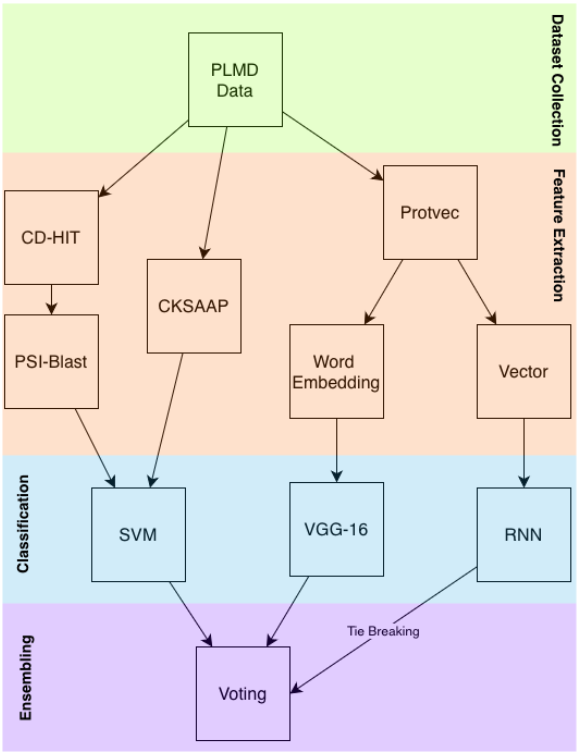

Fig. 2. Flow diagram/ Pipeline of the proposed model.

III. Results and discussion

We proposed a hybrid model combining different aspects of features for both support vector machine as conventional ML and VGG-16 and RNN model. The SVM takes features of two different aspects, i) evolutionary features, ii) positional features, i.e. CKSAAP. For evolutionary features, we incorporated PSI-BLAST with e-value cutoff 0.001 which is considered as standard and number of iteration is 3 as increasing the value was not helping much in terms of accuracy. We used k=5 for CKSAAP technique which we found by trial and error basis. However, on the other hand, we used protvec, a technique inspired from the word embedding, feature extraction where the word are considered each three consecutive amino acids, on modified VGG-16. Later on, feature vector is further used on RNN to break the tie as shown in Algorithm 1. The RNN model itself did not show sufficient performance compared to other, so we only used it for breaking the tie. We used 10-fold cross validation on training data which was split in 90-10% manner. Figure 2 shows the flow of the proposed model. The next section contains the performance we achieved.

A. Comparison with existing models

TABLE I
Comparison of the cross-validation performance between recently developed tools and our models using various classifiers

Model	Acc (%)	Sn (%)	Sp (%)	Mcc
PropPred [20]	75.02	70.03	75.61	0.3085
PropSeek [21]	79.92	-	-	-
TL [7]	83.06	84.54	81.58	0.6615
SVM Alone	95.03	98.34	92.45	0.73
VGG-16 Alone	91.97	90.56	92.93	0.49
RNN Alone	79.26	68.36	90.95	0.53
Proposed Model	96.53	97.27	94.56	0.53

For propionylation prediction, two computing methods that were available in recent works were the PropPred [20] and the PropSeek [21]. The proposed method performs better than the PropPred, which is based on 10-fold cross-validation with 250 optimal features and a window size of 25. The proposed method also outperforms the PropPred in terms of Acc, which is based on 10-fold cross-validation with 187 dimension features. Later on for computationally predicting propionylation sites a transfer learning (TL) based method [7] was presented. Except for the MCC, our proposed method outperforms their method which is based on 10-fold cross-validation with a 29-window size.

In Table I, we separately run each part of our proposed model and found that SVM has been performing better in terms of accuracy among other two singular models. However, VGG-16 itself cannot compete with SVM for our case. We suspected that there can be few example which are correctly predicted by VGG-16 which SVM could not. So, we incorporated both of the models into one. To use the voting scheme as a label decision in ensemble, we also incorporated RNN based model into our model. However, RNN model struggled in terms of accuracy while having acceptable specificity. So, we did not put RNN as equally important as other two models, rather we only used RNN for breaking the tie between SVM and VGG-16. The proposed model is the model shown in figure 2 having the highest performance gain among all of the models. We believe our ensemble model outperforms the existing methods in this problem domain.

IV. Conclusion

In this paper, we propose an ensembled classification technique that integrates conventional and deep neural network architectures on various features, including amino acid positional and evolutionary features, among others. Using the utility cd-hit, we first eliminated duplicates with 40% similarity. PSI-Blast was then used to construct an evolutionary features PSSM matrix. Alternatively, we also generated positional features CKSAAP for K=5. Incorporating deep neural networks such as VGG-16 and RNN can enhance the conventional machine learning technique SVM. However, the feature vector produced by the Protvec tool is inputted into the first layer so that the model considers it an image. The RNN model does not perform as well as SVM and VGG-16, but the pre-history and post-history features are significant because the amino acids upstream and downstream of the lysine residue are influential. Therefore, we utilized the label produced by RNN to break the deadlock for the voting scheme using Algorithm 1. Figure 2 depicts the entire classification technique workflow. Our model surpasses the most advanced models currently available. We believe that the enhanced performance is a result of the incorporation of diverse features and classification methods.

References

[1] S. Shovan, M. A. M. Hasan, and M. R. Islam, "Improved prediction of glutarylation ptm site using evolutionary features with lightgbm resolving data imbalance issue," in *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*. IEEE, 2021, pp. 141–145.

[2] ——, "Accurate prediction of formylation ptm site using multiple feature fusion with lightgbm resolving data imbalance issue," in *2020 23rd International Conference on Computer and Information Technology (ICCIT)*. IEEE, 2020, pp. 1–6.

[3] S. Shovan and M. A. M. Hasan, "Prediction of lysine glycation ptm site in protein using peptide sequence evolution based features," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*. IEEE, 2019, pp. 1–5.

[4] Z. Ju and J.-J. He, "Prediction of lysine propionylation sites using biased svm and incorporating four different sequence features into chou's pseaac," *Journal of Molecular Graphics and Modelling*, vol. 76, pp. 356–363, 2017.

[5] L.-N. Wang, S.-P. Shi, P.-P. Wen, Z.-Y. Zhou, and J.-D. Qiu, "Computing prediction and functional analysis of prokaryotic propionylation," *Journal of chemical information and modeling*, vol. 57, no. 11, pp. 2896–2904, 2017.

[6] M. W. Ahmad, M. E. Arafat, S. Shovan, M. Uddin, O. F. Osama, and S. Shatabda, "Enhanced prediction of lysine propionylation sites using bi-peptide evolutionary features resolving data imbalance," in *2020 IEEE Region 10 Symposium (TENSYMP)*. IEEE, 2020, pp. 1668–1671.

[7] A. Li, Y. Deng, Y. Tan, and M. Chen, "A transfer learning-based approach for lysine propionylation prediction," *Frontiers in physiology*, vol. 12, p. 452, 2021.

[8] M. Zhang, T. Liu, L. Wang, Y. Huang, R. Fan, K. Ma, Y. Kan, M. Tan, and J.-Y. Xu, "Global landscape of lysine acylomes in bacillus subtilis," *Journal of Proteomics*, vol. 271, p. 104767, 2023.

[9] H. Xu, J. Zhou, S. Lin, W. Deng, Y. Zhang, and Y. Xue, "Plmd: An updated data resource of protein lysine modifications," *Journal of Genetics and Genomics*, vol. 44, no. 5, pp. 243–250, 2017.

[10] W. Li and A. Godzik, "Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences," *Bioinformatics*, vol. 22, no. 13, pp. 1658–1659, 2006.

[11] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped blast and psi-blast: a new generation of protein database search programs," *Nucleic acids research*, vol. 25, no. 17, pp. 3389–3402, 1997.

[12] E. Asgari and M. R. Mofrad, "Continuous distributed representation of biological sequences for deep proteomics and genomics," *PloS one*, vol. 10, no. 11, p. e0141287, 2015.

[13] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[14] Y.-P. Zhang, L.-N. Zhang, and Y.-C. Wang, "Cluster-based majority under-sampling approaches for class imbalance learning," in *2010 2nd IEEE International Conference on Information and Financial Engineering*. IEEE, 2010, pp. 400–404.

[15] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, "Support vector machine classification and validation of cancer tissue samples using microarray expression data," *Bioinformatics*, vol. 16, no. 10, pp. 906–914, 2000.

[16] V. N. Vapnik, "An overview of statistical learning theory," *IEEE transactions on neural networks*, vol. 10, no. 5, pp. 988–999, 1999.

[17] S.-L. Weng, H.-J. Kao, C.-H. Huang, and T.-Y. Lee, "Mdd-palm: Identification of protein s-palmitoylation sites with substrate motifs based on maximal dependence decomposition," *PloS one*, vol. 12, no. 6, p. e0179529, 2017.

[18] H.-J. Kao, S.-L. Weng, K.-Y. Huang, F. J. Kaunang, J. B.-K. Hsu, C.-H. Huang, and T.-Y. Lee, "Mdd-carb: a combinatorial model for the identification of protein carbonylation sites with substrate motifs," *BMC systems biology*, vol. 11, pp. 127–140, 2017.

[19] K. Duan, S. S. Keerthi, and A. N. Poo, "Evaluation of simple performance measures for tuning svm hyperparameters," *Neurocomputing*, vol. 51, pp. 41–59, 2003.

[20] Z. Ju and J.-J. He, "Prediction of lysine propionylation sites using biased svm and incorporating four different sequence features into chou's pseaac," *Journal of Molecular Graphics and Modelling*, vol. 76, pp. 356–363, 2017.

[21] L.-N. Wang, S.-P. Shi, P.-P. Wen, Z.-Y. Zhou, and J.-D. Qiu, "Computing prediction and functional analysis of prokaryotic propionylation," *Journal of chemical information and modeling*, vol. 57, no. 11, pp. 2896–2904, 2017.