# A Multimodal Deep Architecture for Large Scale Protein Ubiquitylation Site Prediction

1st Fei He
*School of Information Science and Technology*
*Northeast Normal University*
Changchun, China
hef740@nenu.edu.cn

2nd Lingling Bao
*School of Information Science and Technology*
*Northeast Normal University*
Changchun, China
baoll601@nenu.edu.cn

3rd Rui Wang
*School of Information Science and Technology*
*Northeast Normal University*
Changchun, China
wangr921@nenu.edu.cn

4th Jiagen Li
*School of Information Science and Technology*
*Northeast Normal University*
Changchun, China
lijg803@nenu.edu.cn

5th Xiaowei Zhao
*School of Information Science and Technology*
*Northeast Normal University*
Changchun, China
zhaoxw303@nenu.edu.cn

***Abstract*** In eukaryotes, protein ubiquitylation is an important kind of post-translation modifications, in which the ubiquitin conjugates to a substrate protein. To have a better insight of the mechanisms underlying ubiquitylation, an initial but important step is to identify protein ubiquitylation sites. Many existing computational methods are based on feature engineering, which may lead to biased and incomplete features, especially on large scale data. Deep learning provides multiple layer networks and non-linear mapping operations to detect potential complex patterns in data-driven way. It has been considered promising to solve the problems in existing machine learning method. In this paper, we proposed a multimodal deep architecture for large scale protein ubiquitylation sites prediction. First, we designed different multiple layers to extract hidden informative patterns from three modalities, namely protein fragments, physio-chemistrical properties, and PSSM. Then, the deep representations corresponding to three modalities were merged to implement the classification. On the available largest scale protein ubiquitylation site database PLMD, the performance of our proposed method was measured with 66.7% sensitivity, 66.4% specificity, 66.43% accuracy, and 0.221 MCC value. A range of comparative experiments also showed that our proposed architecture outperformed several popular protein ubiquitylation sites prediction tools. Our source codes are freely available at https://github.com/jiagenlee/deepUbiquitylation.

***Keywords—Protein Ubiquitylation Site Prediction, Multiple Modalities, Deep Learning, Convolution Neural Network, Deep Neural Network***

## I. INTRODUCTION

Ubiquitin is a small protein consists of 76 amino acids[1, 2]. The conjugation of ubiquitin to substrate protein on particular lysine is an important kind of post-translation modifications, which is called ubiquitylation [3, 4]. Protein ubiquitylation, which includes three kinds of enzymes (activating enzymes, ligases and conjugating enzymes), plays an important role in various cellular functions, such as signal transduction, apoptosis and cell proliferation [5, 6]. The conventional experimental techniques such as CHIP-CHIP analysis and mass spectrometry are usually time-consuming, laborious and expensive to detect protein ubiquitylation sites. Thus, the computational approaches that could effectively and accurately identify the protein ubiquitylation sites are urgently needed.

Many computational methods have been developed for the identification of protein ubiquitylation sites. Huang et al. established a predictor called UbiSite, which used a two layered machine learning method with substrate motifs to predict protein ubiquitylation sites [7]. Nguyen et al. proposed a new scheme to characterize and identify protein ubiquitylation sites, and this method included three features including amino acid composition, evolutionary information and amino acid pair composition. Additionally, the motif discovery tool, MDDLogo, was also used in their predictor[8]. Qiu et al. selected sequence evolutionary information and gray system model to construct a protein ubiquitylation site predictor named iUbiq-Lys [9]. UbiProber presented by Chen et al. is another computational protein ubiquitylation site prediction tool, which combined sequence information, physico-chemistrical properties and amino acid composition to Support Vector Machine (SVM) for identifying potential protein ubiquitylation sites [10]. Wang et al. developed an improved protein ubiquitylation sites predictor named ESA-UbiSite using an evolutionary screening algorithm (ESA) to select the effective negative samples from the non-validated lysine sites, and then an ESA-based prediction model was established [11].

These existing machine learning based approaches perform effectively on small scale data, however, some shared

challenges on large scale protein ubiquitylation site prediction still need to address: (1) Weakness of handcrafted protein features. The traditional feature engineering way relied on expert knowledge usually leads to biased and incomplete feature vectors; (2) Heterogeneity among different shallow representations. Most protein ubiquitylation site prediction tools combined multimodal features to improve their accuracies, but neglect the intrinsic heterogeneity among such shallow representations. (3) Unbalanced distributions between positive and negative samples. Only a small size of lysine residue can be attached to ubiquitin in whole proteome, however, existing methods cannot function well to accurately identify potential protein ubiquitylation site under such extreme unbalanced circumstance. Deep learning, as a cutting-edge machine learning technique for big data, has been considered promising to tackle these problems. It provides multiple layer networks and non-linear mapping operations to detect potential complex patterns from raw input signals, and generates homogenous deep representations for classification tasks. The deep learning framework synchronously generates novel features and conducts the classification according to input raw signals in a data-driven way, which could keep away from feature engineering and reduce the mismatch between feature extraction and classifier. Diverse types of deep learning networks have been successfully utilized to genomic and proteomic analyses and researches [12-14], however, there is barely report about applying deep learning technique to protein ubiquitylation site prediction.

In this paper, we proposed a multimodal deep architecture fusing three different categories of protein modalities for large scale protein ubiquitylation site prediction, i.e. raw protein sequence fragment, selected physico-chemistrical properties of amino acids, as well as its corresponding position-specific scoring matrix (PSSM). In the deep architecture, we employed multiple convolution layers as the feature extractor to generate protein sequence representations, and brought several stacked fully connected layers to combine the physico-chemistrical properties of amino acids, and used other multiple convolution layers as a detector to discover evolutionary profile around potential ubiquitylation site. These multiple modalities were transformed into more compatible and abstract representations by our deep architecture. Finally, we integrated these hidden layers in the network to a softmax layer for predicting protein ubiquitylation sites. To the best of our knowledge, this is the first deep architecture for identifying protein ubiquitylation sites. In the comparisons with several recent state-of-the-art protein ubiquitylation site prediction tools, our approach exhibits more encourage performance.

## II. Material and Methods

### A. Large Scale Dataset Collection

For large scale protein ubiquitylation site prediction, we collected 25103 proteins with 121742 ubiquitylated sites from version 3.0 of Protein Lysine Modification Database (PLMD), which is a comprehensive dataset for 20 types of protein lysine modifications, and extends from CPLA 1.0 dataset and CPLM 2.0 dataset. So far as we known, this is the available largest scale protein ubiquitylation site database, and is never mentioned in any other protein ubiquitylation site prediction research. In order to avoid overestimation caused by homologous sequences, we used CD-HIT program [15] to filter the homologous sequences with 40% sequence similarity in all data, and obtained 17406 proteins with 60879 annotated protein ubiquitylation sites. These protein sequences were divided into training dataset and testing dataset by random partition. The training dataset comprised 12100 protein sequences with 54586 ubiquitylation sites while the independent testing dataset consisted of 1345 proteins with 6293 ubiquitylation sites. According to these annotated information and protein sequences, we extracted 427305 and 46080 non-annotated ubiquitylation sites regarded as negative samples from training dataset and independent testing dataset respectively. To construct the training and testing samples, we intercepted a protein fragment with central lysine residue and fixed window length of $2n+1$ for considering $n$ upstream and downstream flanking amino acids around targeting lysine residue as a sample. Furthermore, to prevent the interference that some negative training samples may be homologous to positive training samples, the tool cd-hit-2d was utilized to remove the negative samples with 50% similarity to positive samples [7]. In order to achieve unbiased models, we extracted a small proportion 30% of training samples as validation samples by random sampling in each training iteration. Finally, we obtained the experimental datasets as Table I summarized.

TABLE I. Brief description of collected protein Ubiquitylation site data

| Data set | Description | | | |
|---|---|---|---|---|
| | ***Number of sequences*** | ***Number of positive data*** | ***Number of negative data*** | ***Note*** |
| Training | 12100 | 38211 | 224059 | Random partitioning in each training iteration |
| Validation | | 16375 | 96024 | |
| Testing | 1345 | 6293 | 46080 | Reservation |

### B. Encoding of Protein Segments

In this paper, three types of quantized biological descriptors are employed to encode all involving protein samples.

*1) One hot vector:* each sample contained $m$ amino acids is represented as a $m \times k$ 2-dimensional (2D) matrix, which uses a $k$ dimensional zero vector with a one in the index corresponding to the amino acid in the protein sequence. When the left or right neighboring amino acids cannot fit the window size, a dash will be filled in these positions and be encoded to 0.05. In such encoded scheme, every protein fragment will be mapped to a exclusive and sparse coding, which quantifies amino acids and maintains their relative positions.

*2) Physico-Chemistrical Proverties:* Some researches indicate that there is a strong connection between physico-chemistrical proverties of amino acids and ubiquitylation sites[16] [17]. And physico-chemistrical proverties have been widely used in many types of protein post-translation modification such as phosphorylation, acetylation and

sulfation [10]. Such physico-chemistrical proverties of each amino acid can be found in an AAindex database [18]. Among the 544 physico-chemistrical metrics recorded in AAindex, we only select top thirtheen physico-chemistrical proverties that have been validated by comparing the prediction accuracy of all physico-chemistrical proverties in light of literature[10], and thus form a $m$×13 2D matrix as another encoding modality for each sample.The details of the selected physio-chemistrical properties are given in Table II.

TABLE II. THE SELECTED PHYSICO-CHEMISTRICAL PROPERTIES

| Physico-chemistrical property | Description |
|---|---|
| EISD860102 | Atom-based hydrophobic moment |
| ZIMJ680104 | Isoelectric point |
| HUTJ700103 | Entropy of formation |
| KARP850103 | Flexibility parameter for two rigid neighbors |
| JANJ780101 | Average accessible surface area |
| FAUJ880111 | Positive charge |
| GUYH850104 | Apparent partition energies calculated from Janin index |
| JANJ780103 | Percentage of exposed residues |
| JANJ790102 | Transfer free energy |
| PONP800102 | Average gain in surrounding hydrophobicity |
| CORJ870101 | NNEIG index |
| VINM940101 | Normalized flexibility parameters, average |
| OOBM770101 | Average non-bonded energy per atom |

*3) PSSM Profile:* PSSM is also employed here to represent the evolutionary profile of the protein sequence. We set the non-redundant Swiss-Prot as the search database, and generate the raw PSSMs of all involving protein sequences using the Basic Local Alignment Search Tool (BLAST) with the parameter "-j 3 -h 0.001". In a raw PSSM, it sets 20 dimensional vector to demonstrate the preference of 20 types of amino acids at each position of protein seqence. For the purpose of focusing on the potential ubiquitylation sites, we extract the PSSM fragment corresponding to the window size $m$ from the PSSM result of whole protein sequence which indicates the position-specific evolutionary profile of amino acids neighboring potential ubiquitylation sites. Thus, we obtain a $m$×20 2D matrix as PSSM modality.

## C. Multimodal Deep Architecture Construction

The presented deep architecture, as shown in Fig. 1, includes three parts of sub-nets to separately deal with the above mentioned three kinds of input modalities, and then merges their output hidden states for classification.

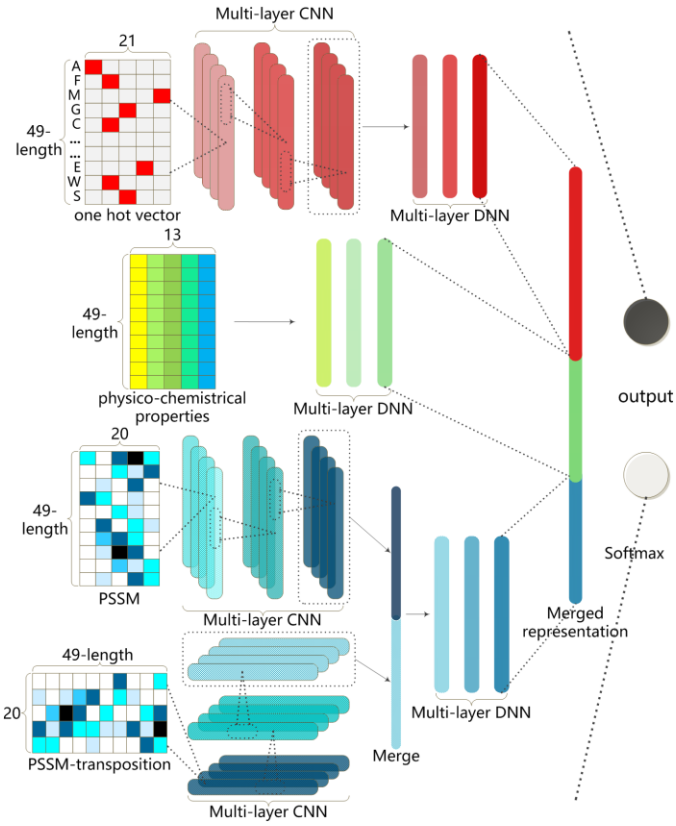


Fig. 1. The structure of our proposed deep architecture

For a given one hot vector, a one dimensional Convolution Neural Network (1D CNN) with 3 hidden layers is designed to extract its implicit local features. Due to the natural data sparsity of one hot vector, the locally connected convolution layers encode it into a range of feature maps, which explicit subtle structural features hidden in raw protein sequence. After the hierarchical convolution, all feature maps will be merged together and generated lower dimensional states by 3 fully connected hidden layers. Such sub-net can detect informative sequential representations in its hidden states.

For the physio-chemistrical properties of corresponding amino acids, we introduce a Deep Neural Network (DNN) with 3 hidden layers to generate their deep representations. Since their components describe the characteristic of potential ubiquitylation sites from different viewpoints, the fully connected DNN structure could interconnect all factors for their joint effect in its hidden states.

For the input PSSM, we also employ 1D CNN with 3 hidden layers to detect potential informative descriptions among amino acids through evolution to the protein fragment. Different from the sub-net of one hot vector, the trans-positioned PSSM vector is then sent into another 1D CNN with 3 hidden layers to obtain deep evolutionary characterization among different sequence positions. The feature maps from the two 1D CNNs are merged together to produce complete PSSM representations by 3 following fully connected hidden layers.

Subsequently, the output layer states of three sub-nets are merged into a mixed representation for fusing the three deep

representations of input multiple modalities at higher level. The merged layer is fully connected to a 2-state output layer for implementing binary classification by softmax function. These deep representations eliminate mutual heterogeneity among their raw shallow representations, therefore, they are more readily used for fusion. The weights between merged layer and output layer may be considered as the contributions of the three deep representations. All hyper-parameters of our proposed architecture are detailed in Table III.

TABLE III. THE HYPER-PARAMETERS OF PROPOSED DEEP ARCHITECTURE

| Subnet | Layer | Hyper-parameters | | | |
|---|---|---|---|---|---|
| | | *Activation function* | *Size*[c] | *Filters* | *Dropout* |
| One hot vector | 1D Convolution | softsign | 2 | 200 | 0.4 |
| | | softsign | 3 | 150 | 0.4 |
| | | softsign | 5 | 150 | 0.4 |
| | | softsign | 7 | 100 | 0.4 |
| | Dense[a] | relu | 256 | -- | 0.3 |
| | | relu | 128 | -- | 0 |
| | | relu | 128 | -- | -- |
| Phsico-chemistrical properties | Dense | softplus | 1024 | -- | 0.2 |
| | | softplus | 512 | -- | 0.4 |
| | | softplus | 256 | -- | 0.5 |
| | | relu | 128 | -- | -- |
| PSSM profile | 1D Convolution | relu | 1 | 200 | 0.5 |
| | | relu | 8 | 150 | 0.5 |
| | | relu | 9 | 200 | 0.5 |
| | 1D Convolution[b] | relu | 1 | 200 | 0.5 |
| | | relu | 3 | 150 | 0.5 |
| | | relu | 7 | 200 | 0.5 |
| | Dense | relu | 128 | -- | 0.3 |
| | | relu | 128 | -- | 0 |
| Merged representations | Dense | softmax | 2 | -- | 0 |

[a.] Dense layers represent for the fully connected layers in keras.

[b.] The layers were designed for trans-positioned PSSM profile

[c.] The size of convolution layers means the kernel sizes, and the size of Dense layers denotes the number of hidden states.

In this study, we introduce a training trick to accelerate the training procedure of the proposed multi-modal deep architecture. Considering the multi-modal subnets, we separately trained each subnet to guarantee the optimality of their weights, and then reloaded these trained weights to the whole multi-modal deep architecture as its initialization. In the training process of whole network, these weights and the weights of last merged layer would be fine tuned until they achieved global optimum. Meanwhile, in order to eliminate the influence caused by the extremely unbalanced distribution of positive and negative samples, we implemented the training procedure of the whole deep architecture and subnets following the bootstrapping strategy. Let *pos* and *neg* denote the number of positive and negative samples respectively. Owing to the relatively small size of positive samples $neg \gg pos$, we randomly chosen *pos* negative samples to form a balanced training dataset with all positive samples in each bootstrapping iteration. Therefore, all negative samples were divided into $N = \lfloor neg / pos \rfloor$ bins, and the deep architecture will be trained $N$ times for modeling a classifier. Such bootstrapping strategy can involve as many as training samples in classification model on the premise of unbiasedness. The early stop rule [19] was adopted to control epoch numbers here, and the training procedure stops automatically by the time the validation accuracy has been stable for default epoch iterations (we set 50 here).

We built this deep architecture using Theano 0.9 and keras 1.1.0, and ran on a graphic processing units (GPU) GTX1080Ti. Taking advantage of GPU computations, we can obtain a trained deep model in 30 minutes.

## III. RESULTS AND DISCUSSION

### A. *Performance of our Mutimodal Deep Architecture*

First, we would like to report our experiments of different window sizes of protein fragments. Owing to its direct effect on the available information involving in prediction algorithm, the best suited window size to our deep architecture should be determined. Some researches adopted empirical value directly, however, different representations and classifier prefer different window sizes [20]. Thus we conducted a series of tests using the window length *m* from 7 to 61 (*n* is from 3 to 30). For each window length, we encoded all training protein fragments into three kinds of input modality and trained their corresponding subnet. The trained subnets were designated to predict the three types of input modality from the validation samples separately. The performance of different window sizes on one hot vector, physico-chemistrical properties and PSSM can be observed in Figure 2.

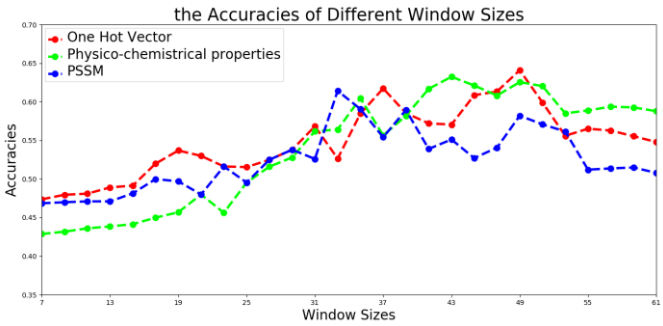


Fig. 2. The accuracy of validation samples using different window sizes on three modalities

Figure 2 suggested that when the window length got 49, the three types of representations might achieve comparable accuracy to other window sizes. This conclusion was inconsistent with some existing studies [7, 10], which indicated

that our deep architecture needed long distance sequence fragments to introduce more raw information for further detecting deep features.

Next, the whole multi-modal network was trained using one hot vector, physico-chemistrical property and PSSM profile inputs synchronously. The trained whole network and trained subnets were tested on independent testing set. Their generative ROC (receiver operating characteristic) curves and precision-recall curves were plotted in Figure 3.

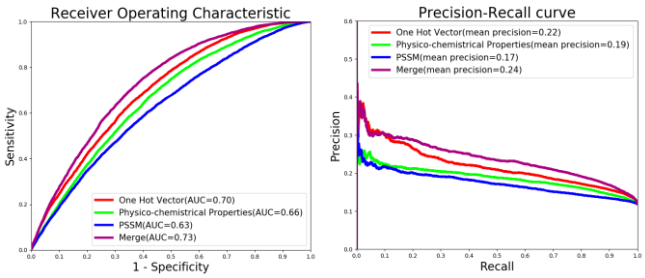


Fig. 3. ROC and precision-recall curves comparing our multi-modal network and subnets of uni-modality

From Figure 3, we can read that the whole multi-modal network showed slight improvements on ROC and precision recall curves. Its AUC (area under the ROC curves) and mean precision (area under the precision-recall curves) reached 0.73 and 0.24, which were beneficial from the data-driven combination way. In the training process of the whole deep architecture, we pre-loaded the weights of trained subnets to ensure that the subnets generate optimal deep representations of one hot vector, physico-chemistrical property and PSSM profile. And then, a supervised fine tune was started to modify the weights of merged layer adaptively. Such process continued until the deep architecture made all input modalities at full capacity. Figure 3 also manifested that one hot vector performed best among the three input modalities. It can be inferred that a proper deep learning network may detect underlying informative expressions from raw protein sequence fragments. In order to validate this merit, we visualized the states in original input layer and the merged layer of whole model using t-SNE[21], to observe the overlap of positive samples and negative samples in independent testing set. The visualization results are shown in Figure 4.

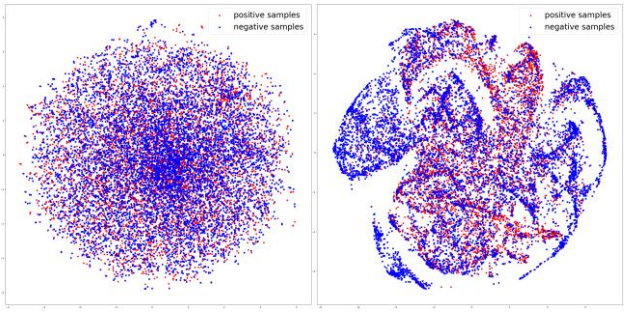


Fig. 4. t-SNE visualization of (a) input layers and (b) merged layer

From the Figure 4, we can see that in the 2D distribution, the positive samples and negative samples were in mixture, which brought the difficulty to direct classification. As the samples were processed layer by layer in our deep architecture and the distinctive features were detected, the two classes of samples tended to separate. It implied that our proposed deep architecture may map raw multiple modalities to deep representations with discriminative ability.

### *B. Comparisons with SVM classifier*

In the next stage, we would like to compare our deep architecture with the most popular used protein ubiquitylation sites prediction classifier: SVM. For fair comparisons, we inputted three types of modalities one hot vector, physico-chemistrical properties, and PSSM profile to train SVM model directly. Considering the unbalanced training samples, we randomly extracted the same number of positive samples and negative samples in each training process. Subsequently, all the three modalities were concatenated into one vector, and this vector was sent to train another SVM model. All these models were trained with 10-fold cross-validation using the same experimental protocol. Their results adopting the best kernel RBF and default decision threshold 0.5 were combined with those of our deep architecture in Table IV.

TABLE IV. COMPARATIVE RESULTS WITH SVM CLASSIFIER

| **Model** | **Input** | **Metrics** | | | |
|---|---|---|---|---|---|
| | | ***Accuracy*** | ***Sensitivity*** | ***Specificity*** | ***MCC*** |
| SVM | One of Key | 14.80% | 95.73% | 3.76% | -0.009 |
| | Physico-chemistrical property | 14.35% | 96.22% | 3.16% | -0.011 |
| | PSSM | 78.82% | 13.21% | 87.72% | 0.009 |
| | Merged | 14.42% | 96.71% | 3.03% | -0.005 |
| Our deep architecture | One of Key | 61.84% | 64.41% | 64.08% | 0.189 |
| | Physico-chemistrical property | 61.84% | 60.97% | 61.95% | 0.151 |
| | PSSM | 56.82% | 58.73% | 56.57% | 0.099 |
| | Merged | 66.43% | 66.67% | 66.40% | 0.221 |

Table IV indicated that the SVM models tended to predict most of samples to one class erroneously. It was illustrated by a high sensitivity along with a extremely low specificity in the performance of SVM models using one hot vector and physio-chemistrical properties as inputs. It implied that nearly all testing samples were ridiculously classified to ubiquitylation sites. On the contrary, the SVM model using PSSM obtained a high specificity but a low sensitivity. It meant that almost testing samples were identified to non-ubiquitylation sites, and rare potential ubiquitylation sites were successfully detected. Such phenomenon showed that SVM model was incapacity of generating discriminative features from raw modalities for the two classes of samples. That may be the reason why existing

tools did not choose to input raw sequence fragments and protein properties, but further transformed these modalities into meaningful feature vectors, i.e. amino acid composition, for SVM training. While our deep architecture was able to adaptively detect useful information hidden in the raw modalities without feature engineering. Consequently, its experimental results looked more promising. The same situation occurred in the experiments of multi-modalities, which revealed that our deep architecture may handle multi-modal fusion problem better than SVM model. Due to their poor performance, the overall estimator Matthews correlation coefficients (MCC) of SVM models were much lower than the MCC of our architecture. It reflected that our bootstrapping training strategy may consolidate the generalization of our architecture on unbalanced training dataset from another respective.

### C. *Comparisons with Other Protein Ubiquitylation site Prediction Tools*

Furthermore, we compared our architecture with some popular protein ubiquitylation site prediction tools supporting batch sample mode, namely Ubisite[7], UbiProber[10], iUbiq-Lys[9],ESA-UbiSite[11]. We fairly submitted our testing protein sequences to their websites, and calculated all comparative metrics of all involving tools according to their feedback results as Table V shown.

TABLE V. COMPARATIVE RESULTS WITH OTHER PROTEIN UBIQUITYLATION SITE PREDICTION TOOLS

| Tool | Metrics | | | |
|---|---|---|---|---|
| | ***Accuracy*** | ***Sensitivity*** | ***Specificity*** | ***MCC*** |
| ESA-Ubisite | 61.26% | 46.14% | 63.34% | 0.064 |
| UbiProber | 55.06% | 62.40% | 54.05% | 0.107 |
| iUbiq-Lys | 84.63% | 3.35% | 96.88% | 0.005 |
| Ubisite | 73.63% | 29.62% | 79.64% | 0.073 |
| Our deep architecture | 66.43% | 66.67% | 66.40% | 0.221 |

In Table V, because the websites of UbiProber and iUbiq-Lys only returned the predicted decisions but not predicted scores, we computed their predicted metrics according to the classification results of the four tools for direct comparisons. From Table V, it can be found out that our deep architecture performed excellent in most estimators, reaching at 66.7% sensitivity, 66.4% specificity, 66.43% accuracy, and 0.221 MCC value with 0.5 decision threshold. Even though the accuracy of our model cannot compare with Ubisite and iUbiq-Lys, their exorbitant specificities implied that they classified most of testing samples into non-ubiquitylation sites. That matched the unbalanced negative distribution of testing samples, and led to higher accuracy. However, they may not be as effective in predicting potential ubiquitylation sites according to their lower sensitivities. Overall, our model achieved simultaneous improvement in both sensitivity and specificity, especially obtained highest sensitivity among all tools. These demonstrated that our deep architecture was more efficient and robust than the existing tools. Moreover, with the predicted scores Ubisite and ESA-Ubisite provided, we also plotted the ROC and precision-recall curves with AUC and mean precision of the two tools and our model as Figure 5 shown.

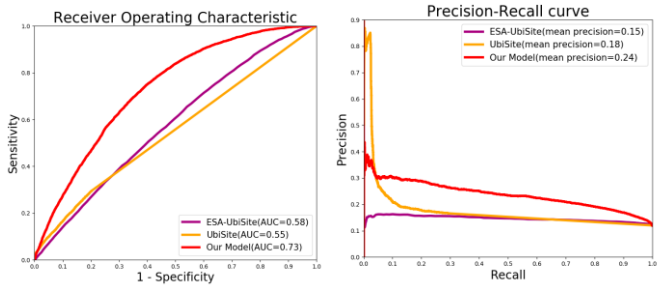


Fig. 5. The ROC and precision-recall curves comparing proposed deep architecture and other protein ubiquitylation site prediction tools

Figure 5 exhibited that our deep architecture performed a higher sensitivity under most certain specificity, and obtained better AUCs and mean precisions relative to other tools in most cases. It validated high confidence of our architecture on large scale protein ubiquitylation site data. It is worth noting that under certain minor recall, Ubisite achieved higher precision among the three methods. Its most probable reason is that Ubisite introduced more prior knowledge from positive training samples to its classification model. It divided positive training samples into 12 subgroups according to the clustered results of significant substrate motifs using MDDLogo tool [22]. And then it trained 12 sub-models using the 12 subgroups of positive training samples and the same number of negative samples to implement a boosting classification. Such classification models emphasized the feature patterns of positive samples, and guided to detect potential homologous protein fragments with high similarity to its positive training samples. Consequently, it resulted in better precision than that of our deep architecture only when the recall was less than 3.89%. Nevertheless, our deep architecture has evident overall advantages in term of ROC and precision-recall caurves.

Although our deep learning architecture has promoted the performance of protein ubiquitylation site prediction on large scale data, there is still room for improvement. In the future, we would like to continue studying the optimization strategy for guiding the selection of deep learning hyper-parameters, and co-operate with biologists to upgrade the model more biologically interpretable and reliable.

## IV. CONCLUSION

In this study, we proposed a multimodal deep architecture for large scale protein ubiquitylation sites prediction. Three kinds of modalities including one hot vector, physio-chemistrical properties and PSSM, which have been demonstrated to be associated with ubiquitylation, were firstly used to encode each input protein fragment. Then a multimodal deep architecture fusing these encoding modalities was established for robust classification. Experimental results on the available largest scale protein ubiquitylation site dataset have proved the effectiveness of the proposed method to deal

with the large scale data. The t-SNE visualization results also indicated that our deep architecture may generate more discriminative features from multiple modalities. The comparative experiments validated that our model outperformed several popular protein ubiquitylation site prediction tools. The success of our method is due to some reasons, including the data-driven feature detection in deep learning, the multimodal fusion of deep representations, and the bootstrapping algorithm.

## ACKNOWLEDGMENT

This research is partially supported by National Natural Science Foundation of China (61403077), the China Postdoctoral Science Foundation funded project (2015T80285), the Scientific and Technological Development Program of Jilin Province (20170520058JH), and the Natural Science Foundation of the Education Department of Jilin Province (2016-505).

## REFERENCES

[1] G. Goldstein, M. Scheid, U. Hammerling, D. H. Schlesinger, H. D. Niall, and E. A. Boyse, "Isolation of a polypeptide that has lymphocyte-differentiating properties and is probably represented universally in living cells," Proceedings of the National Academy of Sciences of the United States of America, vol. 72, no. 1, pp. 11, 1975.

[2] K. D. Wilkinson, "The Discovery of Ubiquitin-Dependent Proteolysis," Proceedings of the National Academy of Sciences of the United States of America, vol. 102, no. 43, pp. 15280, 2005.

[3] C. M. Pickart, and M. J. Eddins, "Pickart CM, Eddins MJ.. Ubiquitin: structures, functions, mechanisms. Biochim Biophys Acta 1695: 55-72," vol. 1695, no. 1-3, pp. 55-72, 2004.

[4] R. L. Welchman, C. Gordon, and R. J. Mayer, "Ubiquitin and ubiquitin-like proteins as multifunctional signals," Nature Reviews Molecular Cell Biology, vol. 6, no. 8, pp. 599, 2005.

[5] J. H. Hurley, S. Lee, and G. Prag, "Ubiquitin-binding domains," Biochemical Journal, vol. 6, no. 8, pp. 610, 2005.

[6] J. Peng, D. Schwartz, J. E. Elias, C. C. Thoreen, D. Cheng, G. Marsischky, J. Roelofs, D. Finley, and S. P. Gygi, "Peng, J. et al. A proteomic approach to understanding protein ubiquitination. Nature Biotech. 21, 921-926," Nature Biotechnology, vol. 21, no. 8, pp. 921-6, 2003.

[7] C. H. Huang, M. G. Su, H. J. Kao, J. H. Jhong, S. L. Weng, and T. Y. Lee, "UbiSite: incorporating two-layered machine learning method with substrate motifs to predict ubiquitin-conjugation site on lysines," Bmc Systems Biology, vol. 10 Suppl 1, no. Suppl 1, pp. 6, 2016.

[8] V. N. Nguyen, K. Y. Huang, C. H. Huang, K. R. Lai, and T. Y. Lee, "A new scheme to characterize and identify protein ubiquitination sites," IEEE/ACM Transactions on Computational Biology & Bioinformatics, vol. 14, no. 2, pp. 393-403, 2017.

[9] W. R. Qiu, X. Xiao, W. Z. Lin, and K. C. Chou, "iUbiq-Lys: prediction of lysine ubiquitination sites in proteins by extracting sequence evolution information via a gray system model," Journal of Biomolecular Structure & Dynamics, vol. 33, no. 8, pp. 1731, 2015.

[10] X. Chen, J. D. Qiu, S. P. Shi, S. B. Suo, S. Y. Huang, and R. P. Liang, "Incorporating key position and amino acid residue features to identify general and species-specific Ubiquitin conjugation sites," Bioinformatics, vol. 29, no. 13, pp. 1614, 2013.

[11] J. R. Wang, W. L. Huang, M. J. Tsai, K. T. Hsu, H. L. Huang, and S. Y. Ho, "ESA-UbiSite: accurate prediction of human ubiquitination sites by identifying a set of effective negatives," Bioinformatics, vol. 33, no. 5, pp. 661, 2017.

[12] H. Y. Xiong, B. Alipanahi, L. J. Lee, H. Bretschneider, D. Merico, R. K. C. Yuen, Y. Hua, S. Gueroussov, H. S. Najafabadi, and T. R. Hughes, "The human splicing code reveals new insights into the genetic determinants of disease," Science, vol. 347, no. 6218, pp. 1254806, 2015.

[13] J. Zhou, and O. G. Troyanskaya, "Predicting effects of noncoding variants with deep learning–based sequence model," Nature Methods, vol. 12, no. 10, pp. 931, 2015.

[14] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey, "Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning," Nature Biotechnology, vol. 33, no. 8, pp. 831, 2015.

[15] Y. Huang, B. Niu, Y. Gao, L. Fu, and W. Li, "CD-HIT Suite: a web server for clustering and comparing biological sequences," Bioinformatics, vol. 26, no. 5, pp. 680, 2010.

[16] C. W. Tung, and S. Y. Ho, "Computational identification of ubiquitylation sites from protein sequences," Bmc Bioinformatics, vol. 9, no. 1, pp. 310, 2008.

[17] P. Radivojac, V. Vacic, C. Haynes, R. R. Cocklin, A. Mohan, J. W. Heyen, M. G. Goebl, and L. M. Iakoucheva, "Identification, analysis, and prediction of protein ubiquitination sites," Proteins-structure Function & Bioinformatics, vol. 78, no. 2, pp. 365-380, 2010.

[18] S. Kawashima, H. Ogata, and M. Kanehisa, "AAindex: Amino Acid Index Database," Nucleic Acids Research, vol. 27, no. 1, pp. 368, 1999.

[19] Y. Yao, L. Rosasco, and A. Caponnetto, "On Early Stopping in Gradient Descent Learning," Constructive Approximation, vol. 26, no. 2, pp. 289-315, 2007.

[20] T. Chun-Wei, "Prediction of pupylation sites using the composition of k-spaced amino acid pairs," Journal of Theoretical Biology, vol. 336, no. 25, pp. 11-17, 2013.

[21] V. D. M. Laurens, G. Hinton, and V. D. M. Hinton, Geoffrey, "Visualizing Data using t-SNE," Journal of Machine Learning Research, vol. 9, no. 2605, pp. 2579-2605, 2008.

[22] T. Y. Lee, Z. Q. Lin, S. J. Hsieh, N. A. Bretaña, and C. T. Lu, "Exploiting maximal dependence decomposition to identify conserved motifs from a group of aligned signal sequences," Bioinformatics, vol. 27, no. 13, pp. 1780, 2011.