Computer Modeling in Engineering & Sciences
DOI:10.32604/cmes.2020.010347
Article

Tech Science Press

Prediction of Intrinsically Disordered Proteins with a Low Computational Complexity Method

Jia Yang[1], Haiyuan Liu[1,*] and Hao He[2]

[1]Nankai University, School of Electronic Information and Optical Engineering, Tianjin Key Laboratory of Optoelectronic Sensor and Sensing Network Technology, Tianjin, 300350, China
[2]Hebei University of Technology, Tianjin, 300400, China
[*]Corresponding Author: Haiyuan Liu. Email: liuhaiyuan@nankai.edu.cn

Received: 27 February 2020; Accepted: 21 July 2020

Abstract: The prediction of intrinsically disordered proteins is a hot research area in bio-information. Due to the high cost of experimental methods to evaluate disordered regions of protein sequences, it is becoming increasingly important to predict those regions through computational methods. In this paper, we developed a novel scheme by employing sequence complexity to calculate six features for each residue of a protein sequence, which includes the Shannon entropy, the topological entropy, the sample entropy and three amino acid preferences including Remark 465, Deleage/Roux, and Bfactor(2STD). Particularly, we introduced the sample entropy for calculating time series complexity by mapping the amino acid sequence to a time series of 0–9. To our knowledge, the sample entropy has not been previously used for predicting IDPs and hence is being used for the first time in our study. In addition, the scheme used a properly sized sliding window in every protein sequence which greatly improved the prediction performance. Finally, we used seven machine learning algorithms and tested with 10-fold cross-validation to get the results on the dataset R80 collected by Yang et al. and of the dataset DIS1556 from the Database of Protein Disorder (DisProt) (https://www.disprot.org) containing experimentally determined intrinsically disordered proteins (IDPs). The results showed that k-Nearest Neighbor was more appropriate and an overall prediction accuracy of 92%. Furthermore, our method just used six features and hence required lower computational complexity.

Keywords: Bioinformatics; intrinsically disordered proteins; machine learning algorithms; sequences; computational complexity

1 Introduction

Proteins are the material bases of life and the important component of all cells. IDPs are proteins which have at least a region lacking of well-defined three-dimensional structure in their native state [1,2]. These regions are popular in nature, which have functional roles in biological processes including transcription and translation [3]. They are associated with many diseases [4,5], such as genetic diseases, Parkinson's disease and cancer. Therefore, predicting IDPs is vital in the structural and functional analysis of proteins. We also know the disorder prediction has been successfully used to guide laboratory science [6].

Various methods for predicting IDPs have been proposed in the past few decades. These methods can be roughly divided into two categories: physicochemical-based and calculation-based. The first category is about employing the amino acid propensity scales and physicochemical properties of the protein sequences to predict IDPs, such as GlobPlot [7], IUPred [8], FoldUnfold [9] and IsUnstruct [10]. The second category is about employing machine learning and deep learning, such as Support Vector Machines (*SVM*), K-Nearest Neighbor (*KNN*), Convolutional Neural Network (*CNN*) and so on. These schemes include RONN [11], ESpritz [12], SPOT-Disorder [13], AUCpreD [14] and NetSurfP-2.0 [15]. It's worth mentioning that Wenliang Zhang et al., who proposed a new method to predict IDPs of different length disordered regions by using different feature teams in 2018. They used different features and its window size composing different feature teams and then every feature team could be used to train a predictor [16]. In 2019, Sarthak Mishra et al. employed a combination of 9890 features to train a supervised deep learning ensemble model. This system is a noval attempt in IDPs prediction [17]. In 2020, Jack Hanson et al. proposed SPOT-Disorder2 improve SPOT-Disorder [13]. SPOT-Disorder2 used long short term memory (*LSTM*) networks to predict disordered regions and obtained great performance [18]. However, a large number of features and deep learning models led to high computational complexity. Due to the high cost of these methods, we choose machine learning algorithm and fewer feature values to train a classification model and get prediction results. We know that the structure is determined by the primary sequence. So sequence information might also be important for predicting disordered regions. It has been demonstrated that disordered regions have lower sequence complexity and amino acid preferences than ordered [19]. According to these theories, the predictor presented in this paper computed six features for each residue of a protein sequence, consisting of the Shannon entropy, topological entropy, sample entropy and three amino acid preferences. However, adjacent residues have similar properties when forming proteins, and the disordered residues are often adjacent in protein sequences [20,21]. In order to increase interdependence between adjacent residues in a sequence, we used a sliding window to continuously intercept the region of window length for the protein sequence. The specific method is reflected in the following section. At last, using seven machine learning algorithms to train and test our system with 10-fold cross-validation. The results showed that *KNN* was more appropriate to train our scheme. Finally, we compared this scheme with SPOT–Disorder2 which is available as a web server and as a standalone program at [18]. The results showed that our scheme was more accurate. We achieved the purpose of improving the accuracy of prediction and reducing the computational complexity at the same time.

The Fig. 1 shows the flow chart of the proposed IDPs predictor and the following sections explain every step.

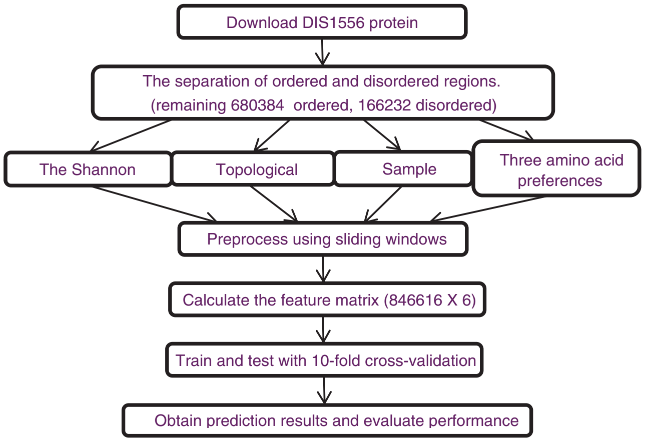

Download DIS1556 protein
The separation of ordered and disordered regions.
(remaining 680384 ordered, 166232 disordered)
The Shannon
Topological
Sample
Three amino acid preferences
Preprocess using sliding windows
Calculate the feature matrix (846616 X 6)
Train and test with 10-fold cross-validation
Obtain prediction results and evaluate performance

Figure 1: The flow chart of the proposed system

The experimental steps of this article are as follows:

Step 1: We need to download the latest 1556 intrinsically disordered protein sequences from DisProt (https://www.disprot.org).

Step 2: The DIS1556 data set includes 846616 amino acid residues. According to the tags in the database, the amino acids are divided into 680384 ordered and 166232 disordered. Then calculate the frequency of the use of these two categories of amino acids and correspond 20 amino acids into numbers 0–9 in order to calculate the sample entropy.

Step 3: Using a suitable sliding window, we need to calculate the Shannon entropy, sample entropy, topological entropy, and three amino acid preferences of the sequence within the window. As the window slides, a six–dimensional feature vector for each amino acid can be obtained. For the data set, we get a feature matrix of 846616 × 6.

Step 4: Our system used 10-fold cross-validation. The DIS1556 data set includes 846616 amino acid residues, 90% of which are randomly used as the train set and the other 10% as the test set. Then, we need to use different machine learning algorithms to train our system and get results. We can adjust every parameter of the model by fixing other parameters to get the best parameter combination. Then we need to calculate four metric: sensitivity (Sens), specificity (Spec), accuracy (ACC) and Matthews correlation coefficient (MCC).

Step 5: Finally, compare our scheme with the current excellent algorithm SPOT–Disorder2.

2 Features Selection and Preprocessing Procedure of IDPs Prediction

Features selection is an important step in building a predictor [22–26]. We know that disordered regions have less sequence complexity than ordered. We choose the Shannon entropy, topological entropy, sample entropy and three amino acid preferences to measure the complexity of a sequence.

Considering a protein sequence $\{w(j), 1 \le j \le N\}$ of length N, which $w(j) = \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q ,R, S, T, V, W, Y\}$, the Shannon entropy [27] can be expressed as:

$$H_s(w) = -\sum_{i=1}^{20} f_i \log_2 f_i \tag{1}$$

where f_i for $1 \le i \le 20$ can be defined as frequency of each amino acid in the protein sequence. If $f_i = 0$, let $f_i \log_2 f_i = 0$.

The topological entropy [28–31] depends on the complexity equation which describes the total number of different n-length subsequences of w:

$$p_w(n) = |\{w(u) : |u| = n\}| \tag{2}$$

where $w(u)$ is a subsequence of w, $|u|$ expressed the length of $w(u)$. For example, a given sequence $w = AAAFAA$, $n = 2$, the subsequence $w(u)$ of w are $\{AA, AF, FA\}$, thus $P_w(n) = 3$.

To solve short protein sequences, we map the amino acid sequence to the 0–1 sequence as shown in Tab. 1 as

Thus, the topological entropy can be defined as:

$$H_{top}(w) = \frac{\log_2 p_{w_1^{2^n+n-1}}(n)}{n} \tag{3}$$

which n should be the unique integer satisfying $2^n + n - 1 \le |w| \le 2^{n+1} + (n+1) - 1$ and $w_1^{2^n+n-1} = w(1) \ldots w(2^n + n - 1)$.

Table 1: Mapping relationship between amino acid sequence and 0–1 sequence

A	R	N	D	C	Q	E	G	H	S	T	K	M	L	P	I	F	W	Y	V
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

We used a sliding window to compute the average of topological entropy. If the length of window is *l*, then the topological entropy of *w* can be redefined as:

$$H_{top}(w) = \frac{1}{N-(2^n+n-1)+1} \sum_{i=1}^{N-(2^n+n-1)+1} \frac{\log_2 p_{w_l^{2^n+n-1+l-1}}(n)}{n} \tag{4}$$

In this paper, we introduced the sample entropy for the first time, and mapped the amino acid sequence to the time sequence of 0–9 as shown in Tab. 2 according to the tendency in the ordered and disordered regions.

Table 2: Mapping relationship between amino acid sequence and 0–9 sequence

A	S	E	G	K	P	R	T	H	M	D	Q	C	N	F	Y	I	V	L	W
0	0	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9

In general, given a time-series data $\{x(j), 1 \le j \le N\}$ of length *N*, which $x(j) = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. We need to construct $X_m(1), \ldots, X_m(N\text{-}m+1)$, where $X_m(i) = \{x(i), x(i+1), \ldots, x(i+m\text{-}1)\}$, $1 \le i \le N\text{-}m+1$. Then, we can define:

$$d[X_m(i), X_m(j)] = \max_{k=0,1,\ldots,m-1}(|x(i+k) - x(j+k)|) \tag{5}$$

For a given X_m (*i*) and count the number of *j* $(1 \le j \le N\text{-}m+1, j \ne i)$, where the distance between X_m (*i*) and X_m (*j*) is less than or equal to a given integer *r*, and record it as B_i, then, $B_i^m(r)$ and $B^m(r)$ can be defined as:

$$B_i^m(r) = \frac{B_i}{N-m-1} \tag{6}$$

$$B^m(r) = \frac{1}{N-m} \sum_{i=1}^{N-m} B_i^m(r) \tag{7}$$

Increasing the dimension to *m*+1 and counting the A_i, then, $A_i^m(r)$ and $A^m(r)$ can be defined as:

$$A_i^m(r) = \frac{A_i}{N-m-1} \tag{8}$$

$$A^m(r) = \frac{1}{N-m} \sum_{i=1}^{N-m} A_i^m(r) \tag{9}$$

The sample entropy can be calculated as follows:

$$H_{samp}(w) = -\ln\left[\frac{A^m(r)}{B^m(r)}\right] \tag{10}$$

For a protein sequence {*w(j)*, 1 ≤ *j* ≤ *N*} of length N, we need to compute the values of three amino acid preferences including Remark 465, Deleage/Roux, and Bfactor(2STD) given in the GlobPlot NAR paper [7] :

$$M_p(w) = \frac{1}{N}\sum_{l=1}^{N} w^p(l), p = 1, 2, 3 \tag{11}$$

where w^p(l) express the sequence after mapping according to the p-th preference as shown in Tab. 3.

Table 3: Mapping relationship of amino acid sequence according to p-th preference

	A	R	N	D	C	Q	E	G	H	I
$P = 1$	0.1739	−0.0537	−0.2141	0.2911	−0.5301	0.3088	0.5214	0.0149	0.1696	−0.2907
$P = 2$	−0.2750	−0.1790	0.4790	0.4645	−0.1255	−0.0550	−0.2745	0.6675	0.1350	−0.5150
$P = 3$	−0.1400	0.0633	0.2120	0.3480	−0.4940	0.1680	0.4560	0.1060	−0.0910	−0.4940
	L	K	M	F	P	S	T	W	Y	V
$P = 1$	−0.3379	0.1984	−0.1113	−0.8434	−0.0558	0.2627	−0.1297	−1.3710	−0.8040	−0.2405
$P = 2$	−0.4385	−0.0495	−0.4765	−0.4970	1.1170	0.2965	0.1450	−0.2570	0.0825	−0.7055
$P = 3$	−0.3890	0.4020	−0.1260	−0.5260	0.1800	0.1260	−0.0390	−0.7260	−0.5060	−0.4630

The preprocessing procedure is a very critical step. Given a protein sequence *w* of length *L*, we can change *w* into a new sequence of length *L* + *N* – 1 by choosing a sliding window of odd length *N* (*N* < *L*) and add N_0 = (*N* – 1)/2 zeros at each end of the sequence. For each area intercepted by the sliding window, a 6-dimensional vector V_i (1 ≤ *i* ≤ *L*) can be calculated, then assign the V_i to the every residue in the window. As the window slides, accumulate and compute the average of all V_i for each residue. By the way, each residue in the sequence will obtain a 6-dimensional feature vector X_j (1 ≤ *j* ≤ *L*):

$$X_j = \begin{cases} \dfrac{1}{j+N_0}\displaystyle\sum_{i=1}^{j+N_0} V_i, 1 \le j \le N_0 \\ \dfrac{1}{N}\displaystyle\sum_{i=j+N_0-N+1}^{j+N_0} V_i, N_0+1 \le j \le L-N_0 \\ \dfrac{1}{L+N_0-j+1}\displaystyle\sum_{i=j+N_0-N+1}^{L} V_i, L-N_0+1 \le j \le L \end{cases} \tag{12}$$

3 The Algorithms in Our Predictor

3.1 Linear Discriminant Analysis

Linear discriminant analysis (*LDA*) can give a linear hyperplane to make the Rayleigh entropy maximization:

$$J(W) = \frac{W^T S_B W}{W^T S_F W} \tag{13}$$

where S_F and S_B can be defined as:

$$S_F = \sum_{\mathrm{i}=1}^{2} \sum_{j=1}^{N_i} (x_j - m_i)(x_j - m_i)^T \tag{14}$$

$$S_B = (m_1 - m_2)(m_1 - m_2)^T \tag{15}$$

$$\mathrm{m}_i = \frac{1}{N} \sum_{\mathrm{j}=1}^{N_i} x_j \tag{16}$$

Using the Lagrange method:

$$L(W, \lambda) = W^T S_B W - \lambda(W^T S_F W - c) \tag{17}$$

It turns out that the best W is:

$$W = S_F{}^{-1}(m_1 - m_2) \tag{18}$$

3.2 Logistic Regression

Logistic regression (*LR*) using the sigmod function:

$$g(z) = \frac{1}{1 + e^{-z}} \tag{19}$$

We assume the prediction function is:

$$\mathrm{h}_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}} \tag{20}$$

$$\theta^T x = \sum_{i=0}^{n} \theta_i x_i \tag{21}$$

Cross entropy loss function which must be minimized is given by:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} (\mathrm{y}_i(\log h_\theta(x_i) + (1{-}\mathrm{y}_i) \log(1 - h_\theta(x_i))) \tag{22}$$

3.3 K-Nearest Neighbor

K-Nearest Neighbor (*KNN*) is a simple method to achieve classification. In the feature space, if most of the *k* nearest samples belongs to a certain category, the sample also belongs to this category. For example, there are two different types of data: squares and triangles, while the circular is the data to be classified.

If $K = 3$, the nearest three neighbors are two triangles and one square. Based on the statistical method, it is determined that the circular data should be classified to triangle class.

If $K = 5$, the nearest five neighbors are two triangles and three squares. So the circular data should be classified to square.

3.4 Naive Bayes Model

Naive Bayes model (*NBM*) is a classification method based on probability theory. Given a training set, with the assumption of independence between feature conditions, learn the joint probability distribution from input to output. Based on the learned model, input X to find the output Y that maximizes the posterior probability.

3.5 Support Vector Machines

For the binary classification problem, Support Vector Machines (*SVM*) can find a hyperplane that can split the training sample points into two parts, and ensure that the classification error rate is minimized (find the optimal hyperplane). The maximum width can be expressed as:

$$width = (x_+ - x_-) \cdot \frac{w}{\|w\|} = \frac{1-b}{\|w\|} - \frac{-1-b}{\|w\|} = \frac{2}{\|w\|} \tag{23}$$

The loss function can be defined as:

$$L = \frac{1}{2}\|w\|^2 - \sum \mathrm{a}_i[\mathrm{y}_i(w \cdot x_i + b) - 1] \tag{24}$$

3.6 Decision Tree and Random Forest

Decision tree (*DT*) and random forest (*RF*) represent a mapping relationship between object attributes and object values. Random forest is a classifier that uses multiple decision trees to train and predict samples.

4 Performance Evaluation

Four metrics are used to evaluate the performance: sensitivity (*Sens*), specificity (*Spec*), accuracy (*ACC*) and Matthews correlation coefficient (*MCC*) [32,33].

We use *TP* (true positive) and *TN* (true negative) to correspond to the number of correctly predicted disordered and ordered residues, *FP* (false positive) and *FN* (false negative) to indicate the number of residues misjudged as disordered and ordered. Sens is the ratio of positive samples which are correctly identified among all positive samples:

$$Sens = TP/(TP + FN) \tag{25}$$

Spec is the ratio of negative samples which are correctly identified among all negative samples:

$$Spec = TN/(TN + FP) \tag{26}$$

ACC and *MCC* can be calculated as follows:

$$ACC = \frac{1}{2}(Sens + Spec) \tag{27}$$

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP) \cdot (TP+FN) \cdot (TN+FP) \cdot (TN+FN)}} \tag{28}$$

5 Results and Discussion

5.1 Sequence Complexity

The Shannon entropy can effectively reflect sequence complexity. It has been verified by experiments that regions of protein with low complexity tend to be disordered (Fig. 2). So, the disordered region has more obvious amino acid usage preference, and the sequence of the disordered region is more conservative [34]. To further analyze the specific differences in amino acid composition between these

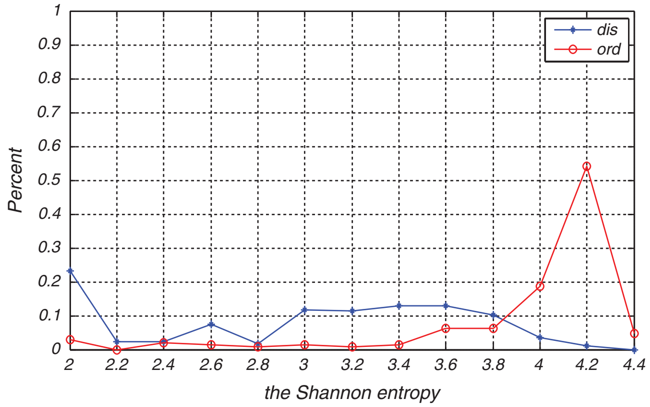


Figure 2: The percent of Shannon entropy in ordered and disordered

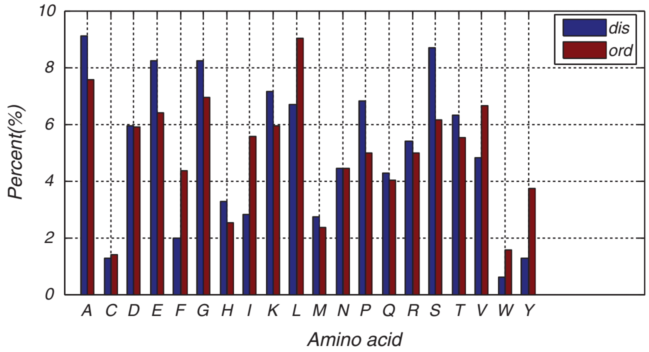


Figure 3: The percent of amino acid in ordered and disordered

two regions, we conducted a detailed analysis by calculating the percentage of 20 amino acids in their respective regions (Fig. 3). To measure intuitively that amino acids tend to be disordered or ordered, we introduced *AAP* [35]:

$$AAP^i_R = \frac{P^i_R}{P^i} - 1 \tag{29}$$

where *i* represents the *i*-th amino acid of the 20 amino acids, *R* represents the disordered or ordered region, and p^i represents the frequency of the *i*-th amino acid. While $AAP > 0$, it indicates that the amino acid is preferred, while $AAP = 0$, the usage is random, while $AAP < 0$, it indicates that the amino acid is not preferred (Fig. 4).

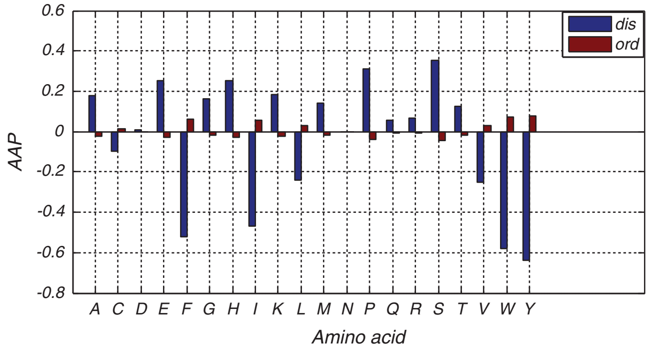


Figure 4: The AAP of amino acid in ordered and disordered

Comprehensive analysis of amino acid preference and sequence complexity, we use the sample entropy to describe sequence complexity firstly. To our knowledge, the sample entropy has not been previously used for predicting IDPs and hence is being used for the first time in our study. The following figure (Fig. 5) shows that there is a significant difference between the ordered and disordered regions.

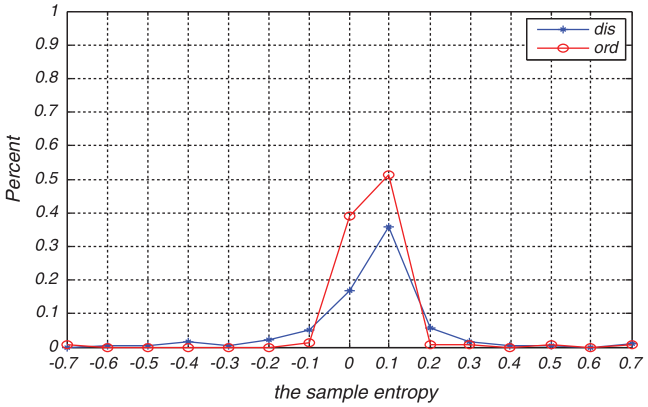


Figure 5: The percent of sample entropy in ordered and disordered

5.2 Impact of Input Features and Sliding Window

Firstly, fixing the machine learning algorithm to be LDA, Tab. 4 shows the predictive performance of different feature inputs on R80. We can find that the ACC and MCC increasing as the feature vector changes. So, we choose X_j as feature vector. In addition, using a sliding window of length 35 can greatly improve prediction performance.

Table 4: Performance comparison with different feature inputs and sliding window

Input Features	Win	Sens	Spec	ACC	MCC
$M_1(w_j)$	NO	0.8632	0.3106	0.5869	0.1190
$M_2(w_j)$	35	0.6859	0.8046	0.7453	0.3530
$M_3(w_j)$					
$H_s(w_j)$	NO	0.8079	0.3249	0.5664	0.0893
$M_1(w_j)$					
$M_2(w_j)$	35	0.7745	0.8372	0.8059	0.4529
$M_3(w_j)$					
X_j	35	0.7515	0.8763	0.8139	0.4963

5.3 Impact of Sliding Window

Next, fixing the feature inputs as X_j and the machine learning algorithm as *LDA*, Tab. 5 shows the prediction performance of different length of sliding window on R80. When the length of the window is larger than 35, the *ACC* tend to be stable. When the length is larger than 31, the *MCC* tends to be stable. So, we choose the sliding window of length 35 in our system.

5.4 Impact of Machine Learning Algorithm

Then, setting the size of window to be 35, using seven machine learning algorithms described in Section 3 to train our system and get the predictive performance on R80 as Tab. 6. Although we tried a number of

Table 5: Performance comparison with different feature inputs and sliding window

Input Features	Schemes	Win	Sens	Spec	ACC	MCC
X_j	LDA	11	0.6455	0.8425	0.7440	0.3725
		15	0.7196	0.8172	0.7684	0.3906
		19	0.7176	0.8512	0.7844	0.4343
		23	0.7070	0.8844	0.7954	0.4792
		27	0.7412	0.8664	0.8038	0.4731
		31	0.7367	0.8859	0.8113	0.5025
		35	0.7672	0.8644	0.8158	0.4876
		39	0.7440	0.8813	0.8127	0.4995
		43	0.7412	0.8820	0.8116	0.4987
		47	0.7549	0.8620	0.8085	0.4756
		51	0.7608	0.8462	0.8035	0.4562
		55	0.7123	0.8813	0.7968	0.4775

Table 6: Performance comparison with different feature inputs and machine learning algorithm

Schemes	Sens	Spec	ACC	MCC
LDA	0.7515	0.8763	0.8139	0.4963
LR	0.7196	0.8681	0.7938	0.4609
NB	0.6444	0.9350	0.7897	0.5409
SVM	0.3944	0.9856	0.6900	0.5163
DT	0.7750	0.9767	0.8759	0.7640
RF	0.8167	0.9932	0.9049	0.8604
KNN	0.9361	0.9938	0.9650	0.9357

machine learning algorithms, we found these seven performing well over the prepared data set, and thus reporting performance of these only.

The result shows that *KNN* is more appropriate and the overall prediction accuracy reaches 96%.

5.5 Impact of Parameters

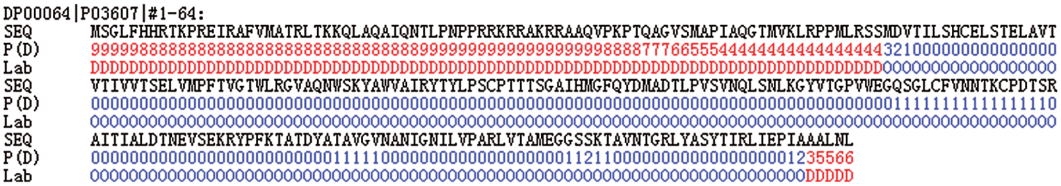
Finally, we adjust the parameters of the *KNN*. Let the NumNeighbors from 1 to 5 as Tab. 7. The result shows that the prediction performance is the best when the NumNeighbors equal to 1.

5.6 Comparing with SPOT-Disorder2

According to the experiments on R80 data set, we adjusted every parameter of our scheme. Then we use the DIS1556 data set which contains 846616 amino acid residues to compare our scheme with SPOT-Disorder2. The prediction results of SPOT-Disorder2 on data set DIS1556 come from their online predictors (https://sparks-lab.org/server/spot-disorder2/) as the following figure (Fig. 6) shows. The result in Tab. 8 shows that our scheme gets better *ACC* and *MCC*. In addition, the SPOT-Disorder2 used 73 input features including the hidden Markov model (*HMM*) profile and the position-specific substitution

Table 7: Performance comparison with different NumNeighbors

NumNeighbors	Sens	Spec	ACC	MCC
1	0.9361	0.9938	0.9650	0.9357
2	0.8806	0.9986	0.9396	0.9249
3	0.9083	0.9942	0.9513	0.9207
4	0.8361	0.9983	0.9172	0.8968
5	0.8583	0.9818	0.9251	0.8799



```
DP00064|P03607|#1-64:
SEQ     MSGLFHHRTKPREIRAFVMATRLTKKQLAQAIQNTLPNPPRRKRRAKRRAAQVPKPTQAGVSMAPIAQGTMVKLRPPMLRSSMDVTILSHCELSTELAVT
P(D)    9999988888888888888888888888888888888899999999999999999999988887776655544444444444444444321000000000000000
Lab     DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD000000000000000000
SEQ     VTIVVTSELVMPFTVGTWLRGVAQNWSKYAWVAIRYTYLPSCPTTTSGAIHMGFQYDMADTLPVSVNQLSNLKGYVVTGPVWEGQSGLCFVNNTKCPDTSR
P(D)    0000000000000000000000000000000000000000000000000000000000000000000000000000000000001111111111111110
Lab     0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
SEQ     AITIALDTNEVSEKRYPFKTATDYATAVGVNANIGNILVPARLVTAMEGGSSKTAVNTGRLYASYTIRLIEPIAAALNL
P(D)    0000000000000000000000000011111000000000000000000000112110000000000000000001235566
Lab     000000000000000000000000000000000000000000000000000000000000000000000000000DDDDD
```

Figure 6: The prediction result of DP00064 processed by SPOT–Disorder2

Table 8: Performance comparison in DIS1556

Schemes	Sens	Spec	ACC	MCC
SPOT-Disorder2	0.8106	0.8642	0.8374	0.6637
KNN	0.8807	0.9742	0.9274	0.8589

matrix (*PSSM*) for each protein residue. The train set of SPOT-Disorder2 comes from the Protein Data Bank (*PDB*) and Database of Protein Disorder (DisProt) including 2615 train proteins. However, our scheme just used six input features and 1556 proteins. Thus, our scheme required lower computational complexity.

We take the DP00064 from DisProt as an example and get the prediction result as Fig. 7. At the same time, the standard prediction result are shown as Fig. 8.

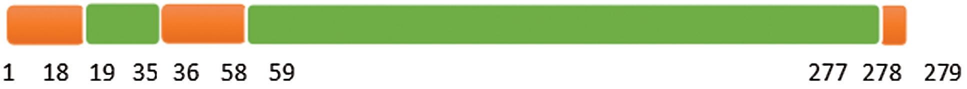


Figure 7: The prediction result of DP00064 based on our system (🟧 dis 🟩 ord)

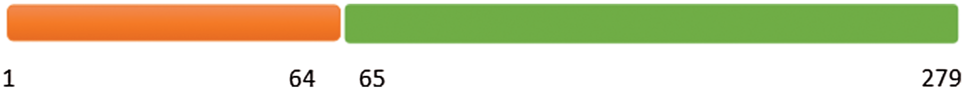


Figure 8: The prediction result of DP00064 based on DisPort (🟧 dis 🟩 ord)

6 Conclusion

In this study, we developed an effective scheme to predict the IDPs by employing sequence complexity and computing the Shannon entropy, topological entropy, sample entropy and three amino acid preferences. In addition, we used a sliding window to link nearby amino acids and improved the prediction accuracy greatly. Compared with the other competitive scheme SPOT–Disorder2, our scheme can get better *ACC* and *MCC*. The result showed that k-Nearest Neighbor was more appropriate and the overall prediction accuracy arrived to 92%. Furthermore, our method just uses six features and hence requires lower computational complexity. In the future, we will try to develop a complete software for use and we will try to explore deep learning algorithms to get better prediction of IDPs [36,37].

Acknowledgement: The authors are thankful for the support from Nankai University, School of Electronic Information and Optical Engineering.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Uversky, V. N. (2010). The mysterious unfoldome: structureless, underappreciated, yet vital part of any given proteome. *Journal of Biomedicine and Biotechnology, 2010(21),* 568068. DOI 10.1155/2010/568068.
2. Liu, Y., Wang, X., Liu, B. (2017). A comprehensive review and comparison of existing computational methods for intrinsically disordered protein and region prediction. *Briefings in Bioinformatics, 20(1),* 330–346. DOI 10.1093/bib/bbx126.
3. Dyson, H. J., Wright, P. E. (2005). Intrinsically unstructured proteins and their functions. *Nature Reviews Molecular Cell Biology, 6(3),* 197–208. DOI 10.1038/nrm1589.
4. Hegyi, H. (2009). Intrinsic structural disorder confers cellular viability on oncogenic fusion proteins. *PLoS Computational Biology, 5(10),* e1000552. DOI 10.1371/journal.pcbi.1000552.
5. Narayanan, R. L., Duerr, R. H. N., Bibow, R. (2010). Automatic assignment of the intrinsically disordered protein tau with 441-residues. *Journal of the American Chemical Society, 132(34),* 11906–11907. DOI 10.1021/ja105657f.
6. He, B., Wang, K., Liu, Y. (2009). Predicting intrinsic disorder in proteins: an overview. *Cell Research, 19(8),* 929–949.
7. Rune, L., Robert, R. B., Victor, N., Toby, G. J. (2003). GlobPlot: exploring protein sequences for globularity and disorder. *Nucleic Acids Research, 31(13),* 3701–3708. DOI 10.1093/nar/gkg519.
8. Dosztányi, Z. (2005). IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content. *Bioinformatics, 21(16),* 3433–3434. DOI 10.1093/bioinformatics/bti541.
9. Galzitskaya, O. V., Garbuzynskiy, S. O., Lobanov, M. Y. (2006). FoldUnfold: web server for the prediction of disordered regions in protein chain. *Bioinformatics, 22(23),* 2948–2949. DOI 10.1093/bioinformatics/btl504.
10. Lobanov, M. Y., Galzitskaya, O. V. (2011). The Ising model for prediction of disordered residues from protein sequence alone. *Physical Biology, 8(3),* 035004. DOI 10.1088/1478-3975/8/3/035004.
11. Yang, Z. R., Thomson, R., McNeil, P., Esnouf, R. M. (2005). RONN: the bio-basis function neural network technique applied to the detection of natively disordered regions in proteins. *Bioinformatics, 21(16),* 3369–3376. DOI 10.1093/bioinformatics/bti534.
12. Walsh, I., Martin, A. J. M., Di Domenico, T., Tosatto, S. C. E. (2012). ESpritz: accurate and fast prediction of protein disorder. *Bioinformatics, 28(4),* 503–509. DOI 10.1093/bioinformatics/btr682.
13. Jack, H., Yuedong, Y. (2017). Improving protein disorder prediction by deep bidirectional long short-term memory recurrent neural networks. *Bioinformatics, 33,* 685–694.
14. Wang, S., Ma, J., Xu, J. (2016). AUCpreD: proteome-level protein disorder prediction by AUC-maximized deep convolutional neural fields. *Bioinformatics, 32(17),* i672–i679. DOI 10.1093/bioinformatics/btw446.
15. Klausen, M. S., Jespersen, M. C., Nielsen, H. (2019). NetSurfP-2.0: improved prediction of protein structural features by integrated deep learning. *Proteins: Structure, Function, and Bioinformatics, 87(6),* 520–527. DOI 10.1002/prot.25674.

16. He, B., Zhang, W., Gao, H., Zhao, C., Feng, W. (2018). Predicting intrinsically disordered proteins based on different feature teams. *Bioinformatics and Computational Biology,* 19–22. DOI 10.1145/3194480.3194484.
17. Mishra, S., Rastogi, Y. P., Jabin, S. (2019). A deep learning ensemble for function prediction of hypothetical proteins from pathogenic bacterial species. *Computational Biology and Chemistry, 83,* 107–147. DOI 10.1016/j.compbiolchem.2019.107147.
18. Hanson, J., Paliwal, K. K., Litfin, T. (2020). SPOT-disorder2: improved protein intrinsic disorder prediction by ensembled deep learning. *Genomics Proteomics Bioinformatics, 17(6),* 645–656. DOI 10.1016/j.gpb.2019.01.004.
19. Jones, D. T., Ward, J. J. (2003). Prediction of disordered regions in proteins from position specific score matrices. *Proteins Structure Function and Bioinformatics, 53,* 573–578.
20. Liu, B., Wu, H., Zhang, D., Wang, X., Chou, K. C. (2017). Pse-analysis: a python package for DNA/RNA and protein/peptide sequence analysis based on pseudo components and kernel methods. *Oncotarget, 8,* 13338–13343. DOI 10.18632/oncotarget.14524.
21. Zhang, Z., Jun, J., Liu, B. (2017). PSFM-DBT: identifying DNA–binding proteins by combing position specific frequency matrix and distance-bigram transformation. *International Journal of Molecular Science, 18(9),* 1856.
22. Huang, Y. A., Chan, K. C. C., You, Z. H. (2018). Constructing prediction models from expression profiles for large scale lncRNA–miRNA interaction profiling. *Bioinformatics, 34(5),* 812–819. DOI 10.1093/bioinformatics/btx672.
23. Chen, J., Guo, M., Wang, X., Liu, B. (2017). A comprehensive review and comparison of different computational methods for protein remote homology detection. *Briefings in Bioinformatics, 19(2),* 231–244. DOI 10.1093/bib/bbw108.
24. Chen J., Guo M., Li S., Liu B., Hancock J. (2017). ProtDec-LTR2.0: an improved method for protein remote homology detection by combining pseudo protein and supervised learning to rank. *Bioinformatics, 33(21),* 3473–3476. DOI 10.1093/bioinformatics/btx429.
25. You, Z. H., Li, X., Chan, K. C. C. (2017). An improved sequence-based prediction protocol for protein-protein interactions using amino acids substitution matrix and rotation forest ensemble classifiers. *Neurocomputing, 228,* 277–282. DOI 10.1016/j.neucom.2016.10.042.
26. Wei, L., Ding, Y. J., Su, R., Tang, J. J., Zou, Q. (2018). Prediction of human protein subcellular localization using deep learning. *Journal of Parallel and Distributed Computing, 117,* 212–217. DOI 10.1016/j.jpdc.2017.08.009.
27. Pritianac, I., Vernon, R. M., Moses, A. M. (2019). Entropy and information within intrinsically disordered protein regions. *Entropy, 21(7),* 662. DOI 10.3390/e21070662.
28. He, H., Zhao, J. (2018). A low computational complexity scheme for the prediction of intrinsically disordered protein regions. *Mathematical Problems in Engineering, 2018,* 1–7. DOI 10.1155/2018/8087391.
29. Shuilin, J., Renjie, T., Qinghua, J. (2014). A generalized topological entropy for analyzing the complexity of DNA sequences. *PLoS One, 9(2),* e88519.
30. Koslicki, D. (2011). Topological entropy of DNA sequences. *Bioinformatics, 27(8),* 1061–1067. DOI 10.1093/bioinformatics/btr077.
31. Hao, H., Jiaxiang, Z., Guiling, S. (2019). The prediction of intrinsically disordered proteins based on feature selection. *Algorithms*. DOI 10.3390/a12020046.
32. Cheng, C., Sweredoski, M. J., Baldi, P. (2005). Accurate prediction of protein disordered regions by mining protein structure data. *Data Mining and Knowledge Discovery, 11(3),* 213–222. DOI 10.1007/s10618-005-0001-y.
33. Liu, B., Xu, J., Fan, S., Xu, R., Zhou, J. et al. (2015). PseDNA-Pro: DNA-binding protein identification by combining Chou's PseAAC and physicochemical distance transformation. *Molecular Informatics, 34(1),* 8–17. DOI 10.1002/minf.201400025.
34. Radivojac, P., Obradović, Z., Brown, C. J. (2003). Prediction of boundaries between intrinsically ordered and disordered protein regions. *Pacific Symposium on Biocomputing,* 216–227.
35. Radivojac, P., Iakoucheva, L. M., Oldfield, C. J. (2007). Intrinsic disorder and functional proteomics. *Biophysical Journal, 92(5),* 1439–1456. DOI 10.1529/biophysj.106.094045.
36. Necci, M., Piovesan, D., Dosztányi, Z., Tompa, P., Tosatto, S. C. E. (2018). A comprehensive assessment of long intrinsic protein disorder from the DisProt database. *Bioinformatic, 34(3),* 445–452. DOI 10.1093/bioinformatics/btx590.
37. Yumeng, L., Xiaolong, W., Bin, L. (2018). IDP-CRF: intrinsically disordered protein/region identification based on conditional random fields. *International Journal of Molecular Sciences, 19(9)*. DIO 10.3390/ijms19092483.