# Protein Disorder Prediction Using Machine Learning Techniques

**Badee Balto† and Amr Munshi††,**

† Laboratory and Blood Bank, King Abdullah Medical City, Saudi Arabia
†† Computer Engineering Department, Umm Al-Qura University, Saudi Arabia

**Summary**
This work attempts to develop a computationally fast protein disorder prediction model that has a high sensitivity and stable MCC (Matthews Correlation Coefficient) score, when compared to similar predictors. Further, this work focuses on these goals to ensure a very low number of false negative predictions by the presented model. However, with this focus on sensitivity, the model may produce an increased amount of false positive predictions. For that, it is important to monitor the MCC score and make sure to keep it relatively high as well. Accordingly, this confirms the efficiency of the presented model. The obtained results recommend the use of model developed for disordered protein prediction.
***Keywords:*** *Amino Acids, data mining, disordered proteins, machine learning, protein sequences*

## 1. Introduction

Understanding protein structures is essential for many fields of research including bioengineering and drug design. Protein sequences are composed of Amino Acids that are arranged in a linear order and joined together by a peptide bond. There are 20 unique types of Amino Acids that can be arranged in any order to form protein sequences [1]. The arrangement of these Amino Acids dictates many different attributes of the protein, including hydrophobicity, polarity and structure. These attributes of Amino Acids may vary by region of the protein sequence. This allows for each region to have either a fixed structure, such as helix, coil, or sheet, or no structure at all [2]. This structure determines the protein's biological function. In some cases, it is found that an entire protein sequence can have no structure and is referred to as being fully disordered. Predicting fully disordered proteins has become an area of research interest. For that, this work focuses on the prediction of fully disordered proteins using machine learning techniques.

This work attempts to develop a computationally fast protein disorder prediction model that has a high sensitivity and stable MCC (Matthews Correlation Coefficient) score [3], when compared to similar predictors. Further, this work focuses on these goals to ensure a very low number of false negative predictions by the presented model. However, with this focus on sensitivity, the model may produce an increased amount of false positive predictions. For that, it is important to monitor the MCC score and make sure to keep it relatively high as well. This will confirm the efficiency of the presented model.

The remained of the paper is structured as follows. Section 2, presents the general methodology to predict protein disorder, including the feature engineering stage. Further, the prediction system is presented in Section 3. The application on a real-world data to verify the results is presented in Section 4. The conclusions and remarks are drawn in Section 5.

## 2. Methodology and Feature Engineering

The Data Mining and Knowledge Discovery (DMKD) approach is adopted for the prediction of protein disorder problem. Figure 1 presents the general layout of the followed methodology. The data considered to illustrate the methodology includes 247 protein sequences, with 24 fully disordered proteins and 223 structured proteins.

For fully disordered protein prediction, seven kinds of features are collected which are calculated based on the Amino Acid sequence of each individual protein. These features can be categorized in structural and physicochemical features that were obtained from the PROFEAT website [4]. The structural and physicochemical features considered are:

1) Amino acid composition
2) Dipeptide composition
3) Autocorrelation descriptors
4) Composition, Transition and Distribution
5) Quasi-sequence-order Descriptors
6) Amphiphilic pseudo-amino acid composition
7) Total Amino Acid properties (TAAP)

Manuscript received March 5, 2022
Manuscript revised March 20, 2022
https://doi.org/10.22937/IJCSNS.2022.22.3.74

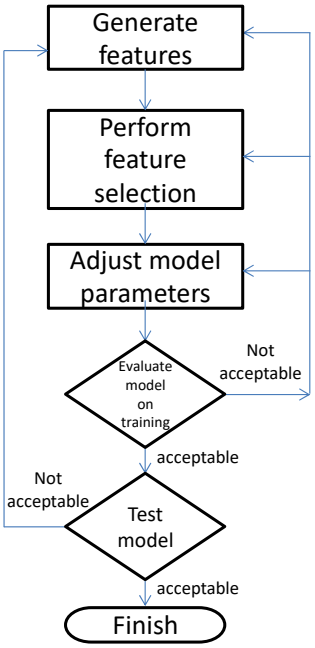


Fig. 1 The general layout of the followed methodology.

Using these features, 1080 numerical features are obtained. In the first process, combinations of these features are used, by selecting the top attributes selected among each of the categories listed above. Numerous feature selection methods are applied and different number of features are selected at each run. Although this process requires much time on working with these combinations of features, acceptable results were not obtained. However, the best results were obtained using the SVMAttribut evaluator and Ranker search method [5] and considering 5-fold cross validation and applying different data mining algorithms shown in Table 1.

It can be seen in Table 1 that the MCC of the models based on these features are not consistent. These unpleasant results encourage to alter the properties considered. For that, each individual property was analyzed. Consequently, it was observed that utilizing features based on Amino Acid Indices present the top results and are promising to utilize. In the listed properties the seventh feature "Total Amino Acid properties (TAAP)" uses the Amino Acid Indices to compute these particular features of Amino Acid sequences [6]. From this, 484 features were obtained using Amino Acid Indices, however, this is considered a large number of features and needs to be decreased to a lesser number.

Several feature selection methods such as Gain Ratio, Info Gain, Costly Gain Ratio and Costly info Gain, which are based on competing the entropy, were run to choose the best features among the 484 features.

In order to determine how many features should be selected from the 484 features, a search was conducted and the MCC scores for training and test datasets compared. The initial search was performed for number of features in (10, 15, 20, 25, 30) and then the search was further refined to add granularity around 20 ±4. The search resulted in the following results, shown in Figure 2.

These results show that the initial decision to select 20 features, which was chosen somewhat arbitrarily, is actually a solid choice given the input feature-set and provides solid training and test MCCs without overfitting. The final 20 features indices from AAIndex1 are as follows (ordered by their rank):

1- Feature 67
2- Feature 191
3- Feature 211
4- Feature 279
5- Feature 128
6- Feature 192
7- Feature 241
8- Feature 354
9- Feature 149

Table 1: Initial results during the data preparation step in the DMKD process

| Total # features | Feature selection method | # selected features | Dataset | Naïve Bayes | Logistic | Simple logistic | SMO | Threshold selector |
|---|---|---|---|---|---|---|---|---|
| 1080 | SVMAttributeEval Ranker Search | 50 | Training | 0.589 | 0.705 | 0.558 | 0.718 | 0.726 |
| | | | Test | 0.116 | 0.154 | 0.144 | 0.144 | 0.105 |
| 1080 | SVMAttributeEval Ranker Search | 35 | Training | 0.608 | 0.695 | 0.623 | 0.800 | 0.660 |
| | | | Test | 0.125 | 0.235 | 0.269 | 0.217 | 0.230 |
| 2800 | SVMAttributeEval Ranker Search | 50 | Training | 0.615 | 0.569 | 0.612 | 0.749 | 0.424 |
| | | | Test | 0.288 | 0.136 | 0.136 | 0.122 | 0.156 |

10- Feature 193
11- Feature 242
12- Feature 393
13- Feature 170
14- Feature 194
15- Feature 248

16- Feature 434
17- Feature 184
18- Feature 210
19- Feature 278
20- Feature 477

## 3. Prediction System

Numerous classifiers for such problems exist, due to the effectiveness of SVM [7] in many applications, the SVM is utilized to differentiate fully disordered proteins. The protein disorder prediction system is shown in Figure 3. Also, choosing the SVM as having a high likelihood of being a successful classifier is for the following reasons:

1- Support Vector Machines were used in 28% of the 18 competing classification methods;
2- SVM provide very good accuracy particularly against continuous features such as ours;
3- SVM are simple and will provide an optimal classifier for the chosen features once the parameters are tuned;
4- The procedure of [8] was adopted to achieve solid SVM classification.

A key step to classification with SVM is to scale the input data in order to keep features with higher overall numeric values from "swamping" those features with lower overall numeric values. For that, all features are scaled to the range [0,1]. Also, it is also important to keep this vector of scaling constants consistent between the training and test datasets. This will result in the training data set being scaled to the range [0,1], but the test datasets to slightly lower or higher values if the test data feature values fall outside the range for the same feature in the training dataset. In order to satisfy this, the LibSVMsvm-scale tool was utilized.

The second step is to choose the Radial Basis Function kernel for LibSVM [9]. This is a good first-step kernel because it allows for features to be mapped into a nonlinear feature space (but is not required), and can approximate the sigmoid kernel through different hyperparameter selection.

The RBF kernel takes two input hyperparameters: C and γ. Adequate values of these hyperparameters cannot usually be predetermined and generally a grid search method is used to test various values of both and observe classification results. It is important to use cross-validation in this step, in order to avoid hyperparameter selection which overfits the training data. It is generally suggested that good ranges for C and γ are as follows: C in (2-5, 2-3, … 215) and γ in (2-15, 2-13, … 25). Operating against our selected training data, with 5-fold cross validation, and plotting the resulting MCC, the obtained surface is given in Fig.

One can see from Fig. that the choice of C and γ has a huge effect on the outcome. Furthermore, some refinement around the ridge of peak MCCs may reveal a location of a global maximum MCC.

After four rounds of hyperparameter refinement, the 5-fold cross training MCC was maximized (and no longer improving) at a value of 0.307.

This was associated with:

$$C = 2^{2.732} = 6.643760193,$$
$$\gamma = 2^{-8.286} = 0.003203800$$

With these parameters the model was re-trained over the entire training dataset (no cross-validation) and marked as the final model. The final model was subsequently applied to a properly-scaled version of the test dataset. This model produced an MCC of 0.328 on the test dataset.

## 4. Experimental Results

The PROFEAT website [4] was utilized to generate attributes from the Amino Acid Indices, 484 features were obtained, which is considered to be a large number of features. After applying the feature selection algorithms (Table 2) on the Amino Acid Indices features and choosing the 20 best ranked features. Several classifiers were applied,

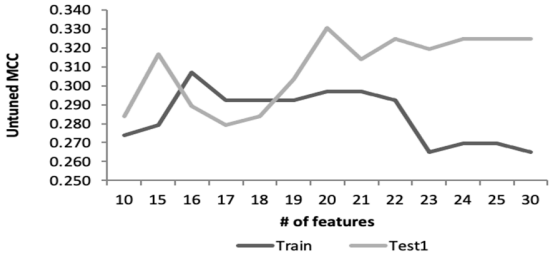


Fig. 2 Search for optimal number of features to select.

Table 2: Results for the feature selection algorithms

| Base dataset | # of features selected | Feature select algorithm | Best unturned MCC | |
|---|---|---|---|---|
| 484 | 20 | GainRatio | 0.297 | 0.330 |
| 484 | 20 | InfoGain | 0.297 | 0.322 |
| 484 | 20 | Costly GainRatio | 0.283 | 0.302 |
| 484 | 20 | Costly InfoGain | 0.256 | 0.304 |

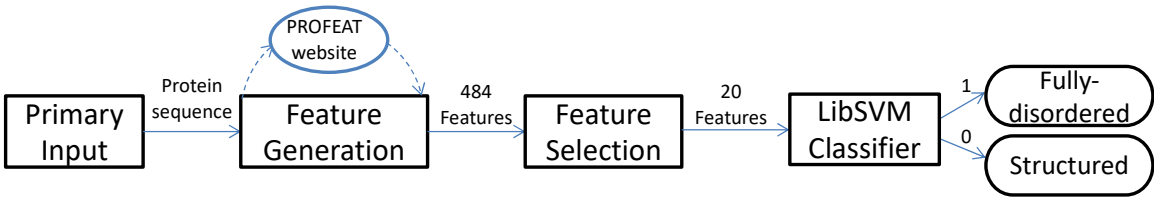


Fig. 3 The protein disorder prediction system.

Table 3: Results obtained by the presented model and other alternatives

| Predictor | Training dataset | | | | | | | | Test dataset | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | mcc | acc | sens | spec | TP | FP | TN | FN | mcc | acc | sens | spec | TP | FP | TN | FN |
| CSpritzLONG | 0.413 | 91.1 | 37.5 | 96.9 | 9 | 7 | 216 | 15 | 0.575 | 92.7 | 42.9 | 99.1 | 12 | 2 | 217 | 16 |
| CSpritzSHORT | 0.147 | 89.9 | 8.3 | 98.7 | 2 | 3 | 220 | 22 | 0.310 | 89.9 | 10.7 | 100.0 | 3 | 0 | 219 | 25 |
| DISOPRED | -0.030 | 89.5 | 0.0 | 99.1 | 0 | 2 | 221 | 24 | 0.253 | 89.5 | 7.1 | 100.0 | 2 | 0 | 219 | 26 |
| Disprot&FPR_Espritz | 0.388 | 89.1 | 45.8 | 93.7 | 11 | 14 | 209 | 13 | 0.425 | 90.3 | 35.7 | 97.3 | 10 | 6 | 213 | 18 |
| Disprot&SW_Espritz | 0.510 | 87.0 | 79.2 | 87.9 | 19 | 27 | 196 | 5 | 0.620 | 90.3 | 82.1 | 91.3 | 23 | 19 | 200 | 5 |
| IUPredLONG | 0.213 | 90.7 | 8.3 | 99.6 | 2 | 1 | 222 | 22 | 0.253 | 89.5 | 7.1 | 100.0 | 2 | 0 | 219 | 26 |
| MD | 0.374 | 90.7 | 33.3 | 96.9 | 8 | 7 | 216 | 16 | 0.440 | 91.1 | 25.0 | 99.5 | 7 | 1 | 218 | 21 |
| MFDp | 0.434 | 91.1 | 41.7 | 96.4 | 10 | 8 | 215 | 14 | 0.476 | 91.5 | 28.6 | 99.5 | 8 | 1 | 218 | 20 |
| PONDR-FIT | 0.174 | 90.3 | 8.3 | 99.1 | 2 | 2 | 221 | 22 | 0.359 | 90.3 | 14.3 | 100.0 | 4 | 0 | 219 | 24 |
| Ucon | 0.194 | 90.7 | 4.2 | 100.0 | 1 | 0 | 223 | 23 | 0.253 | 89.5 | 7.1 | 100.0 | 2 | 0 | 219 | 26 |
| x-ray&FPR_Espritz | 0.174 | 90.3 | 8.3 | 99.1 | 2 | 2 | 221 | 22 | 0.359 | 90.3 | 14.3 | 100.0 | 4 | 0 | 219 | 24 |
| x-ray&SW_Espritz | 0.303 | 91.1 | 16.7 | 99.1 | 4 | 2 | 221 | 20 | 0.441 | 91.1 | 21.4 | 100.0 | 6 | 0 | 219 | 22 |
| Min | -0.03 | | | | | | | | 0.253 | | | | | | | |
| Avg. | 0.275 | 90.13 | 24.3 | 97.21 | 5.8 | 6.3 | 216 | 18 | 0.397 | 90.5 | 24.6 | 98.89 | 6.9 | 2.4 | 216 | 21 |
| Max | 0.510 | 91.10 | 79.2 | 100.0 | 19 | 27 | 223 | 24 | 0.620 | 92.7 | 82.1 | 100.0 | 23 | 19 | 219 | 26 |
| Presented Model | 0.307 | 79.4 | 62.5 | 81.2 | 15 | 42 | 181 | **9** | 0.328 | 81.4 | 57.1 | 84.5 | 16 | 34 | 185 | **12** |
| Percentile | 55% | 0% | 95% | 0% | 95% | 0% | 0% | 96% | 31% | 0% | 94% | 0% | 94% | 0% | 0% | 6% |

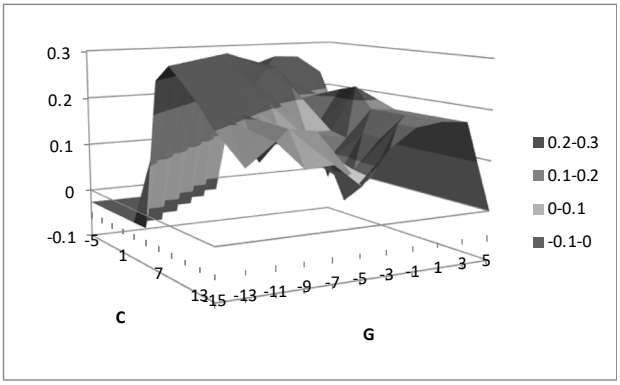


Fig. 4 Training MCC for selecting optimal parameters C and G for LibSVM

such as SMO, J4.5 and BayesNet using those features and managed sometimes to gain high MCC scores for the training set, but this was not the case for the test set. It was observed that there was a significant drop of MCC score for the test set. In that case, the probability of the randomness of the selected features and the capabilities of the selected classifier was considered. For that, it was essential to analyze the features and it was assumed that they were efficient as they were ranked as the first 20. Also, it was noticed that the features were continuous values. As the values of the 20 selected features were continuous, it was appropriate to use a classifier that is capable of handling continuous values. For that, the SVM classifier was chosen, which is efficient handling continuous values, to build the prediction model. It can be noticed, the stability between the values of the MCC for the training set and the test set indicates that the selected features were not random. Moreover, the MCCs for the test sets were better than the train sets. This encouraged to continue the process with those features.

Table 3 presents the results obtained using the 20 features selected and LibSVM algorithm with the optimal parameters obtained; and the results given by alternative models already existing. Also, it summarizes the results of applying the SVM classifier on the 20 Amino Acids Indices features. The number of selected features was considered to be a reasonable number and could predict sequences of Amino Acids rapidly, which proves that model is efficient. The achieved MCC after using the SVM classifier was

0.307 for the training set and 0.328 for the test set. According to the achieved MCC results, consistency is evident with a ±0.015 stability ratio. Also, achieving a significant consistent sensitivity rate, which we successfully obtained and was 95% on the training set and 94% on the test set was considered.

## 5. Conclusions

The work presented in this paper adopted a systematic DMKD process for the prediction of protein disorders. During the process, it was necessary to go back to previous steps of the DMKD process several times, to reach to the desired goal. It was necessary to repeat data preparation, data mining and analysis steps iteratively throughout the work. Following the procedure led to achieve substantially adequate results. The final accepted model achieved most of the objectives of the work. The presented model was capable of classifying fully disordered proteins from the structured proteins. MCC values for training and test sets are reasonably high and consistent. It should be noted here that MCC for training and test data were close which describes the stability of the model. As MCC does not vary for training and test data sets, model is expected to perform equally efficiently on any other data set or in other words the model is highly reliable.

Though the MCC values obtained by the presented model were not relatively high, however, they are considered reasonably high. One important point to note here is that, the model is able to predict disordered proteins and structured proteins in almost equally efficient way (specificity and sensitivity values are high). The obtained results recommend the use of model developed for disordered protein prediction.

## References

[1] J. E. Hall, Guyton and Hall Textbook of Medical Physiology. 13th edition. Philadelphia, PA: Elsevier, 2016.

[2] A. György, and J. A. Marsh, “Alpha helices are more robust to mutations than beta strands,” PLoS computational biology, vol. 12, no. 12, 2016.

[3] D. Chicco, M. J. Warrens, and G. Jurman, “The matthews correlation coefficient (MCC) is more informative than cohen’s kappa and brier score in binary classification assessment,” IEEE Access, vol. 9, pp. 78368-78381, 2021.

[4] P. Zhang, L. Tao, X. Zeng, C. Qin, S. Y. Chen, F. Zhu, S. Y. Yang, Z. R. Li, W.P. Chen, and Y. Z. Chen. “PROFEAT update: A protein features web server with added facility to compute network descriptors for studying omics-derived networks,” J Mol Biol. 2016.

[5] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software: An update. SIGKDD Explorations, vol. 11, no. 1, 2009.

[6] D. Dwyer, Amino Acids: Chemical Properties. 2008.

[7] N. Cristianini, and E. Ricci, Support Vector Machines. In: Kao MY. Encyclopedia of Algorithms. Springer, Boston, MA. 2008. https://doi.org/10.1007/978-0-387-30162-4_415

[8] C. W. Hsu, C. C. Chang, and C. J. Lin, A practical guide to support vector classification. Department of Computer Science, National Taiwan University, 2003. Retrieved from www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

[9] X. Ding, J. Liu, F. Yang, and J. Cao, “Random radial basis function kernel-based support vector machine,” Journal of the Franklin Institute, vol. 358, no. 18, pp. 10121-10140, 2021.