Replicating DeepPPI Model for Boosting Prediction of ProteinProtein Interactions with Deep Neural Networks

Asma Hakouz[1]

***Abstract*— This project is a refactoring of DeepPPI model using Julia programming language**

I. INTRODUCTION

On one hand, there are a lot of experimental methods for the detection of protein protein interactions (PPI) such as x-ray crystallography, Nuclear Magnetic Resonance NMR, and Tandem Affinity Purification. However, these methods have many downsides, such as being costly, time consuming, labor-intensive, and highly affected by equipments resolution and environmental disturbances during the experiment. Thus, the need for a powerful computational methods to predict PPI is rising. Du et al. proposes using a Deep Neural Network model to predict PPI using protein descriptors extracted from empirical data stored in multiple databases such as Database of Interacting Proteins (DIP). The novelty of using DDN, based on the paper is that it can automatically extract high-level meaningful and abstract features of proteins from noisy data instead of hand-picking and crafting discriminant features which in addition to requiring a solid domain knowledge, might also be prone to errors due to the noise that might be present in the features. Based on the paper, applying the proposed DNN model achieved the results shown in table I.

TABLE I: DeepPPI model average results

Accuracy	Precision	Recall	Specificity	MCC
92.5%	94.38%	90.56%	94.49%	85.08%

DeepPPI Predictor performance was evaluated using eight different PPI datasets taken from literature, they are described in details in the paper.

II. DATASETS

The number of features used in datasets is 1164 divided as detailed in table II.

TABLE II: DeepPPI features vector / descriptor components

Total number of features	1164
Amino Acid Composition	20
Dipeptide Composition	400
Composition	72
Transition	72
Distribution	360
Quasi- Sequence- Order	160
Amphiphilic Pseudoamino Acid Composition	80

More details about how to calculate each individual feature are provided in the original paper and are out of the scope of this project.

A. Training dataset

- **Positive training set** was taken from *Saccharomyces cerevisiae* PPIs data set which can be downloaded from the Database of Interacting Proteins (DIP; version 20160731)[2]
  - Original number of samples: 22975 protein pairs.
  - Filtering and preprocessing: every pair that has a protein with less than 50 amino acids in its chain was eliminated. Then, after applying cluster analysis using CD-HIT program [3] , pairs with high sequence identity (i.e., having similar amino acid sequences) were clustered and a non-redundant subset was chosen from the clustered data resulting in an overall sequence identity level of 40%.
  - Final number of samples after filtering and preprocessing: 17257 pairs.
- **Negative training set** was generated based on proteins cellular localization information. Where each pair of proteins in the negative set was picked so that one of the protein is localized in one part of the cell while the other is localized in a different part. Thus, ensuring that this pair of protein should not have an interaction. The cellular localization information was taken from Swiss-Prot [4] database. A total of 48594 pairs were generated using this approach.

B. Testing datasets

In the first step, as described in the original paper, 8 different datasets were used for testing and evaluation and comparison with prediction methods that are not knowledge-based.

- The first dataset was collected by You et al. [11] from the S. cerevisiae core subset in DIP.
  Number of instances:
  Total: 11188 pairs
  Positive: 5943 pairs
  Negative: 5245 pairs

- The second dataset is Helicobacter pylori protein pairs described by Martin et al.[12]
  Number of instances:
  Total: 2916 pairs
  Positive: 1458 pairs
  Negative: 1458 pairs

[1]A. Hakouz, Computer Science and Engineering, koc University ahakouz17@ku.edu.tr

- The third dataset was collected from Human Protein Reference Database (HPRD) as described by Huang et al.[13]
  Number of instances:
  Total: 8161 pairs
  Positive: 3899 pairs
  Negative: 4262 pairs

- The last five datasets were chosen as a species-specific PPI data. As used and described by Zhou et al.[14]
  Number of instances:
  Caenorhabditis elegans: 4013 interacting pairs
  Escherichia coli: 6954 interacting pairs
  Homo sapiens: 1412 interacting pairs
  Mus musculus: 313 interacting pairs
  H. pylori: 1420 interacting pairs

In the second step, 3 different datasets were used for testing and comparison with knowledge-based prediction methods. Given that the same features were used in both approaches. The source of all data sets was from Saha et al.[6]

- Silver dataset
  Number of instances:
  14677 Yeast and 27419 Human interacting proteins

- Gold dataset
  Number of instances:
  2117 Yeast and 1582 Human interacting proteins

- All interaction dataset which contains Human and Yeast PPIs that have been confirmed using at least one experimental method
  Number of instances:
  190377 Yeast and 57576 Human interacting proteins

III. Model Training

A. Training Data Preprocessing

In order to train the model, a training dataset was prepared with a 1:1 positive to negative ratio. The original dataset had a 17257:48594 positive to negative ratio. Therefore to ensure an equal ratio, 17257 negative samples were randomly picked out of all negative samples, then it was concatenated to the positive samples, shuffled and mapped to the labels and protein features to prepare it as a training dataset. Finally, the dataset was divided into three parts: trn/dev/tst based on the ratios shown in figure 1.
As shown in the figure, the TRAIN SET is divided into training and validation sets with 77%, & 23% respectively. These sets are used during training the model with different hyper parameters and different optimization methods, where the model with the highest performance on the validation set was selected. The following metrics were calculated and taken into consideration while training the model:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

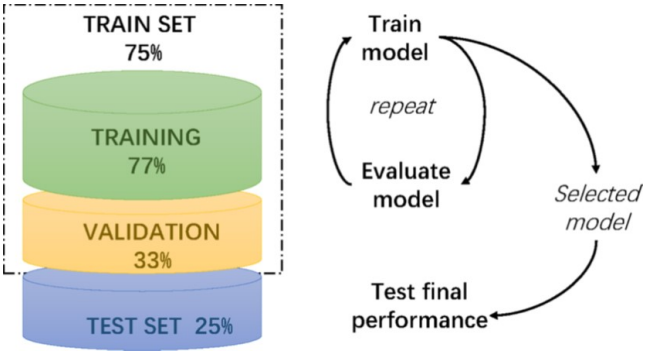

Fig. 1: Holdout validation [1]

$$Recall/Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

$$Precision = \frac{TP}{TP + FP}$$

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

$$Precision = \frac{2TP}{2TP + FP + FN}$$

Where: TP: True Positive predictions FP: False Positive predictions TN: True Negative predictions FN: False Negative predictions MCC: Matthews correlation coefficient, which is a quality measure used mainly in the case of binary classification. And it takes into consideration true positive, true negative, false positive and false negative values. F1: is a statistical quality measure used in statistical analysis of binary classification and its value is between 0 and 1 and indicates how well was the performance of the predictor.

B. DeepPPI Models

In the original DeepPPI paper two suggested models with different architectures were trained to predict PPIs and their performances were compared and analyzed. In my project I re-emplemented both models using Julia [15].

- ***DeepPPI-Con***
  DeepPPI-Con is a multilayer perceptron. The input to this model is a concatenated vector of the feature vectors of the protein pair in each sample. The model has only one neural network with four hidden layers with the dimensions 512, 256, 128, 128. The output of the last hidden layer passes through a softmax layer with 2 output, which represents a one-hot encoding label. Therefore, a ”01” output means that the input pair are not predicted to interact while a ”10” output predicts an interaction between the input protein pair.
  Total number of trainable parameters: 777,474
- ***DeepPPI-Sep***
  At the initial phase of DeepPPI-Sep there are two seperate neural networks where the feature vector of both partners from the input protein pair is input to

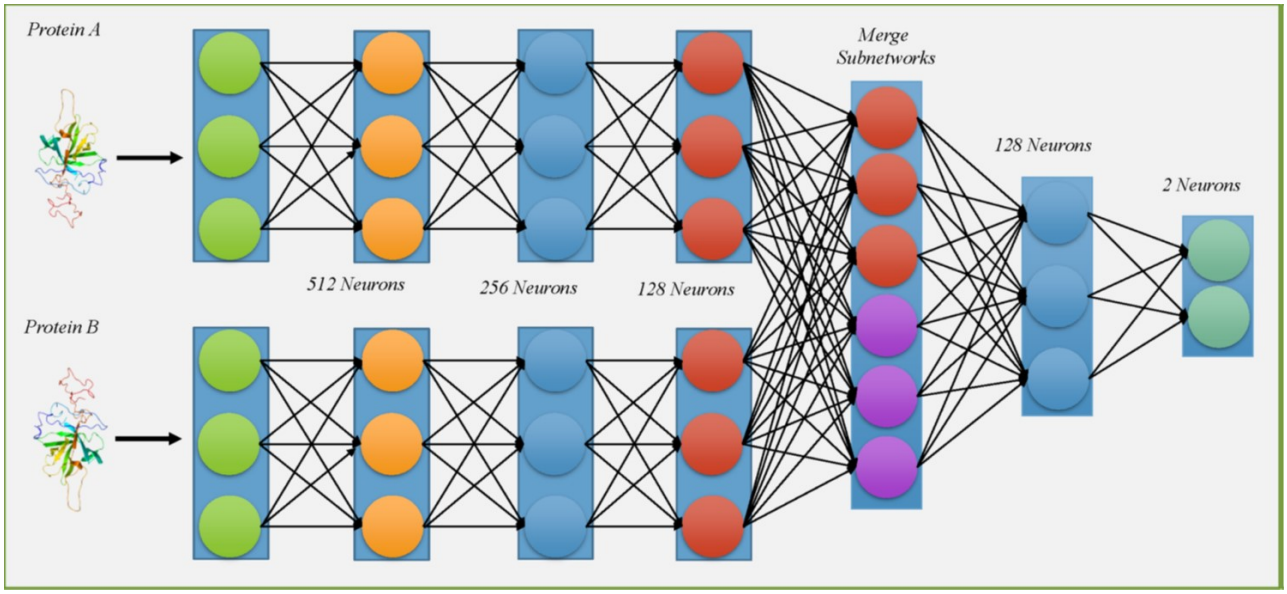

Fig. 2: DeepPPI-Sep Model Architecture [1]

one of the separate network. Both these networks has 3 hidden layers with 512, 256, 128 hidden units. The outputs of these two networks are concatenated to form the input of a third neural network with one hidden layer containing 128 hidden units. The output layer is the same as in DeepPPI-Con model. The model architecture is shown in figure 2

Total number of trainable parameters: 1,554,562

C. Base Model Analysis

before training both models, a baseline performance was analyzed where the performance of a randomly initialized model was analyzed for both DeepPPI-Con and DeepPPI-Sep. Baseline model results were as in table III.

TABLE III: Error and Loss for DeepPPI-Sep and Con architectures

DeepPPI-Con	loss	error
Train set	0.69301564	0.5609971509971509
Dev set	0.69300413	0.565680693069307
Test set	0.69298553	0.5705928853754939
DeepPPI-Sep	**loss**	**error**
Train set	0.69301564	0.49954472843450476
Dev set	0.69300413	0.49435541310541314
Test set	0.69314665	0.5093719120553359

D. Training the Models

Both DeepPPI-Con and DeepPPI-Sep models were trained using several hyper parameter combinations to optimize the perormance of the PPI predictor. After picking the model with the highest performance, to evaluate PPI predictor performance, the S. cerevisiae data set was used, where Five different subdatasets were constructed using the previously mentioned method for randomly picking negative samples to have a 1:1 positive to negative ratio in the dataset. Comparison between the results of the trained model with the results of the model from the original paper are presented in the table 4 and table 5.

E. Improvements on Model

After the analysis of the trained model, I noticed a consistent high False Positive rate. Thus, after further investigation and research of what might have caused the problem, I came with of the highly probable reasons which is the criteria of constructing the negative training dataset. As I mentioned before, in the paper I am replicating, they randomly selected pairs of proteins having different cellular localization, which assumes that proteins in different cell locations would not interact. However, this assumption can be inaccurate due to proteins re-location and the existence of interactions between cellular compartments through organelle membrane. Due to these facts, using the suggested dataset might have to led to a high FP rate. To solve this issue, I used the Negatome 2.0 dataset [16], which is a manually curated set of non-interacting protein pairs derived from literature and proteins' 3D structures stored in the Protein Data Bank (PDB) [17]. Therefore, Negatome dataset contains only experimentally proven non-interacting protein pairs.

- ***Negatome set description***

  As shown in figure 3, Negatome dataset was constructed using two complementary efforts:

  1) Manual curation of literature.
  2) Analyzing protein complexes with known 3D structure in PDB.

  Negatome contains nine different datasets based on the method used to extract the data or the type of the data that it contains. See figure 4 The dataset contains a list

Table IV: DeepPPI-Sep						
My DeepPPI-Sep Model results						
Dataset	Accuracy	Precision	npv	Recall	Specificity	MCC
dataset1	0.938810986	0.918725468	0.959050721	0.957641396	0.921323201	0.878370192
dataset2	0.932553019	0.912661738	0.95256917	0.950890708	0.915530726	0.865825966
dataset3	0.937536215	0.922899354	0.952292297	0.951225315	0.924536828	0.87547685
dataset4	0.931857689	0.920881671	0.942810836	0.941413662	0.922728303	0.863917206
dataset5	0.940317534	0.915984086	0.964187328	0.961670762	0.921254661	0.881547343
Average	0.936215089	0.918230463	0.95418207	0.952568369	0.921074744	0.873027512
Original DeepPPI-Sep Model results						
Dataset	Accuracy	Precision	npv	Recall	Specificity	MCC
dataset1	0.924440839	0.943369175	0.906615661	0.904881962	0.944444444	0.849655558
dataset2	0.923629621	0.941162458	0.907018732	0.905569562	0.942100328	0.847925502
dataset3	0.925020281	0.944497607	0.906720611	0.904881962	0.945616502	0.850858266
dataset4	0.921775409	0.940726577	0.903937008	0.902131561	0.941865916	0.844330466
dataset5	0.925136169	0.945363048	0.906193896	0.904194362	0.946554149	0.851152631
Average	0.924000464	0.943023773	0.906097182	0.904331882	0.944116268	0.848784485
Performance Difference	+1.22%	-2.48%	+4.81%	+4.82%	-2.34%	+2.42%

Table V: DeepPPI-Con						
My DeepPPI-Con Model results						
Dataset	Accuracy	Precision	npv	Recall	Specificity	MCC
dataset1	0.938810986	0.918725468	0.959050721	0.957641396	0.921323201	0.878370192
dataset2	0.932553019	0.912661738	0.95256917	0.950890708	0.915530726	0.865825966
dataset3	0.937536215	0.922899354	0.952292297	0.951225315	0.924536828	0.87547685
dataset4	0.931857689	0.920881671	0.942810836	0.941413662	0.922728303	0.863917206
dataset5	0.940317534	0.915984086	0.964187328	0.961670762	0.921254661	0.881547343
Average	0.936215089	0.918230463	0.95418207	0.952568369	0.921074744	0.873027512
Original DeepPPI-Con Model results						
Dataset	Accuracy	Precision	npv	Recall	Specificity	MCC
dataset1	0.901958512	0.915819342	0.888636363	0.887691955	0.916549461	0.804348554
dataset2	0.898365975	0.914407988	0.88311101	0.881503552	0.915611814	0.797317158
dataset3	0.901726735	0.906734552	0.896703807	0.898005959	0.905532114	0.803488215
dataset4	0.902422065	0.913748531	0.891403749	0.891129956	0.913970933	0.805126585
dataset5	0.899988411	0.910028116	0.890162807	0.890213156	0.909985935	0.800195007
Average	0.90089234	0.912147706	0.890003547	0.889708916	0.912330051	0.802095104
Performance Difference	0.74%	-1.18%	2.60%	2.41%	-0.90%	1.47%

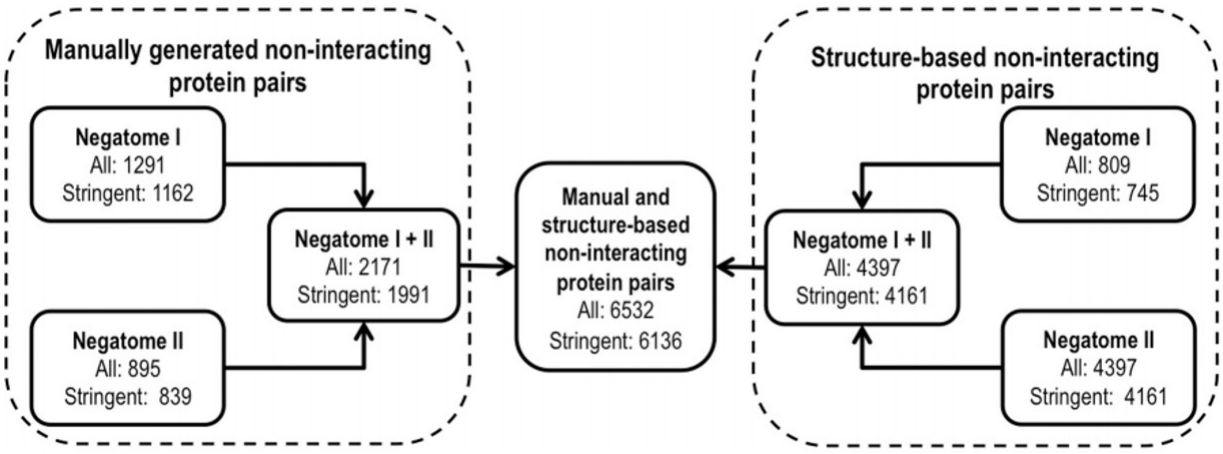

Fig. 3: Negatome dataset structure [1]

of protein pairs represented by their UniProt IDs.

- ***Constructing Negatome set***
  For training my model, I used the Negatome-Combine dataset. Since the set only contains UniProt IDs for the protein pairs, I had to manually extract the features of the proteins from Protein Feature Server(PROFEAT) webserver [18]. In order to achieve this task, I followed the procedure below:

1) I downloaded Negatome-Combine dataset from the server as a txt file.
2) I wrote a Python script to automate reading the file, accessing the PROFEAT webserver, filling the required information, and extracting the features.
3) I constructed a new file, *Negatome_features.csv*, which contains a mapping between the UniProt ID of the protein and its features. This file was used later in training to construct the negative samples

Overview of the Negatome datasets

	Dataset name	Derived from	Description	# of pairs
Structurally analyzed NIPs	PDB	The PDB database	Protein pairs that are members of at least one structural complex but do not interact directly. Organism of origin is not restricted.	4397
	PDB-stringent	PDB	The PDB dataset filtered against the IntAct dataset.	4161
Manually curated NIPs	Manual	Manual literature annotation	Manually annotated literature data describing the lack of protein interaction. High-throughput data are not included. The data is restricted only to mammalian proteins.	2171
	Manual-stringent	Manual	The Manual dataset filtered against the IntAct dataset.	1991
	Combined	Combines both PDB and Manual datasets		6532
	Combined-stringent	Combines both PDB-stringent and Manual-stringent datasets		6136

Fig. 4: Negatome different dataset structure [1]

My DeepPPI-Con Models Scores		
	Average	Diff. with paper
Accuracy	0.90826283	0.737083463
Precision	0.90033305	-1.181494626
npv	0.91602894	2.602493571
Recall	0.91381704	2.410803806
Specificity	0.90334553	-0.898447251
MCC	0.81676217	1.466716799

My DeepPPI-Con Models Scores with Negatome		
	Average	Diff. with paper
Accuracy	0.9666	6.5708%
Precision	0.95952	4.7372%
npv	0.973597	8.3593%
Recall	0.9731137	8.34047%
Specificity	0.960333	4.8003%
MCC	0.933283	13.1188%

Fig. 5: Comparison of DeepPPI-Con Model Scores

of the training data set.

- ***Results of Training with the Negatome set*** Figures 5 and Figure 6 c shows the comparison between the performance evaluation of the trained model suggested by the original paper, my first model, and my model after using the Negatome dataset as my negative set.

My DeepPPI-Sep Models Scores		
	Average	Diff. with paper
Accuracy	0.936215089	1.2214625
Precision	0.918230463	-2.479331
npv	0.95418207	4.8084888
Recall	0.952568369	4.8236487
Specificity	0.921074744	-2.3041524
MCC	0.873027512	2.4243027

My DeepPPI-Sep Scores with Negatome		
	Average	Diff. with paper
Accuracy	0.966	4.20%
Precision	0.962	1.9%
npv	0.971	6.49%
Recall	0.971	6.67%
Specificity	0.962	1.79%
MCC	0.933	8.42%

Fig. 6: Comparison of DeepPPI-Sep Model Scores

REFERENCES

[1] Xiuquan Du, Shiwei Sun, Changlin Hu, Yu Yao, Yuanting Yan, and Yanping Zhang, 2017, DeepPPI: Boosting Prediction of ProteinProtein Interactions with Deep Neural Networks, Journal of Chemical Information and Modeling 2017 57 (6), 1499-1510.

[2] Salwinski L, Miller CS, Smith AJ, Pettit FK, Bowie JU, Eisenberg D. (2004) The Database of Interacting Proteins: 2004 update. NAR 32:D449-51.

[3] Ying Huang, Beifang Niu, Ying Gao, Limin Fu and Weizhong Li. CD-HIT Suite: a web server for clustering and comparing biological sequences. Bioinformatics, (2010). 26:680

[4] Apweiler, R., Bairoch, A., Wu, C. H., et al. (2004) UniProt: the Universal Protein knowledgebase. Nucleic Acids Res. 432, D115D119.

[5] Wang, J.; Zhang, L.; Jia, L.; Ren, Y.; Yu, G. Protein-Protein Interactions Prediction Using a Novel Local Conjoint Triad Descriptor of Amino Acid Sequences. Int. J. Mol. Sci. 2017, 18, 2373. DOI: 10.3390/ijms18112373

[6] Saha, I.; Zubek, J.; Klingstrom, T.; Forsberg, S.; Wikander, J.; Kierczak, M.; Maulik, U.; Plewczynski, D. Ensemble Learning Prediction of Protein-Protein Interactions Using Proteins Functional Annotations. Mol. BioSyst. 2014,10, 820830. DOI:10.1039/c3mb70486f

[7] Enright, A. J., Iliopoulos, I., Kyrpides, N. C., & Ouzounis, C. A. (1999). Protein interaction maps for complete genomes based on gene fusion events. Nature, 402(6757), 86.

[8] Yamada, K. D., & Kinoshita, K. (2017). De novo profile generation based on sequence context specificity with the long short-term memory network.

[9] Yamada, K. D. (2018). Derivative-free neural network for optimizing the scoring functions associated with dynamic programming of pairwise-profile alignment. Algorithms for Molecular Biology, 13(1).

[10] Wong, L.; You, Z. H.; Li, S.; Huang, Y. a.; Liu, G. Detection of ProteinProtein Interactions from Amino Acid Sequences Using A Rotation Forest Model with a Novel PR-LPQ Descriptor; Springer International Publishing: Cham, Switzerland, 2015; pp 713720.

[11] You ZH, Zhu L, Zheng CH, Yu HJ, Deng SP. Prediction of protein-protein interactions from amino acid sequences using a novel multi-scale continuous and discontinuous feature set. Bmc Bioinforma.2014;15(15):S9 doi: 10.1186/1471-2105-15-S15-S9

[12] Martin, S.; Roe, D.; Faulon, J. L. Predicting Protein-Protein Interactions Using Signature Products. Bioinformatics 2005, 21, 218 226.

[13] Huang, Y. A.; You, Z. H.; Gao, X.; Wong, L.; Wang, L. Using Weighted Sparse Representation Model Combined with Discrete Cosine Transformation to Predict Protein-Protein Interactions from Protein Sequence. BioMed Res. Int. 2015, 2015, 110.

[14] Zhou, Y. Z.; Gao, Y.; Zheng, Y. Y. Prediction of Protein-Protein Interactions Using Local Description of Amino Acid Sequence. Communications in Computer and Information Science 2011, 202, 254262.

[15] Julia: A Fast Dynamic Language for Technical Computing. Jeff Bezanson, Stefan Karpinski, Viral B. Shah and Alan Edelman (2012) . arXiv: 1209.5145.

[16] Negatome 2.0: a database of non-interacting proteins derived by literature mining, manual annotation and protein structure analysis. Philipp Blohm,Goar Frishman,Pawel Smialowski,Florian Goebels, Benedikt Wachinger,Andreas Ruepp,andDmitrij Frishman

[17] RCSB Protein Data Bank: Sustaining a living digital data resource that enables breakthroughs in scientific research and biomedical education (2018) Protein Science 27: 316330 doi: 10.1002/pro.3331

[18] P. Zhang, L. Tao, X. Zeng, C. Qin, S.Y. Chen, F. Zhu, S.Y. Yang, Z.R. Li, W.P. Chen, 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. pii:S0022-2836(16)30428-4. (2016).