FUZZY PATTERN TREES FOR PREDICTING PROTEIN BINDING SITES

Georgina Mirceva
Faculty of computer science and engineering
Skopje, Macedonia

Andrea Kulakov
Faculty of computer science and engineering
Skopje, Macedonia

ABSTRACT

Protein molecules are very important in the living organisms, since they are involved in many processes in the organisms. The knowledge of their functions is crucial for designing new drugs. There are various experimental methods for determining their functions, but they are very complex, so the number of known protein structures with undetermined functions is growing too fast. Therefore, one of the main research directions in bioinformatics community is investigating new computational methods for determining the protein functions. In this research paper, we present a two-step fuzzy pattern tree based method for predicting the binding sites of the proteins. Further, this method could be incorporated in a framework for protein function annotation. The binding sites of the proteins are the amino acid residues where interactions between protein structures occur, while their features determine the functions that the proteins have in these interactions. In the first step of our method, we extract the most important features of the amino acids of the protein molecules. In the second step, using the amino acids' features we induce fuzzy pattern trees that would be used to classify the amino acids as binding or non-binding sites. We present some experimental results of the evaluation of the fuzzy pattern trees based method.

I. INTRODUCTION

Protein molecules are involved in many processes in the living organisms, so they are very important compounds in the organisms. The knowledge of protein functions is essential for designing new drugs, better crops and synthetic biochemical. There are many experimental methods for determining the protein functions. However, these methods are very expensive and too complex. As a consequence of this, there are many protein molecules with known structures that are not functionally annotated yet. Thus, there is an evident need of development of computational methods for determining protein functions.

In the literature, there are various methods for annotating protein structures, and they consider different information about the protein molecules. One group of methods examines the structural and sequence homology of the protein molecules [1]. Nevertheless, these methods are able to discover only a global similarity of the protein structures, while proteins with similar local similarity and even dissimilar global similarity could still share common functions. Other group of methods [2] annotates protein molecules by analysing the protein-protein interaction networks. However, these methods require a priori knowledge about the proteins that interact with a given query protein, whereas the acquisition of this knowledge is very expensive. Third group of methods examines the conservation of the proteins' sequences and structures [3], which could be determined by multiple-alignment of the proteins' sequences and structures. Fourth group of methods annotates the protein structures by examining the features of the protein binding sites [4]. Protein binding sites are the amino acids where interactions between the proteins occur. Molecular biologists manually annotate the protein structures in similar manner, so this group of methods is the most relevant one. For that reason, in this research paper we focus on predicting the protein binding sites.

The protein binding sites could be predicted by considering different features of the amino acids, like Accessible Surface Area (ASA) [5], depth index (DPX) [6], protrusion index (CX) [7], hydrophobicity [8], and other physico-chemical and geometrical features. There is no single feature that completely distinguishes the protein binding and non-binding sites, so in the protein binding sites prediction several features should be considered.

There are various methods for predicting the protein binding sites [9, 10, 11, 12, 13]. However, they are sensitive to small changes in the data (the amino acids' features in this case) obtained due to the protein evolution. To overcome this, we use the fuzzy theory and induce fuzzy pattern trees for predicting the protein binding sites. Inspired by the classical decision trees, there are several studies on fuzzy decision trees (FDTs). In [14, 15], different methods for inducing FDTs are presented, while [16] and [17] present some optimizations of FDTs. The fuzzy decision trees are widely studied in [18], and their advantages and disadvantages over classical decision trees are presented. However, FDTs use only a single fuzzy aggregation operator, while in fuzzy pattern trees (FPTs) [19] induction different fuzzy aggregation operators could be used at the same time. Therefore, in this research we induce FPTs.

In this research paper, we present a two-step fuzzy pattern tree based method for predicting protein binding sites. In the first step, we extract the most relevant features of the amino acids of the proteins. Then, in the second step we induce fuzzy pattern trees using the extracted amino acids' features. The induced fuzzy pattern trees would be used to classify the amino acids as binding or non-binding sites. We present some experimental results of the evaluation of the method.

In section 2, our method is presented. Section 3 provides some experimental results of the evaluation of the method, while section 4 concludes the paper and gives some directions for further improvements.

II. OUR FUZZY PATTERN TREE BASED METHOD

In this paper we present a method for detecting protein binding sites. Our method contains two steps. In the first step, we extract the most relevant features of the amino acids of the protein molecules. In the second step, we induce fuzzy pattern trees that would be used to classify the amino acid residues as binding or non-binding sites.

A. First step – Extraction of the most relevant features of the amino acid residues

In this step, we extract the following amino acids' features: accessible surface area, depth index, protrusion index and hydrophobicity. These features are most commonly found in the literature as the most relevant features for predicting the protein binding sites. Therefore, in this step we focus only on these features.

The accessible surface area (ASA) is one of the most important characteristic of the amino acids regarding their preferences to be involved in interactions with other proteins. ASA is first described in 1971 by Lee and Richards [20]. In the literature, it is commonly calculated using the rolling ball algorithm [5]. In this algorithm, a rolling 'probe' with some predefined radius is used, and it is rolled around the inspected protein. In this research we use a rolling 'probe' with radius of 1.4 Å, which is same as the radius of the water molecule. This value is most commonly used in the literature. Amino acids contain several atoms which are folded in a particular way in the 3D space. For each atom, the accessible surface is calculated as the surface area of the atom that can get in contact with the rolling 'probe'. Finally, the ASA of a given amino acid is calculated as a sum of the ASA values of the atoms that constitute the amino acid.

Many amino acids which are part of the protein molecule are deeply in its interior, and they could not be reached by the amino acids of the interacting proteins. Therefore, they could not be binding sites. In order to avoid unnecessary predictions about these amino acids, first we estimate which amino acids of a given protein are on its surface. This estimation is made based on the previously calculated values for the accessible surface area of each amino acid. Namely, according to [21] if the fraction of the surface area of a given amino acids that could be reached by the rolling 'probe' is equal or greater than 5%, then we consider that amino acid as a surface residue. Otherwise, the amino acid is considered as a deeply buried residue. In this estimation we use the values for the total surface area of the amino acids given in [21]. In the second step of our method, we take into account only the surface amino acids, because the amino acids located in the protein interior could not be binding sites.

Another amino acids' feature which is also widely used for protein binding sites prediction is the depth index (DPX) [6]. For each of the atoms of the amino acid, we calculate its depth index, which is the distance between the atom and its closest solvent accessible atom (atom with ASA > 0). Then, we calculate the depth index of the amino acid as an average of the depth indices of all its atoms.

Next, we extract the protrusion index (CX) [7] of the protein amino acids. For each of the non-hydrogen atoms, we calculate the number of heavy atoms within a sphere with some predefined radius. According to [7], we set this radius to 10 Å. In order to calculate the occupied volume in the sphere V_{int}, we multiply the previously calculated number of heavy atoms in the sphere by the mean volume of the atoms (20.1 Å). Then, V_{ext} is calculated as the remaining volume of the sphere, while the protrusion index of the atom is calculated as $CX=V_{ext}/V_{int}$ [7]. In this way, the non-hydrogen atoms which are surrounded by many heavy atoms within a given radius would have low protrusion index, while the non-hydrogen atoms surrounded by few heavy atoms would have large protrusion index. Finally, the protrusion index of a given amino acid is calculated as an average of the protrusion indices of its non-hydrogen atoms.

The fourth amino acids' feature that we consider in this research is hydrophobicity. This feature is related with the hydrophobic effect. Namely, hydrophobic amino acids are more commonly found in the protein interior, while hydrophilic amino acids are more likely located towards the protein surface. In the literature, several scales for amino acids' hydrophobicity can be found. In our research, we use the scale proposed by Kyte and Doolittle [8], which is the most commonly used scale.

B. Second step – Inducing fuzzy pattern trees

In this step, we induce fuzzy pattern trees (FPTs) [19] for predicting the binding sites of the protein molecules. In this research, for inducing fuzzy pattern trees we use the method given in [19]. For each class (classes of binding and non-binding sites in our case), a separate tree is induced. In the test phase, a given query amino acid residue is presented to the tree for each class, and it is classified in the class for which the highest similarity is achieved.

The induction of fuzzy pattern trees starts with fuzzification of the data set. We use the fuzzy membership functions (FMFs) introduced by Zadeh [22], which are straight-line FMFs. They are simple, and in many cases can lead to models with high prediction accuracy. However, according to Zadeh [22], the convex membership functions can significantly improve the prediction power of the models. Therefore, in this research we also use the Gaussian FMF. In the fuzzification, the fuzzy set is labelled with fuzzy terms. In this research we use four amino acids' features, so if the number of FMFs per attribute (N) is set to 5, in the process of inducing tree for a given class we obtain 20 primitive trees (trees at level 0). The primitive trees for a given class can be used for classification in that class based on the membership value of a single fuzzy term. For each primitive tree we calculate the similarity between the membership values the corresponding fuzzy term for an amino acids' feature and the membership values of a given fuzzy term for the class attribute. Then, we select the primitive tree that lead to highest similarity. In this research we use the standard RMSE similarity metric.

Primitive trees could not gain a high prediction power, thus these trees are further aggregated using fuzzy aggregation operators. There are several types of fuzzy aggregation operators, and in this research we use only the basic operators, i.e. algebraic AND and OR. The main advantage of the fuzzy pattern trees (FPTs) [19] over the fuzzy decision trees (FDTs) [14] is that in the induction of FPTs different fuzzy aggregation operators could be used together at same time, thus providing models with higher prediction power.

Using fuzzy aggregation operator, we build the trees at level 1 by aggregating the primitive tree with highest similarity and the other primitive trees from level 0. In this way, if the number of primitive trees is 20 the number of trees

at level 1 would be 19. Again, among the trees from level 1 the tree with the highest similarity is chosen, and at level 2 it is aggregated with the other trees from level 0 and level 1 that do not have highest similarity among the trees from their levels. In this way at level 2 we obtain 37 trees. The same process is repeated for the other levels. In this way, we build the final tree bottom-up.

In the process of inducing the final tree, the current tree with highest similarity could be aggregated with primitive trees (trees at level 0), or some more complex trees (trees from the other levels). In this way we can obtain two types of models: simple model (SM) and general model (GM). The main difference between the SMs and GMs is that in SMs the aggregation is done between the current tree with highest similarity and the primitive trees from level 0 excluding the primitive tree with highest similarity, while in GMs in the aggregation also the trees from the other levels are considered. In order to control the model complexity, the number of levels in the tree is limited to some predefined value. Namely, we limit the number of levels with parent nodes in the model trees. In this research we constrain the number of levels with parent nodes to 5. In section 3, besides the prediction power of the model, we will also inspect the model complexity that would be measured as a number of leaf nodes in the tree, which are actually the nodes labelled with fuzzy terms.

III. Experimental results

In this section, we present experimental results regarding the evaluation of our method. We use the BIND database [23], which contains knowledge about the protein binding sites that is obtained in experimental manner. In the research, we do not consider the entire BIND database, but we take into account only a representative protein chains. In this selection, we filter the protein chains thus each pair of protein chains has less than 40% sequence similarity using the selection criterion given in [24]. Next, from this representative data set we form the test set by selecting representative protein chains so that each pair of protein chains in the test set has less than 20% sequence similarity [24]. The rest of the protein chains in the representative data set are taken as training data. In this way, we obtained a training data set with 1062 protein chains, and a test set with 1858 protein chains. In the training data set, the total number of amino acid residues in the proteins is 365862, while in the test set this number equals 608434. After applying the filter for extracting the surface amino acids (amino acid residues with at least 5% accessible surface area), we obtain 284168 surface amino acids in the training data set, and 484637 surface amino acids in the test data set. According to the knowledge stored in the BIND database, from all surface amino acids in the training data set, only 26889 are classified as binding sites, while in the test set only 47501 amino acids are classified as binding sites. Since the non-binding sites class is dominant over the binding sites class, before the prediction models are induced, we first balance the training data set in order to prevent building models that would be biased towards the dominant class, the non-binding sites class in this case. This balancing is done in that way that the binding amino acids are taken into account several times until the distribution in the data become uniform.

Since the balancing is not done on the test data set, we have to be careful in the decision which evaluation measure to use in the evaluation of the method. The measure should be appropriate for cases where some class is dominant, as it is in this case. We chose the Area under ROC curve (AUC-ROC) evaluation measure. In order to calculate this measure, first we have to calculate TP (true positives), TN (true negatives), FP (false positives) and FN (false negatives). Positive residues are the residues which are predicted as protein binding sites, while negative residues are the residues which are predicted as non-binding sites. TP is the number of the correctly predicted positive residues, TN is the number of correctly predicted negative residues, FP is the number of non-binding sites residues predicted as binding sites, while FN is the number of binding sites residues predicted as non-binding sites.

Next, we calculate the following measures: True Positive Rate (TPR) and True Negative Rate (TNR). TPR is calculated as TP/(TP+FN), while TNR is calculated as TN/(TN+FP). Finally, the Area under ROC curve (AUC-ROC) is calculated as AUC-ROC=TPR*TNR+TPR*(1-TNR)/2+TNR*(1-TPR)/2 =(TPR+TNR)/2. This measure is the most appropriate measure, especially when the classes have very different number of examples, which is case in our research. AUC-ROC obtains values between 0 and 1, where 1 corresponds to perfect prediction, and 0 corresponds to inverse prediction.

Besides the models' accuracy measured by TPR, TNR and AUC-ROC, we will also examine the model complexity. As it was described in Section 2, we obtain separate model for each class, and in the testing phase, the query amino acid is classified in the class for which the tree gives highest similarity. Since we have two classes (binding and non-binding sites), we will obtain two fuzzy pattern trees in the model, therefore in the evaluation of the model complexity we will consider the total number of leaf nodes in both trees.

In this research we present the analysis using simple models (SMs), where only primitive trees are aggregated with the current tree with highest similarity. However, we have also made some experiments using general models (GMs), but the results showed that even if we allow higher granulation to the tree by aggregating more complex trees, the tree still remains compact as in simple models. Therefore, the values for TPR, TNR, AUC-ROC and model complexity remain the same as in SMs.

In Table 1, Table 2 and Table 3, we present the results of the evaluation of our method using different types of FMFs (triangular, trapezoidal and Gaussian) and different number of FMFs per feature (N = 2, 3, 4, 5 and 10). In this analysis we use RMSE similarity measure, algebraic AND and OR fuzzy aggregation operators, 0 low level trees (thus we induce only SMs), and the maximum number of levels with parent nodes in the tree (the number of levels with nodes associated with fuzzy aggregation operators) is set to 5. Using this experimental set up, the trees could have maximum 6 leaf nodes, and each model could have maximum 12 leaf nodes.

Table 1: The results using different number (N) of triangular fuzzy membership functions per feature

N	TPR	TNR	AUC-ROC	Number of leaves
2	0,3962	0,6262	0,5112	12
3	0,4016	0,7113	0,5564	12
4	0,6472	0,4815	0,5644	10
5	0,2876	0,8042	0,5459	9
10	0,5504	0,5550	0,5527	12

Table 2: The results using different number (N) of trapezoidal fuzzy membership functions per feature

N	TPR	TNR	AUC-ROC	Number of leaves
2	0,4132	0,6290	0,5211	8
3	0,2250	0,8522	0,5386	12
4	0,4016	0,7120	0,5568	12
5	0,5533	0,5766	0,5649	8
10	0,3432	0,7635	0,5534	12

Table 3: The results using different number (N) of Gaussian fuzzy membership functions per feature

N	TPR	TNR	AUC-ROC	Number of leaves
2	0,3859	0,6434	0,5147	12
3	0,4016	0,7113	0,5564	12
4	0,6472	0,4815	0,5644	10
5	0,2204	0,8568	0,5386	11
10	0,5539	0,5531	0,5535	12

According to the results given in Table 1, we can conclude that for triangular FMF the most accurate model is obtained using 4 FMFs per attribute. Similarly, the most appropriate number of FMFs per attribute using trapezoidal and Gaussian FMFs is 5 and 4 respectively, see Table 2 and Table 3. From the results given in these tables, we can also conclude that when lower number of FMFs per attribute is used, the prediction power of the model is worse, and as this number increases towards 4 or 5, the predication power also increases. However, for too large number of FMFs per attribute, the induction of the models lasts significantly longer because the number of primitive trees increases, while the prediction power is still not improved. So the optimal number of FMFs per attribute is 4 for the triangular and Gaussian FMFs, and 5 for the trapezoidal FMF.

It is also interesting that the most accurate models obtained in this analysis are even with lower complexity than most of the other models. For example, for triangular FMF using N=4, the model has 10 leaf nodes, that is less than the number of leaf nodes using N = 2, 3 and 10. The same conclusion can be made for the other types of FMFs. According to the results given in previous tables, we can conclude that the most accurate model obtained using 5 trapezoidal FMFs per feature achieves AUC-ROC=0.5649, and has only 8 leaf nodes.

However, in this research we used only the most basic types of FMFs. Also other convex FMFs like bell and log-normal could be used. According to the similarity measure, we used the most basic measure, i.e. RMSE. Further, we can apply some other more sophisticated measures in order to improve the method. Regarding fuzzy aggregation operators, in this research we used only the algebraic AND and OR operators. Besides them, we can also incorporate other fuzzy aggregation operators that could also improve the prediction power of our method.

IV. Conclusion

In this paper, we presented a fuzzy pattern trees based method for predicting the protein binding sites. The method performs in two steps. In the first step, the most relevant features of the amino acid residues of the protein structure were extracted. Later, in the second step, these features were used for inducing fuzzy pattern trees for classifying the amino acid residues as binding or non-binding sites.

In the evaluation of our method we used the BIND database, which contains experimentally obtained knowledge about the protein binding sites. We examined tree different types of membership functions, i.e. triangular, trapezoidal and Gaussian. For triangular FMF the most accurate model is obtained using N = 4 FMFs per attribute. Similarly, the most appropriate number of FMFs per attribute using trapezoidal and Gaussian FMFs is 5 and 4 respectively. The results showed that when lower number of FMFs per attribute is used, the accuracy of the model is lower, while for too large number of FMFs per attribute the induction process lasts longer and the predication power is still not increased. The optimal number of FMFs per attribute is 4 for the triangular and Gaussian FMFs, and 5 for the trapezoidal FMF. We also noticed that generally the models with higher prediction power have lower complexity. The most accurate model obtained in this research achieved AUC-ROC = 0.5649, and lowest complexity (8 leaf nodes).

We identified several directions for further improvement of our method. First, regarding the types of the FMFs, in this research we used only the basic FMFs, so furthermore some other convex FMFs can be used. For measuring the similarity, we used the most basic measure, RMSE, so we expect that using other similarity measure we can improve the method. Although in this research we used the fuzzy pattern trees which against fuzzy decision trees allow simultaneous usage of different fuzzy aggregation operators, we still used only the most basic fuzzy aggregation operators, i.e. algebraic AND and OR. Further, we plan to improve the method by incorporating other more sophisticated fuzzy aggregation operators.

References

[1] A. E. Todd, C. A. Orengo and J. M. Thornton, "Evolution of function in protein superfamilies, from a structural perspective," *J. Mol. Biol.*, vol. 307, no. 4, pp. 1113–1143, 2001.

[2] M. Kirac, G. Ozsoyoglul and J. Yang, "Annotating proteins by mining protein interaction networks," *Bioinformatics*, vol. 22, no. 14, pp. e260–e270, 2006.

[3] A. R. Panchenko, F. Kondrashov and S. Bryant, “Prediction of functional sites by analysis of sequence and structure conservation,” *Protein Science*, vol. 13, no. 4, pp. 884–892, 2004.
[4] N. Tuncbag, G. Kar, O. Keskin, A. Gursoy and R. Nussinov, “A survey of available tools and web servers for analysis of protein-protein interactions and interfaces,” *Briefings in Bioinformatics*, vol. 10, no. 3, pp. 217–232, 2009.
[5] A. Shrake and J. A. Rupley, “Environment and exposure to solvent of protein atoms,” *Lysozyme and insulin, J. Mol. Biol.*, vol. 79, no. 2, pp. 351–371, 1973.
[6] A. Pintar, O. Carugo and S. Pongor, “DPX: for the analysis of the protein core,” *Bioinformatics*, vol. 19, no. 2, pp. 313–314, 2003.
[7] A. Pintar, O. Carugo and S. Pongor, “CX, an algorithm that identifies protruding atoms in proteins,” *Bioinformatics*, vol. 18, no. 7, pp. 980–984, 2002.
[8] J. Kyte and R. F. Doolittle, “A simple method for displaying the hydropathic character of a protein,” *J. Mol. Biol.*, vol. 157, no. 1, pp. 105–132, 1982.
[9] A. S. Aytuna, A. Gursoy and O. Keskin, ”Prediction of protein-protein interactions by combining structure and sequence conservation in protein interfaces,” *Bioinformatics*, vol. 21, no. 12, pp. 2850–2855, 2005.
[10] H. Neuvirth, R. Raz and G. Schreiber, “ProMate: a structure based prediction program to identify the location of protein-protein binding sites,“ *J. Mol. Biol.*, vol. 338, no. 1, pp. 181–199, 2004.
[11] J. R. Bradford and D. R. Westhead, “Improved prediction of protein-protein binding sites using a support vector machines approach,” *Bioinformatics*, vol. 21, no. 8, pp. 1487–1494, 2005.
[12] U. Ogmen, O. Keskin, A. S. Aytuna, R. Nussinov and A. Gursoy, “PRISM: protein interactions by structural matching,” *Nucleic Acids Res.*, vol. 33, no. 2, pp. W331–W336, 2005.
[13] S. Jones and J. M. Thornton, “Prediction of protein-protein interaction sites using patch analysis,” *J. Mol. Biol.*, vol. 272, no. 1, pp. 133–143, 1997.
[14] C. Z. Janikow, “Fuzzy decision trees: issues and methods,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 28, no. 1, pp. 1–14, 1998.
[15] C. Olaru and L. Wehenkel, “A complete fuzzy decision tree technique,” *Fuzzy Sets and Systems*, vol. 138, no. 2, pp. 221–254, 2003.
[16] A. Suárez and J. F. Lutsko, “Globally optimal fuzzy decision trees for classification and regression,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1297–1311, 1999.
[17] X. Wang, B. Chen, G. Olan and F. Ye, “On the optimization of fuzzy decision trees,” *Fuzzy Sets and Systems*, vol. 112, no. 1, pp. 117–125, 2000.
[18] Y. -L. Chen, T. Wang, B. -S Wang and Z. –J. Li, “A Survey of Fuzzy Decision Tree Classifier,” *Fuzzy Information and Engineering*, vol. 1, no. 2, pp. 149–159, 2009.
[19] Z. H. Huang, T. D. Gedeon and M. Nikravesh, ”Pattern trees induction: a new machine learning method,” *IEEE Transaction on Fuzzy Systems*, vol. 16, no. 3, pp. 958–970, 2008.
[20] B. Lee and F. M. Richards, “The interpretation of protein structures: Estimation of static accessibility,” *J. Mol. Biol.*, vol. 55, no. 3, pp. 379–400, 1971.
[21] C. Chothia, “The Nature of the Accessible and Buried Surfaces in Proteins,” *J. Mol. Biol.*, vol. 105, no. 1, pp. 1–12, 1976.
[22] L. A. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, pp. 338–353, 1965.
[23] G. D. Bader, I. Donaldson, C. Wolting, B. F. Ouellette, T. Pawson and C. W. Hogue, “BIND: the Biomolecular Interaction Network Database,” *Nucleic Acids Res.*, vol. 29, no. 1, pp. 242–245, 2001.
[24] J. –M. Chandonia, G. Hon, N. S. Walker, L. L. Conte, P. Koehl, M. Levitt and S. E. Brenner, “The ASTRAL Compendium in 2004,” *Nucleic Acids Res.*, vol. 32, pp. D189–D192, 2004.