Bioinformatics and Fuzzy Logic

Dong Xu, Member, IEEE, Rajkumar Bondugula, Mihail Popescu, Member, IEEE, and James Keller, Fellow, IEEE

Abstract—Many biological systems and objects are intrinsically fuzzy. Fuzzy set theory and fuzzy logic are ideal frameworks for describing some biological systems/objects and providing suitable computational methods for a widely range of bioinformatics problems. In this paper, we present two examples of using fuzzy set theory in bioinformatics, one in fuzzy measurement of ontological similarity and its application in bioinformatics, and the other in the application of the fuzzy k-nearest neighbor algorithm in protein secondary structure prediction. We also review other “fuzzy” methods for bioinformatics applications.

I. INTRODUCTION

Since the early 1980s, the advent of DNA sequencing has led to exponential growth in molecular data. Sequencing of hundreds of genomes has been completed and many more are in progress. Genomic sequencing has opened a new avenue to study biological systems on large scales, paving the way for studying other high-throughput data. Today, due to the availability of high-throughput measurement technologies, it is possible to use a broad range of experimental data to expand the genome-scale studies from sequence-level information to higher-level functions. Microarray technology is a powerful tool to systematically measure gene expression across whole cells and tissues under varying experimental conditions or over a time course. Different types of experimental methods can generate different types of protein-protein interaction information. As data expands exponentially, the demand to analyze the data, including data mining and hypothesis generation, also increases drastically. This leads to the rise of bioinformatics.

There are numerous definitions of bioinformatics. A widely accepted definition for bioinformatics by the US National Institutes of Health (http://www.bisti.nih.gov/) is “Research, development, or application of computational tools and approaches for expanding the use of biological, medical, behavioral or health data, including those to acquire, represent, describe, store, analyze, or visualize such data”. The scope of bioinformatics is very broad. Currently, bioinformatics is focused on biomolecular aspects, including biological sequence analysis, protein structure analysis and prediction, gene expression data analysis, computational proteomics, gene ontology, biological pathway prediction, epigenomics analyses, etc. Bioinformatics has been developed extremely fast and produced enormous impact on the research of biology and medicine in recent years. Thousands of bioinformatics databases and tools are available. More and more experimental biologists have realized the importance of bioinformatics, as the need to manage and analyze the massive amount of data generated. Many biologists have begun to use bioinformatics tools themselves, especially through Web interfaces. The sequence comparison tool BLAST [1] becomes a household name to biologists. Coming from the study of bioinformatics, some novel experimental technologies have been developed.

As massive biological data have become a fundamentally important resource during discovery of new biological knowledge, a central dogma for bioinformatics is to identify meaningful information (or statistically significant patterns) from data and correlate such information to biological knowledge. However such a task is highly challenging in many cases. The information-rich data are heterogeneous in nature, noisy, and incomplete, as well as containing misleading outliers. Furthermore, biological systems, due to adaptability, evolution, redundancy, robustness, and emergence, are extremely complex. The challenge has drawn a wide range of studies from computer sciences and various computer science technologies have been applied. The most notable applications include dynamic programming, neural networks, hidden Markov models, support vector machines, etc. Fuzzy set theory and fuzzy logic have been used in bioinformatics, but to much less extent than graph theory and machine learning. We believe there is much bigger potential for fuzzy set theory and fuzzy logic in bioinformatics.

In this paper, we will discuss why fuzzy concepts and methods can play a more important role in studying biological problems. We will present two examples of using the fuzzy set theory in bioinformatics. The first example is fuzzy measurement of ontological similarity and its application in bioinformatics. This example shows descriptions of intrinsic fuzziness in biological concepts. The second example applies the fuzzy k-nearest neighbor algorithm for protein secondary structure prediction, demonstrating fuzzy set theory as a powerful computational

Dong Xu (phone: 573-882-7064; fax: 573-882-8318; email: xudong@missouri.edu) and Rajkumar Bondugula are with Department of Computer Science and CS Bond Life Science Centre, University of Missouri-Columbia MO 65211-2060. USA.

Mihail Popescu is with Health Management and Informatics Department, University of Missouri-Columbia MO 65211-2060. USA.

James Keller is with Electrical and Computer Engineering Department, University of Missouri-Columbia MO 65211-2060. USA.
tool for bioinformatics. We will also review other bioinformatics applications using fuzzy techniques. Finally we will summarize and provide a future outlook.

II. FUZZINESS IN BIOLOGICAL PROCESSES AND CONCEPTS

Almost all of bioinformatics problems up to now are formulated in a deterministic manner. Most of these problems are defined by a fixed objective function and solved through optimization. Many dynamic processes, such as gene expression regulation, are also modeled using differential equations with deterministic behavior. However, there are at least three situations in which fuzziness should be considered, i.e., intrinsic fuzziness in biological systems, multiple roles of a biological object, and fuzzy descriptions of biological phenomena.

In recent years, there is an increasing awareness in the fuzzy aspects of biological systems, which is sometimes referred to as a paradigm shift for “new biology”. More and more evidence has been found that in biological systems many processes are intrinsically fuzzy rather than deterministic. Numerous examples have demonstrated that fuzzy effects are physiologically and evolutionarily important in the development and function of living organisms. For example, it was found that the immune system, as a consequence of central tolerance, is able to recognize both self and nonself antigens in a fuzzy manner [2]. In this case, the key players of the immune system, T cells and antibodies, can recognize a given self or “foreign”-reactive cell (nonself) to a certain degree, but not deterministically. Such a fuzzy feature of the immune system may shed light on mechanisms of autoimmune diseases, such as cancers. Fuzziness and random fluctuations are intrinsically important for balancing fidelity and diversity in eukaryotic gene expression and may produce variability in cellular behavior [3]. The fuzziness also provides additional functional modalities on the enzymatic futile cycle mechanism that include stochastic amplification and signaling. The stochastic/fuzzy behavior may offer a novel type of control mechanism in pathways that contain these cycles [4].

A biological object may have multiple roles, resulting in fuzzy memberships. One gene may be involved in different functions or pathways. Beta-catenin is such a multifunction protein, playing important roles in both cell-cell adhesion and intracellular signaling [5]. In practice, when we cluster genes using biological data (e.g., microarray gene expression data), fuzzy memberships for a gene or fuzzy clustering may be useful to serve as descriptor. In this case, one gene can be present in two or more clusters simultaneously, with partial or full membership in each cluster [6].

Another type of fuzziness in biomedical research is fuzzy description of biological terms. Our characterizations of many biological concepts often have difficulty fitting into a deterministic (crispy) description. As a result, knowledge, concepts, and representations of biological terms may also be fuzzy, and fuzzy set theory is useful to describe these terms. To illustrate, the concept of “protein function” is rather fuzzy because it is often based on whimsical terms or contradictory nomenclature [7]. This currently presents a challenge for functional genomics. In addition, descriptions for similarity and typicality can be fuzzy: how much do two proteins resemble each other, what properties do they (partially) share, how close is a given protein to the prototypical sequence of a protein family, etc. Such fuzziness could result from the limitations of classifications, natural language or poor understanding of the underlying mechanisms. Tolerance of fuzziness allows us to explore these biological concepts effectively.

III. APPLICATION OF FUZZY SET THEORY IN ONTOLOGICAL SIMILARITY

Due to the fuzziness of many biological concepts, fuzzy set theory and fuzzy logic are well suited for studying some biological problems. Here we illustrate the utility of fuzzy models related to ontological similarity.

One of the most important objects in bioinformatics is a gene product (protein or RNA). A gene product is described by various attributes among the most important being its sequence (nucleotides or amino acids) and Gene Ontology [8] annotation. An ontology is a knowledge representation of a certain scientific domain that consists in a hierarchically organized controlled vocabulary. The controlled vocabulary facilitates the retrieval of the synonymic terms. The hierarchical organization (taxonomy), usually developed by human experts, encodes the domain knowledge. The hierarchical organization of ontological terms facilitates the quantification of semantic relations between terms. Ontologies have become widely used in modern information retrieval mainly due to their controlled vocabulary properties. There are many ontologies used in bioinformatics such as KEGG, EcoCyc, Ontology for Molecular Biology (MBO), etc. On the other hand, the Gene Ontology (GO), which classifies gene properties (functions, cell localization and processes), has the widest acceptance. An example of GO is shown in Figure 1. As described in Section II, sometimes gene functions should be fuzzy. As a result, the classification of GO and the mapping between gene and GO can be fuzzy as well.

Most of the time the similarity between two gene products is assessed based on their sequences and is performed using the BLAST software [1]. However, important functional relations between gene products can be uncovered by using the ontological similarity. Although current methods for gene ontology have achieved significant success, we believe that the hierarchical structure has not been fully exploited for computing the semantic relations between objects described by ontology terms.

Formally, given two gene products, G1 and G2, we can consider them as being represented by collections of GO
Figure 1. Partial view of the Gene Ontology (GO) taxonomy. GO has three types of terms (branches): molecular function (biochemical function), biological process (cellular role) and cellular component (location in the cell). Each GO term has assigned a GO id (for example GO:0005198 represents "cAMP catabolism"). The computational use of GO requires the assignment of a weight, \( g \), to each GO term that reflects its “information content”. The higher the \( g \) of a GO term, the more important the term is in the computing the similarity of two genes that contain it. The root term, “gene ontology”, has \( g=0 \) while a leaf term has \( g \) close to 1 (\( g<1 \)).

```
<table>
<thead>
<tr>
<th>GO Term</th>
<th>GO ID</th>
<th>( g )</th>
</tr>
</thead>
<tbody>
<tr>
<td>cAMP catabolism</td>
<td>GO:0005198</td>
<td>0.42</td>
</tr>
<tr>
<td>GMP catabolism to IMP</td>
<td>GO:0005201</td>
<td>0.58</td>
</tr>
<tr>
<td>cell surface receptor linked signal transduction</td>
<td>GO:0007155</td>
<td>0.44</td>
</tr>
<tr>
<td>galactosylceramide metabolism</td>
<td>GO:0005581</td>
<td>0.65</td>
</tr>
</tbody>
</table>
```

![Gene Ontology Diagram](image1.png)

Figure 2. BLAST similarity matrix (a) and fuzzy measure similarity matrix (b) computed for a set of 194 gene products. The gray scale indicates the level of similarity between the sequences of two genes, with darker spots showing higher similarities. Each point \((i,j)\) in the left matrix represents the similarity \( s_B(G_i, G_j) \), computed using the percent similarity of the sequences \( G_i \) and \( G_j \) as given by BLAST. To normalize, \( s_B \) is computed as \( s_B(G_i, G_j) = \frac{s_B(G_i, G_j)}{\max\{s_B(G_i, G_i), s_B(G_j, G_j)\}} \) where \( s_B(G_i, G_j) \) is the actual percent sequence similarity obtained by running BLAST between \( G_i \) and \( G_j \). Each point in the right matrix represents the fuzzy measure similarity, \( s_{FMS}(G_i, G_j) \), between \( G_i \) and \( G_j \). The 194 gene products were clustered in 3 families based on the BLAST similarity by the ENSEMBL browser [www.ensembl.org]: myotubularins, receptor precursors, and collagens. The FMS uncovers a further subdivision of the collagen family (the lower-right corner of the fuzzy measure similarity matrix), which is supported in the biological literature [13]. The related region in the BLAST matrix (indicated by circles) has no obvious structure.

![BLAST vs FMS Similarity Matrix](image2.png)

The approach is to compute the similarity using only the controlled vocabulary property of GO, that is by exactly matching the terms. The best known similarities from this category are Jaccard, \( s_J = |G_1 \cap G_2| / |G_1 \cup G_2| \), and Dice, \( s_D = 2|G_1 \cap G_2| / (|G_1| + |G_2|) \), where \(|\cdot|\) denotes the cardinality of the set. There are several problems with this approach based on counting. First, since the number of annotation for each gene is low (typically \( n<10 \)), the similarity between most genes is 0 (no common terms exist). From Figure 1, if \( G_1 = \{T_1\} \) and \( G_2 = \{T_1, T_2\} \) then \( s_J(G_1, G_2) = 0 \). Second, counting approaches (such as Jaccard or Dice) while fast, do not account for the information content of the terms [9]. As a consequence, for example, two genes described only by the term GO:0007154 (“cell communication”) could be considered identical ignoring the fact that GO:0007154 is a very general term.
To further quantify a GO relationship, we use information content to describe the relationship between any two GO terms. The information content of a term can be computed using many approaches [10], among the most popular includes the one based on the depth in the taxonomy and the one based on the term frequency in a given corpus [11]. As a result, each term obtains a weight, \( g \in [0,1] \) (see Figure 1), that reflects its information content. The easiest way to compute the term’s depth is using the depth, \( d \), of a GO term in the GO taxonomy such as \( g = 1 - 1/2^d \) (where \( d = 0 \) for the root term). Another way is to use the number of appearances of a GO term in a certain set of documents (database) [12]. Then, the similarity \( s_{ij}(G_1, G_2) \) has to account for the weights attached to the set of GO terms that describe \( G_1 \) and \( G_2 \): \( \{g_{11}, ..., g_{1i}, ..., g_{1n}\} \) and \( \{g_{21}, ..., g_{2j}, ..., g_{2m}\} \). To address the above problems a fuzzy measure approach for computing the similarity of two gene products is presented in [9]. The value is given as:

\[
s_{FMS}(G_1, G_2) = 0.5(\mu_1(G_1 \cap G_2)^k + \mu_2(G_1 \cap G_2)^k),
\]

where, \( \mu_1, \mu_2 \) are fuzzy measures computed on \( G_{1,2} \) respectively and \( (G_1 \cap G_2)^k \) denotes the augmented intersection set created by considering both the terms and their nearest common ancestors (NCA). Formally, the augmented intersection is:

\[
(G_1 \cap G_2)^k = (G_1 \cap G_2) \cup \{\text{NCA}(T_{1i}, T_{2j})\},
\]

The advantage of the ontological similarity is that it better reflects the functional relations between gene products than BLAST. Figure 2.a shows the similarity between a set of 194 gene products computed based on their sequence using BLAST [1]. Figure 2.b displays the similarity between the same gene products computed based on GO using sFMS. Clearly, the FMS was able to uncover a further subdivision of the collagen family (see the lower-right corner of the fuzzy measure similarity matrix), which in fact is in agreement with a later published result in the biological literature [13].

The best known approaches [10] that account for term information content compute first the pair-wise term similarities, \( s_{ij}(T_{1i}, T_{2j}) \) (as the information content of the nearest common ancestor [11]) and then use an aggregation operator (such as average or maximum) to compute \( s(G_1, G_2) \). In [14] the approach was generalized to any ordered weighted average (OWA) operator as:

\[
s_k(G_1, G_2) = \sum_{i=1}^{m} w_{ik} s_{ik},
\]

where \( s_{ik} \) denotes an ordering of \( s_{ij}, s_{i(1)j}, ..., s_{i(n)j} \), and \( w \) is a weight vector such that \( \sum w_{ik} = 1 \).

The fuzzy similarity measures introduced so far were designed to generalize the previously introduced similarity measure based on max and average [12,15]. Furthermore, each term \( T_i \) in an annotation set may have a given certainty attached to it, \( c_i \). To account for the term (un)certainity the Choquet integral has been used to compute the gene similarity [9] as:

\[
s_k(G_1, G_2) = \min\left\{ \sum_{i=1}^{m} \mu(c_{i(1)}, ..., c_{i(n)}) - \mu(c_{i}, ..., c_{i(n)}) \right\} \cdot s_{ik},
\]

where \( s_{ik} \) denotes an ordering of the pair-wise similarities, \( s_{ij} \), and \( \mu \) is a fuzzy measure on the set of corresponding pairwise uncertainties \( c_{ij} \) computed as \( c_{ij} = \min(c_{1i}, c_{2j}) \).

The ontological similarity between gene products has been used in various bioinformatics applications, such as functional summarization of gene clusters in microarray experiments [16,17], gene family detection [9,14], GO based cluster validity measures [18], protein-protein interaction prediction [19] and microarray data analysis [15,20].

Reference [16] addressed the problem of constructing a functional summarization of groups of gene products, which are found by clustering a database of such products annotated by GO. Our method builds the “most representative term” (MRT) for each cluster in three increasingly sensitive ways. Initially, we perform crisp hierarchical clustering using BLAST and the fuzzy measure similarity and find the MRTs as the terms of highest frequency in the description of the gene products. Using weights from the fuzzy partition matrix generated by a relational fuzzy clustering algorithm (i.e., NERFCM), we show how more specific MRTs can be made. Finally, weighting these memberships by the information content of each term further increases the specificity of the functional annotation of the clusters. A related approach can be found in [17].

In [9] three subfamilies of the collagen \( \alpha \) family were discovered by hierarchically clustering the similarity matrix computed using the fuzzy measure similarity (Figure 2). One can observe in the lower-right corner of the fuzzy measure similarity matrix (Fig. 2.b, circled) a 3-cluster structure. The related region of the BLAST similarity matrix (Fig. 2.a, circled) does not show any obvious structure.

The clustering of the gene product similarity matrices was performed in [9] using hierarchical clustering. However, other clustering procedures are possible. In [16] a relational fuzzy c-means (NERFCM) was used to assign fuzzy memberships to gene products in clusters. However, object-based fuzzy c-means could be employed to cluster the gene similarity matrix. In this case, each gene product \( G_i \) is considered to be represented by its similarities to all \( N \) gene products. Hence, the feature vector for \( G_i \) is \( \{s(G_i, G_1), ..., s(G_i, G_n)\} \) and \( s(G_i, G) \) which is, in fact, a row in one of the matrices from Figure 2. Using such similarities as a feature vector, we can use any other objectual clustering algorithm such as crisp c-means.

Reference [18] introduced a cluster validity measure defined from GO similarity to discover functionally related expressed gene clusters in microarray experiments. GO similarity was also used in the analysis of microarray experiments [15]. A more general approach to ontology-based fuzzy information retrieval is presented in [21].
An interesting direction for the ontology based similarity utilizes support vector machines (SVM) [19,22]. In one such application [19], the GO similarity matrix computed between the proteins from a training set has been used to predict the behavior of an unknown protein. Due to the additive property of kernels, different type of information (such as sequence) can be fused just by adding the related similarity matrices.

IV. APPLICATION OF FUZZY SET THEORY IN PROTEIN SECONDARY STRUCTURE PREDICTION

Fuzzy set theory and fuzzy logic, as a powerful computational approach, also provides many useful techniques for bioinformatics problems, in optimization, prediction, modeling, etc, even when the related biological concepts are defined in a crisp manner. Here we use an application of protein structure prediction to explain.

Proteins are one of the most important molecules in life. They play a variety of roles depending on their types including structural proteins, catalytic proteins, storage and transport proteins, regulatory proteins, immune system proteins, signaling proteins, and so on. A protein is a sequence of amino acids that are linked by peptide bonds to form a poly-peptide chain called its primary structure. Short runs of these amino acids form regular structures called secondary structures. There are three types of secondary structures, i.e., helices, strands, and coils. The secondary structure elements are packed together into compact tertiary structure of the protein. Secondary structure prediction from a protein’s sequence plays an important role in characterizing protein structures and providing a basis for tertiary structure prediction [23,24]. In particular, the secondary structures of a protein provide the computational methods with constraints that reduce the search space greatly and therefore make the tertiary structure prediction more accurate and more efficient.

In spite of research efforts for more than three decades, the task of accurately predicting secondary structures of a protein still remains a challenging problem. Of all the successful prediction methods, the most popular systems are based on neural network approach, nearest neighbor techniques and hidden Markov model methods. Among them, nearest neighbor algorithms are simple and transparent, and they do not require retraining when new data is available. They are successful when sequences similar to the query sequence can be found in Protein Data Bank (PDB)[25], but have limited performance otherwise. We will now describe a secondary structure prediction system that uses the fuzzy k-nearest neighbor algorithm (FKNN) [26] and a neural network to enhance the prediction.

MUPRED is a secondary structure prediction system based on a Position Specific Scoring Matrix (PSSM) [27], the FKNN and a neural network. The method uses the PSSM information more effectively than existing systems, while it also has a better potential to utilize the information in PDB. An enhanced version of the MUPRED web server and executables can be freely accessed by general public at http://digbio.missouri.edu/mupred. The block diagram of the MUPRED is depicted in Figure 3.

We incorporated the evolutionary information of the protein into the prediction system in the form of the PSSM of the query protein. First, we calculate the PSSM of the query protein using the PSI-BLAST [27] program and the nr database (the non-redundant sequence database at http://www.ncbi.nlm.nih.gov). PSI-BLAST identifies sequences in nr that are similar to the query protein sequence, and then builds PSSM, which is basically a profile of aligning the query protein against these similar proteins sequences. The calculated PSSM is used to generate the feature set that is fed to the neural network. The feature set is generated using the FKNN with the following procedure: the PSSM is used to search for the protein fragments in the database of known protein structures with similar sub-sequences of the query protein. The hits obtained in the search process are collected and scored. Each hit is scored based on its expectation value (E-value) [27] using the following scoring scheme. The value S, which describes the distance between the two compared sequences in the hit, can be calculated as:

$$S = \max\{1.7 + \log_{10}(E-value)\},$$  \hspace{1cm} (5)

where 7 is an empirical value based on training. Based on this equation, the more similar between the two compared sequences, the smaller the E-value, and the smaller the distance S.

Homologous fragments whose similarities to the segments of query sequence are statistically significant have a low expectation value and therefore low S value and vice versa for fragments whose similarities are not significant. Any other scoring scheme that behaves like ‘distance’ can be used for this purpose. In the next step, these hits are labeled by the class to which the residue of the neighbor belongs. If the residue of the neighbor that is aligned with the current residue is in Helix state, the membership of the neighbor in Helix class is ‘1’ and ‘0’ in Strand and Coil classes. These labeled neighbors are then used to calculate the membership
value of the current residue in three classes. These membership values represent the confidence with which the current residue belongs to the three secondary structure classes.

The secondary structure state of each residue can be predicted from class membership values of the neighbors with the FKNN algorithm. The following algorithm adopted and modified from [26] provides the procedure to calculate the membership values of the current residue from the labeled neighbors. Let \( P = \{r_1, r_2, \ldots, r_i\} \) represent a protein with \( I \) residues. Each residue \( r \) has \( k \)-nearest neighbors, i.e., hit fragments that have a residue aligned with the current residue. Also, let \( u_j \) be the membership in the \( i^{th} \) class \( (i \in \{\text{Helix, Strand, Coil}\}) \) of the \( j^{th} \) neighbor. For each \( r \), the predicted membership value \( u_{i}(r) \) in class \( i \) can be calculated using the following equation:

\[
u_{i}(r) = \frac{\sum_{j=1}^{k} u_j \left( \frac{1}{\|S(r_i)/u_{j}\|^m} \right)}{\sum_{j=1}^{k} \left( \frac{1}{\|S(r_i)/u_{j}\|^m} \right)},\]

where \( S(r_i)/u_{j} \) represents the score of the \( j^{th} \) neighbor and \( m \) is a parameter.

It can be noticed from Equation 6 that the contribution of each neighbor (hit \( r_j \)) in the calculation of membership value of the current residue in each class is determined by the score \( S \). The influence of the score can be controlled by the fuzzifier \( m \) [26]. In our case, we set the value of \( m \) to 1.5. The FKNN algorithm uses these hits and calculates the membership value of each residue in each of Helix, Strand and Coil classes. The membership values are converted into vectors suitable for neural networks using a sliding window scheme i.e., the vector that represents the membership values of the current residue in three classes flanked by its neighbors on the both sides. The number of residues that will be flanked on each side is determined by the window size \( W \).

The system was trained and tested using a subset of proteins from the July 2005 release of PDBSelect (http://bioinfo.tg.fh-giessen.de/pdbselect/) database, which is a non-redundant set of known protein structures in PDB. The PDBSelect database consists of proteins such that the sequence identity between any two proteins in the database does not exceed 25%. The initial database of 2810 protein chains was filtered to select high-quality structures. We choose the proteins whose structures have been determined by X-ray crystallography method with a resolution not exceeding 3 Å and are of length greater than 40 residues. Only 1795 proteins remain after filtering the database and they constitute the local database of non-redundant structures. Of these 1795 proteins, we randomly chose 500 proteins for training both the FKNN algorithm and the neural network and reserved the rest of the 1295 structures for the test set.

We employ a fully connected neural network, trained with standard back propagation algorithm to further refine the secondary structure. The class membership values \( (u_i(r)) \) generated by the FKNN algorithm are used as the inputs to the neural network. We experimentally determined that the 25 nodes in the hidden layer result in an optimum performance. The output layer consists of 3 nodes that generate the final membership values in each of the three secondary structure classes. The prediction accuracy of the MUPRED is measured in terms of average 3-state accuracy \( (QS, QH, QC) \) number of correctly predicted residues in all three states divided by total number of residues) and average per-state accuracies \( (QH, QS, QC) \), number of correctly predicted residues in Helix, Strand and Coil divided by total number of residues in Helix, Strand and Coil respectively).

The performance of the prediction system on the training data is as follows: \( Q_{H} = 77.8\% \), \( Q_{S} = 73.3\% \), \( Q_{C} = 71.4\% \) and \( Q_{T} = 78.6\% \). On the 1295-protein test data, the performance of the prediction system is as follows: \( Q_{H} = 77.5\% \), \( Q_{S} = 73.6\% \), \( Q_{C} = 70.7\% \) and \( Q_{T} = 79.0\% \). Salamov and Solovyev [28] used sequence alignments and crisp nearest neighbor method for protein secondary structure prediction. Their results are: \( Q_{H} = 72.2\% \), \( Q_{S} = 72.4\% \) and \( Q_{C} = 52.2\% \). However, we caution the readers that the results may not be directly comparable, as we used a larger training and testing datasets, as well as a more recent and larger database \( (nr) \) to build the PSSM.

V. OTHER APPLICATIONS USING FUZZY SET THEORY

Fuzzy set theory and fuzzy logic, like many other computational intelligence methods, can be used for a wide range of bioinformatics problems. Here we highlight some of these applications.

First, we can apply fuzzy set theory and fuzzy logic in analyzing protein sequences. Using a k-nearest neighbor approach similar to the secondary structure prediction described above, a method was developed to predict solvent accessibility of each amino acid from a protein sequence [29]. FKNN algorithm has also been applied to predict a protein’s subcellular location [30], i.e., where the protein localizes in a cell (including extracellular, cytoplasm, nucleus, etc.). The study used dipeptide composition of a protein sequence and applied a class membership function similar to Equation 5. The overall prediction accuracy of about 80% was reported. In addition, fuzzy methods have been used for protein motif identification. A motif is a short sequence segment with (nearly) conserved amino acids among related proteins. Motifs typically represent important biological functions. Motifs sometimes are fuzzy or flexible, i.e., the conservations of amino acids do not have to be strict. Fuzzy logic was used to describe such flexibility of protein motifs in conjunction with neural networks [31]. In another study, researchers combined information theory with fuzzy logic search procedures to identify sequence motifs [32].
fuzzy clustering approach was also applied for revealing patterns of motifs [33]. For genomic sequences (DNA), a fuzzy inference engine based on information-theoretic considerations was developed to predict coding regions (the sequence segments that represent proteins) [34]. Researchers also used polynucleotides (words consisting of A, T, C, and G) as fuzzy sets and introduced a means of measuring dissimilarities between them as points in a hypercube [35, 36]. The method was used as a tool to compare different genomic sequences. Furthermore, fuzzy scoring functions based on diverse biological information (e.g., genome sequences, functional annotations and conservation across multiple genomes) were used to predict operon (a closely related group of neighboring genes on a DNA sequence), an important structure in bacterial genomes [37].

Fuzzy approaches are also suitable to describe and compare molecular structures. Fuzzy logic was employed in the analysis of a database of small molecular structures [38]. In particular, a fuzzy inference system was used to describe small molecule’s geometric surface that is essential for biochemical reactions, as the requirement for the geometric surface is not crisp. The study suggested a complicated interdependence among the constituted atoms in order to achieve fuzzy requirements of the geometric surface for biochemical reactions. The inference system was used for retrieving small molecules with similar structural features. Another method simplified flexible 3D chemical descriptions through clustering techniques and created "fuzzy" molecular representations called FEPOPS (feature point pharmacophores) [39]. The representations were used for flexible 3D similarity search given one or more active compounds without a priori knowledge of bioactive features. For protein structure comparison, a structure-alignment method was developed with a cost function containing both fuzzy assignment variables and atomic coordinates [40].

VI. DISCUSSIONS

Fuzzy set theory and fuzzy logic fit many bioinformatics problems for both representing the underlying biological mechanisms and applying fuzzy methods as techniques for analyses and predictions. Many biological properties and concepts are fundamentally fuzzy. Fuzzy methods do not require a complete underlying model, which is often the case for bioinformatics problems. The two main examples in this paper, as well as other fuzzy methods cited demonstrate the effectiveness of using fuzzy approaches in bioinformatics. In some of these cases, fuzzy models are more suitable for the biological problems. In other cases, both crisp and fuzzy approaches can apply. More systematic comparisons using sizable benchmarks to compare between crisp and fuzzy methods are needed. Although the fuzzy set theory and fuzzy logic have been applied in bioinformatics, given their potential, we expect to see more and more such applications in the future.

On the other hand, bioinformatics applications also raised new challenges for fuzzy set theory. For example, there are usually a large number of free parameters in many applications. How to systematically derive suitable parameters for computational models in biological systems is non-trivial. For crisp methods, this problem has been addressed to certain extent using methods such as orthogonal arrays [41]. The problem is less addressed for fuzzy models and fuzzy models often introduce more parameters to describe the fuzziness. Such a challenge calls for both theoretical developments in fuzzy set theory and novel combination of the biological knowledge for problem solving. Another challenge is that biologists often expect more than a fuzzy value as the overall assessment. Providing a more quantitative confidence assessment for prediction results based on the fuzzy evaluation is often important. In particular, instead of providing a fuzzy value ranging from 0 to 1, which may be hard to interpret, it would be useful to represent the value in terms of percentage of accuracy. Fuzzy probability theory can address this issue, but its application in bioinformatics has not been reported to our knowledge. Alternatively, in a particular bioinformatics application it may be practical through benchmarking the relationship between the fuzzy value and prediction accuracy.

A bigger challenge is the dimension of many bioinformatics problems, which is much larger than what fuzzy set theory typically addressed in the past. With the new concept of “systems biology”, a critical issue is how to effectively integrate various types of data, from sequence, gene expression, protein interaction to phenotypes, each being noisy and mono-perspective, to infer biological knowledge. Each set of these data often has at least thousands of dimensions. Applications of fuzzy approaches in these problems may require both more powerful computers and new frameworks in fuzzy set theory. Ultimately, while fuzzy set theory helps bioinformatics, biological questions will provide a driving force for new developments in fuzzy set theory itself.

ACKNOWLEDGEMENTS

Dong Xu and Rajkumar Bondugula have been supported by a Research Broad Grant at the University of Missouri). Mihail Popescu is supported by the National Library of Medicine Biomedical and Health Informatics Research Training grant 2-T15-LM07089-14.

REFERENCE:


