Mining Spatially Cohesive Itemsets in Protein Molecular Structures

Cheng Zhou[1,3], Pieter Meysman[1,2], Boris Cule[1], Kris Laukens[1,2], Bart Goethals[1]
[1]ADReM, University of Antwerp, Belgium
[2]Biomedical informatics research center Antwerpen (biomina), Belgium
[3]National University of Defense Technology, China
cheng.zhou@student.ua.ac.be
{pieter.meysman, boris.cule, kris.laukens, bart.goethals}@ua.ac.be

ABSTRACT

In this paper we present a cohesive structural itemset miner aiming to discover interesting patterns in a set of data objects within a multidimensional spatial structure by combining the cohesion and the support of the pattern. The usefulness of this algorithm is demonstrated by applying it to find interesting patterns of amino acids in spatial proximity within a set of proteins based on their atomic coordinates in the protein molecular structure. The experiments show that several patterns found by the cohesive structural itemset miner contain amino acids that frequently co-occur in the spatial structure, even if they are distant in the primary protein sequence and only brought together by protein folding. Further various indications were found that some of the discovered patterns seem to represent common underlying support structures within the proteins.

Keywords

itemset mining, multidimensional data, cohesion, protein structure

1. INTRODUCTION

Pattern discovery in sequences is a popular data mining task. Usually, a pattern is evaluated based on how close to each other its elements occur (cohesion), and how often the pattern itself occurs (support). Recently, attempts have been made to mine interesting patterns in sequences by combining cohesion and support [6]. Here we extend this method into data objects with a multidimensional structure and explore its potential to find interesting amino acid patterns within a set of proteins based on their atomic coordinates and molecular structure information.

Proteins are linear chains composed of twenty different amino acids (often referred to as 'residues'). In living cells these chains fold into specific three-dimensional structures that perform a great variety of biological functions. In the structure of a single protein we distinguish the primary structure, which corresponds to the sequence of the amino acids as they occur along the protein chain; the secondary structure, which is a local shape, such as α-helices or β-sheets, adopted by small segments of consecutive amino acids; the tertiary structure, which is the complete three-dimensional structure of the protein; and the quaternary structure, which corresponds to intermolecular interactions that proteins undergo. There is a vast amount of molecular structure data publicly available in biological databases. The RCSB Protein Data Bank (PDB), which is the single worldwide repository of molecular structures of large biological molecules currently contains the three-dimensional atomic coordinates of more than 90 000 structures [12]. Although the discovery of conserved structural motifs in proteins is a widely explored field in bioinformatics, the majority of protein pattern mining algorithms focus on the sequence dimension and do not consider the other spatial dimensions. The extraction of spatial patterns can potentially reveal significant biological insights into the properties of different proteins classes. The discovery of patterns within the tertiary structure of proteins unavoidably requires advanced computational algorithms due to its dimensionality. Here we explore the concept of cohesion for high dimensional itemset mining to extract sets of amino acids that frequently spatially co-occur in a given set of three-dimensional protein structures.

The concept of finding amino acids that are in close proximity to each other within a protein structure is somewhat similar to the purpose of protein contact maps. These maps are two-dimensional matrices detailing the pairwise inter-residue contacts of a protein, where a contact between two amino acids is defined if the distance between them is lower than a given threshold. The construction of such a contact map is a common step in the *ab initio* prediction of the full molecular structure of a protein from its sequence [19]. While itemset mining techniques have been successfully applied to such protein contact maps, the primary goal of these studies remained the improvement of *ab initio* prediction [10].

The goal of this paper is to explore whether cohesive structural itemset mining can reveal potentially interesting biological relationships. The type of interaction explored differs greatly from those contained within contact maps. Firstly, the presented algorithm directly mines the three-dimensional co-ordinates of the amino acids and thus suffers no loss of information due to a conversion to a two-dimensional space. Secondly, the recent development of the

BIOKDD'13 August 11, 2013, Chicago, IL, USA.
Copyright 2013 ACM 978-1-4503-2327-7 ...$15.00.

cohesion concept allows the algorithm to mine the data without setting a cut-off on the maximum distance in which relationships between amino acids can occur. This potentially allows the discovery of relationships where the amino acids are not in direct contact, such as, for example, residues forming a metal-binding site. Thirdly the application of itemset mining on the protein structure itself allows discovery of patterns that concern several amino acids, instead of the pairwise combinations of amino acids.

2. PROBLEM SETTING

We consider a data object with an n-dimensional structure as a list of points where a point v is a pair (a, c) consisting of an item $a \in I$ and a n-dimensional coordinate $c \in \mathbb{R}^n$, where I is the set of all possible items and $n \geq 1$. Clearly, two points can never occur at the same position, i.e. with the same coordinate. On the other hand, an item a_i may occur many times at different positions in a data object d_g. Thus there may be many points containing a_i in d_g and we denote such points as V_{gi}. Here, we denote a data object by $d = \{v_1, \dots, v_l\}$, where l is the number of points in the data object. A database DB is a set of data objects. The set of all data objects in DB is denoted by D.

The patterns considered in this paper are itemsets, or sets of items coming from the set I. The support count of an itemset is defined as the number of different data objects in which the itemset occurs, regardless of how many times the itemset occurs in any single data object. In other words, when looking for the support count of a single itemset, we can stop looking at a data object as soon as we have encountered the first occurrence of the itemset in that data object.

To determine the interestingness of an itemset, however, it is not enough to know how many times the items making up the itemset occur. In this paper, we are specifically investigating patterns of items occurring spatially in close proximity. To do this, we will define interesting itemsets in terms of both support and cohesion.

2.1 Support

For a given itemset X, we denote the set of data objects that contain all items of X as $N(X) = \{d \in D | \forall a \in X, \exists (a, c) \in d\}$. The *support* of X in all data objects D can now be defined as

$$S(X) = \frac{|N(X)|}{|D|}. \qquad (1)$$

2.2 Cohesive Radius

Given a set of points $V = v_1, \cdots, v_q$, let $MB(V)$ denote the ball with the smallest radius that contains V, namely the *smallest enclosing ball*. It has been shown that $MB(V)$ always exists and is unique [8]. Intuitively, we consider the points V in n-dimensional space cohesive if the radius of $MB(V)$ is small enough.

Given an itemset $X = \{a_1, \dots, a_m\}$, assume that each item a_i occurs n_i times in a given data object $d_g \in N(X)$. If we wish to find the exact smallest enclosing ball of X in d_g, there are $\prod_{i=1}^{m} n_i$ combinations for each of which we need to find a smallest enclosing ball, and then find the one with the minimal radius. This process is time consuming. Here, we approximate this process as following:

1. select an item a_1 from X, and for each point $v_j \in V_{g1},\ j = 1, 2, \dots, n_1$, we find a nearest point in each set of points of other items in X, namely $V_{g2}, \dots, V_{gm}$. We thus obtain the set of nearest points

$$NV_j = \{v | v = \underset{w \in V_{gi}}{\arg\min} D(w, v_j),\ i = 2, 3, \dots, m\}, \qquad (2)$$

where $D(w, v_j)$ is the Euclidean distance between point w and point v_j.

2. we get n_1 sets of m points from step 1. We denote $B_j = \{v_j\} \cup NV_j$ and $B = \{B_j | j = 1, 2, \dots, n_1\}$.

3. for each set $B_j \in B$, find $MB(B_j)$ and get its radius $R(X, v_j)$.

4. denote the smallest radius in a given data object $d_g \in N(X)$ as

$$R_g(X) = \min_{j=\{1,\dots,n_1\}} R(X, v_j). \qquad (3)$$

There are only n_1 smallest enclosing balls to find in each data object d_g, much fewer than if we tried to find the the exact smallest enclosing ball of X in d_g, resulting in a considerable reduction in time complexity.

In the worst case, the smallest radius we find could be nearly twice as large as the exact radius of the smallest enclosing ball containing items of an itemset X, as illustrated in Figure 1. In this simple two-dimensional example, assume we are evaluating itemset abc, and we picked item a as the starting point. We look for the nearest b and the nearest c, and find the only b, and c_1, which is closer to a than c_2, resulting in the ball drawn with a dashed line. However, the smallest possible ball containing a, b and c is much smaller, and is depicted using a solid line.

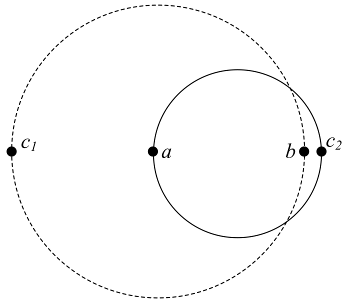

Figure 1: An example of the approximate computation of the smallest enclosing ball.

To evaluate the cohesion of an itemset X in the whole dataset, we need to compute the smallest radius $R_g(X)$ in each data object d_g that contains X. We define the *cohesive radius* of X in D as

$$R(X) = \frac{\sum_{d_g \in N(X)} R_g(X)}{|N(X)|}. \qquad (4)$$

Since we are using an average over a large number of data objects, the effect of the approximation will be amortised. For an itemset of size 2, we will always find the exact smallest ball, and for itemsets of size 3 or bigger, the chance of the worst-case error occurring (as described above) decreases as the size of the itemset grows. On one of the small datasets (*Winged*) we used in our experiments (see Section 4 for more details), it was possible to compute the exact smallest enclosing balls. We set the minimum support threshold to

0.8 and the maximum cohesive radius threshold to 4. Table 1 shows the average error made by our algorithm. The reported average was obtained by dividing the sum of all relative errors with the total number of the computed smallest balls. As can be seen in the run-times reported in Table 1, the complexity of the exact algorithm is prohibitive on large datasets, while the average error of the approximate algorithm is kept within reasonable limits. In this small example, we can see that we miss out on less than 4% of the patterns we would discover using the exact method, which would take nearly 5 000 times longer to complete the search.

Table 1: The comparison of our approximate method with the exact computation of the smallest enclosing ball on a small dataset.

Method	Output size	Runtime	Average error
Approximate	158	1.160s	0.01855
Exact	164	5755.443s	0

2.3 Interesting Itemset

Given a minimum support threshold *min_sup* and a maximum cohesive radius threshold *max_rad*, X is an *interesting itemset* if $S(X) \geq min\_sup$ (X is frequent) and $R(X) \leq max\_rad$ (X is cohesive). Note that the smaller the radius $R(X)$ the higher the cohesion of X.

3. GENERATING THE COMPLETE SET OF INTERESTING ITEMSETS

In this section we present an algorithm for mining interesting itemsets in a database consisting of data objects, each of which containing a number of multidimensional points. Note that the cohesive radius of an itemset is not a monotonic measure. In other words, in rare cases, it is possible for the cohesive radius of a smaller itemset to be greater than the cohesive radius of one of its supersets. Consider the following simple example. Assume that the dataset consists of just three data objects, d_1 and d_2, containing items a, b and c, and d_3, containing only items a and b. It is perfectly possible that the radii of the smallest balls containing itemset *abc* in both d_1 and d_2 are smaller than the radius of the smallest ball containing itemset *ab* in d_3. In this case, $R(abc)$ (the cohesive radius of itemset *abc*, as defined in Equation 4) will be smaller than $R(ab)$, even though *ab* is a subset of *abc*.

Although the cohesive radius of an itemset is not monotonic, we can still use it for prunning certain candidates from the search space. Our pruning method is based on two observations:

1. If itemset X is a subset of itemset Y, and they both occur in a data object d_i, then $R_i(X) \leq R_i(Y)$.

2. Given a minimum support threshold *min_sup*, an itemset must occur in at least $\lceil min\_sup \times |D| \rceil$ data objects to be frequent. Assume that itemset X occurs in k data objects, with $k \geq \lceil min\_sup \times |D| \rceil$, and sort these data objects such that $R_1(X) \leq \ldots \leq R_k(X)$. For any frequent itemset Y that is a superset of X, it holds that

$$R(Y) \geq \frac{\sum_{i=1,\ldots,\lceil min\_sup \times |D| \rceil} R_i(X)}{\lceil min\_sup \times |D| \rceil} = LBR(X).$$

In other words, $LBR(X)$ as defined above, can serve as a lower bound for the cohesive radius of all frequent supersets of X. As a result, if X is frequent, but its cohesive radius is large enough, we can be sure that none of its supersets can be both frequent and cohesive.

Therefore, our algorithm generates all interesting itemsets in two steps. In the first step, we use an Apriori-like algorithm to find the frequent itemsets. In the second step, we determine which of the frequent itemsets are actually spatially cohesive and utilise the two observations above to prune the itemsets that cannot be both frequent and cohesive.

Let *n-itemset* denote an itemset of size n. Let F_n denote the set of frequent *n-itemset*s. Let C_n be the set of candidate *n-itemset*s and T_n be the set of interesting *n-itemset*s. The algorithm for generating the complete set of interesting itemsets in a given set of data objects D is shown in Algorithm 1. Two optional parameters, *min_size* and *max_size*, can be used to limit the output only to interesting itemsets with a size bigger than or equal to *min_size* and smaller than or equal to *max_size*.

Algorithm 1: GENERATINGITEMSETS. An algorithm for generating all interesting itemsets in a dataset D.

```
input : dataset $D$, minimum support threshold
        $min\_sup$, maximum cohesive radius threshold
        $max\_rad$, minimum size constraint $min\_size$
        and maximum size constraint $max\_size$
output: all interesting itemsets $T$
1  $C_1 = \{a | a \in I\}$, $I$ is the set of all items occurring in $D$;
2  $F_1 = \{f | f \in C_1, S(f) \geq min\_sup\}$;
3  if $1 \geq min\_size$ then
4      $T_1 = F_1$ ;
5  $C_1 = F_1$;
6  $n = 2$;
7  while $C_{n-1} \neq \emptyset$ and $n \leq max\_size$ do
8      $T_n = \emptyset$;
9      $C_n$ = candidateGen($C_{n-1}$);
10     $F_n = \{f | f \in C_n, S(f) \geq min\_sup\}$;
11     $C_n = \emptyset$;
12     foreach frequent itemset $f$ in $F_n$ do
13         if $LBR(f) \leq max\_rad$ then
14             $C_n = C_n \cup \{f\}$ ;
15             if $n \geq min\_size$ and $R(f) \leq max\_rad$ then
16                 $T_n = T_n \cup \{f\}$ ;
17     $n++$;
18 $T = \bigcup_{i=1}^{n-1} T_i$;
19 return $T$;
```

Lines 1-4 count the supports of all the items to determine the interesting 1-itemsets. Lines 6-19 discover all interesting itemsets of different sizes n ($max\_size \geq n \geq 2$). First, the already discovered candidates of size $n-1$ (C_{n-1}) are used to generate the candidate itemsets C_n using the candidateGen function (line 9). The candidateGen function is similar to the function Apriori-gen in the Apriori algorithm [1]. In line 10, we store the frequent itemsets from C_n into F_n. In lines 13-14, we prune the candidates that cannot be both frequent and cohesive, while in lines 15-16, we store the interesting itemsets (as defined in Section 2) from F_n into T_n. The final set of all interesting itemsets in D is stored in T and produced as output.

The two most time consuming steps are the candidate generation and the evaluation of the cohesive radius. For these two steps we use the Apriori algorithm [1] to generate candidates, and an existing implementation[1] of the algorithm for computing the smallest enclosing ball [8], respectively. The time complexity of these algorithms has been extensively analysed in the papers that originally proposed them. Since the smallest enclosing ball must be computed only for itemsets that have been found to be frequent, the runtime will be proportional to the number of generated candidate itemsets.

4. EXPERIMENTS

The cohesive structural itemset miner was applied to extract patterns from a real biological dataset, namely protein molecular structures. The structural information on these proteins was extracted from the PDB public archive [12]. PDB contains the atomic coordinates and molecular structure information for various proteins and other biological macromolecules. The relative locations of the atoms to each other within these molecules were determined by a variety of methods, such as X-ray crystallography, NMR spectroscopy and cryo-electron microscopy. These three-dimensional coordinates of the amino acids of a set of related proteins will make up the backbone of our analysis.

For the purposes of applying the methodology on a wide range of data, four sets of proteins were collected. Two smaller datasets consisted of the proteins annotated by SCOP as containing respectively 'winged helix DNA-binding domain' (*Winged*) or a 'lambda repressor-like DNA-binding domain' (*Lambda*) [2]. As an additional constraint on this smaller dataset, only structures reporting both the protein and the DNA structure were utilised. Thus only proteins known to be in their active and bound state are considered during the rule mining as the free-floating potential inactive state may display considerable differences in its conformation. This approach guarantees the uniformity of the structures to evaluate in these datasets. Two larger sets were based on the molecular function of the protein. To this end, using their gene ontology molecular function annotations, one set of proteins with 'kinase activity' (*Kinase*) and another set with 'peptidase activity' (*Peptidase*) were collected [4]. These datasets therefore represent a wide diversity of proteins that each share a common molecular function. In cases where multiple macromolecules were present in the same PDB entry, only one protein was presented to the algorithm, i.e., the one with a description matching certain keywords (e.g., trypsin or protease for the peptidase set) or the protein with a description similar to the title of the stored structure. In cases of ambiguity (e.g., for k-mer proteins), the first reported protein matching the above criteria was selected.

From the reported protein molecular structure only the position of the α-carbon atom of the amino acid was considered. This atom is present in every amino acid and is the carrier of the side chain unique to each type of amino acid. Each Cα was then labelled with the three-letter name of the corresponding amino acid. This label was further extended with the secondary structure information, which is also included in most PDB structures. The secondary structure concerns the local shape of the amino acids, and a collection of residues within a single protein can form an α-helix (denoted in the itemsets as X_H), a β-sheet (X_B) or a loop of unstructured amino acids (X_U). The input data thus consisted of the (x, y, z) coordinates of the Cα atom labelled by the corresponding amino acid and the secondary structure. In this manner, a protein is converted to a list of points where a point v is a pair (a, c) consisting of the label $a \in I$ and a three-dimensional coordinate $c \in \mathbb{R}^3$, where I is the set of all possible labels (in our case, amino acids). The algorithm as presented in Section 3 was then used to generate the interesting itemsets found across these proteins, with each itemset representing a pattern of spatially co-occurring amino acids.

Table 2 shows the run-times of our algorithm on the four datasets with *min_sup* fixed at 0.8, *max_rad* fixed at 4, *min_size* set to 1 and *max_size* unlimited. All experiments are performed on a laptop computer with Intel i7 (2 CPUs 2.7GHz), 4GB memory and Windows 7 Professional. From the table, we can see that the run-time largely depends on the number of proteins in the dataset and the number of candidate itemsets. This matches the conclusions of the time complexity analysis performed in Section 3.

Table 2: Run-times of the algorithm on 4 datasets. The second column contains the number of proteins in the datasets, while $|C|$ denotes the number of generated candidates.

Dataset	Num of proteins	$\|C\|$	Runtime
Lambda	47	569	2.924s
Winged	62	235	1.160s
Kinase	2749	766	249.892s
Peptidase	2558	415	96.715s

4.1 Lambda Repressor-like Proteins

The first small dataset the algorithm was applied to consists of 47 proteins annotated with a lambda repressor-like DNA-binding domain. This set therefore consists mostly of transcription factors, which are DNA-binding proteins that regulate the expression of downstream genes. The archetypical protein for this type of domain is the bacteriophage lambda C1 repressor, which is a viral regulator [5]. Several proteins containing a lambda repressor-like domain are of great biological importance and the mechanism by which such proteins interact with the DNA molecule are well understood. For example, the lactose repressor (LacI) is commonly used as a model for transcriptional regulation and the interaction between LacI and its binding sites has been the subject of intensive study over the past several decades [11]. The typical lambda repressor-like domain consists of four α-helices in a closed leaf motif. This protein dataset is therefore an ideal case study to evaluate if the patterns uncovered through the presented methodology can be related to known biological significance.

The cohesive structural itemset miner was applied to these protein structures to find amino acids that were consistently grouped in close proximity across a large fraction of the proteins. The reported patterns were filtered based on their uniqueness to a specific dataset at a support cut-off of 80%. The cohesive radius threshold was set to 4, *min_size* to 3, and *max_size* to 6. The most cohesive patterns specific for the lambda repressor-like proteins can be found in Table 3.

[1] http://www.inf.ethz.ch/personal/gaertner/miniball.html

A total of 171 patterns were found within the set thresholds, of which 160 were itemsets containing three amino acids, whie the other 11 contained four amino acids. No itemsets of size 5 or more were found. This is likely due to the trade-off between adding additional amino acids to the itemset and a resulting decrease in cohesion and frequency of the pattern. Indeed, due to the steric constraints of amino acid placement, one can expect that adding a single amino acid would have a great effect on the cohesive radius of any pattern.

It is apparent from the labels of the extracted itemsets that most describe amino acids in α-helices. This can be expected as the annotated domain used to create this dataset consisted mostly of α-helices. Also amino acids within a single α-helix can be expected to be frequently co-occurring. However, a comparison between the itemsets and an alignment of the sequences reveals that not all patterns are limited to the conserved region between these proteins. In the next step, the locations of the itemsets within the protein structure are visualised to give an overview of their distribution throughout the structure.

Table 3: The 30 most cohesive patterns out of 171 total patterns extracted with the cohesive structural miner from the molecular structures of the proteins annotated with a lambda repressor-like domain. Each itemset consists of a set of amino acids found to be in close proximity of each other in high frequency within the group of studied proteins. Provided are the cohesion radius in angstrom and the itemset frequency in the dataset.

Itemset	Cohesion radius	Frequency
ARG_H GLU_H ILE_H	2.81	0.8
ALA_H GLU_H VAL_H	2.87	0.93
ARG_H ALA_H PHE_H	2.88	0.82
MET_H ALA_H LEU_H	2.88	0.93
ALA_H GLU_H LYS_H	2.89	0.93
ALA_H GLU_H ASP_H	2.91	0.93
ARG_H GLU_H LYS_H	2.92	0.93
GLU_H LYS_H VAL_H	2.92	0.93
ALA_H LYS_H VAL_H	2.92	0.93
ARG_H ALA_H LYS_H	2.93	0.97
ARG_H ALA_H GLU_H	2.95	0.93
ALA_H LEU_H GLU_H	2.96	0.93
ALA_H LEU_H GLY_H	2.99	0.93
ALA_H LEU_H VAL_H	3.02	0.93
ALA_H GLU_H ILE_H	3.04	0.8
ARG_H ASN_H VAL_H	3.05	0.91
ARG_H VAL_H SER_H	3.07	0.93
ARG_H ALA_H VAL_H	3.08	0.93
ALA_H VAL_H ILE_H	3.11	0.8
ARG_H ALA_H LEU_H	3.12	0.97
ARG_H GLU_H THR_H	3.13	0.91
ALA_H LEU_H PHE_H	3.14	0.82
ALA_H LYS_H ILE_H	3.17	0.85
ALA_H VAL_H ASP_H	3.17	0.93
ALA_H LEU_H TYR_H	3.19	0.93
GLU_H VAL_H ILE_H	3.19	0.8
ARG_H LEU_H ILE_H	3.2	0.85
ALA_H GLU_H THR_H	3.24	0.91
ALA_H VAL_H SER_H	3.24	0.93
ARG_H ALA_H GLN_H	3.27	0.97

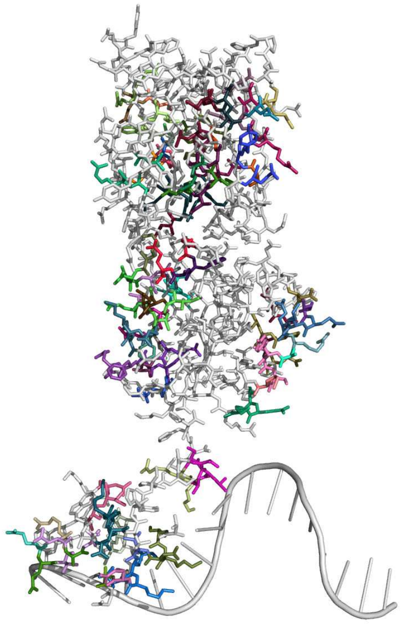

Figure 2: The molecular structure of the *E. coli* PurR transcription factor (as reported by PDB 1PNR) plotted using the open source version of Pymol. Note that the reported structure in the PDB file only contained one side of the symmetrical protein-DNA complex and thus only features one protein within the protein complex and one DNA strand of the DNA-helix. The atoms of the protein are presented in the stick representation while those of the DNA molecule are reduced to a cartoon representation. The amino acids matching the patterns extracted for the lambda repressor-like domain proteins are provided in a colour corresponding to the amino acid content of the pattern, while amino acids not part of any pattern are given in grey.

Figure 2 shows the protein structure of the *Escherichia coli* PurR repressor (from PDB 1PNR) where the amino acids matching the discovered 171 patterns are marked. This protein is a bacterial regulator of purine metabolism and is part of the LacI-GalR protein family. This transcription factor is annotated as containing a similar DNA-binding domain as the Lambda C1 repressor on the N-terminal domain, except that it is missing the first α-helix. It also displays a C-terminal domain with a ligand-binding and dimerisation motif similar to the ligand binding sites of periplasmic sugar-binding proteins. The two domains are connected with a hinge sequence that also contains several functional residues for DNA-binding. For example, the leucine present at position 54 in the hinge helix is known to intercalate into the DNA molecule during complex formation causing the induction of a DNA bend [3]. As can be seen in Figure 2, several patterns match amino acids that form the DNA-binding domain. Additionally there are other patterns that are present in the C-terminal domain of the protein or as part of the hinge helix. Inside the hinge helix, most of the amino acids matched up to one or more of the discovered patterns. Several of these patterns include the intercalating leucine residue, such as the pattern ARG_H, ALA_H, LEU_H and VAL_H (i.e. the combination of arginine, alanine, leucine and valine in a helix conformation). As not all lambda repressor-like proteins contain the hinge helix, it is interesting that so many patterns are still found within this segment. Within the DNA-binding domain, there is a notable lack of the central threonine (THR_{16}) residue in any pattern, most likely because this amino acid is missing in several members of the LacI-GalR family. The presence or the absence of threonine at this position in the protein has been shown to confer differential specificity between LacI-GalR proteins to their DNA targets [14]. Similar findings could be observed for the other proteins within this dataset. Most patterns do not match the amino acids specific for a single protein, which, for example, confer the DNA-binding specificity, but instead match 'supporting' amino acids which seem to be necessary for the overall protein structure and the presentation of the specific residues to the ligands that can be bound by the protein.

4.2 Winged Helix Proteins

The second small dataset contains 62 proteins annotated with a winged-helix DNA-binding domain. The winged-helix domains typically consist of three α-helices, three β-strands forming a twisted antiparallel β-sheet and two large loops or 'wings' [7]. While most proteins present in this set are transcription factors, this set also includes DNA replication initiation proteins (e.g., the F plasmid RepE: PDB 2Z9O), helicases (e.g., *Archaeoglobus fulgidus* Hel308: PDB 2P6R) and endonucleases (e.g., *Planomicrobium okeanokoites* FokI: PDB 1FOK). Thus while these proteins share significant structural similarity, their molecular function is very divergent. In this experiment, the support threshold was set to 80%, *max_rad* to 5, *min_size* to 3, and *max_size* to 6. The application of the presented algorithm to this dataset then resulted in 133 patterns, of which all but five consisted of three amino acids and the remainder of four amino acids. The most cohesive patterns for this dataset can be found in Table 4. As was reported for the lambda repressor-like proteins, many of the patterns include amino acids contained within α-helices. Comparison with sequence alignment of the proteins reveals that while several patterns are derived from the the α-helices present in the winged-helix domain, the majority of the patterns occur in other segments of the protein.

Figure 3 shows the molecular structure of the *E. coli* CRP protein, a transcription factor with a winged helix domain present in the training data. The CRP transcription factor usually binds DNA as a protein complex with two copies of the CRP protein and is known to regulate more than 180 genes, mostly those associated with the carbon metabolism, in *E. coli.* The CRP protein consists of a C-terminal DNA-binding domain containing the winged helix motif and an N-terminal dimerisation domain consisting of β-sheets and a long α-helix. This α-helix is critical for the conformational changes resulting in the activation of CRP induced upon the binding of its ligand, cAMP [18]. The patterns extracted for the entire winged helix protein set concern the amino acids that make up the DNA-binding domain and those contained within the long α-helix directed towards the dimerisation interaction region.

The results for the winged helix proteins with different molecular functions are very similar to those reported above for the CRP protein. The RepE protein involved in the replication initiation of the F plasmid, is known to contain two winged helix domains: one at the N-terminal side of the protein and the other at the C-terminal side. These two domains are separated by a linker region, which accepts a conformational change necessary for dimerisation of RepE [15]. Amino acids present in the winged helix domain and the linker domains match various patterns found in the entire dataset. Several of these patterns, such as ARG_H LEU_H LYS_H (i.e., Arginine, Leucine and Lysine in α-helix conformation), match the LEU_{39} residue of the RepE which is not part of the dimerisation interface but has been postulated to aid in the correct placement of an α-helix necessary to stabilise the protein dimer [15]. As can be seen in Table 4, the majority of the patterns found for the winged helix proteins contain a leucine amino acid. Given that several leucine residues in RepE act as 'scaffold' amino acids to stabilise the dimer conformation, it seems likely that at least some of the leucine residues within these itemsets perform a similar function in a number of the winged helix domain proteins.

Table 4: The 30 most cohesive patterns out of the 133 patterns extracted with the cohesive structural itemset miner form the molecular structure of the proteins annotated to contain a winged-helix domain. Each pattern consists of a set of amino acids found to be in close proximity in high frequency within the studied proteins. Provided are the cohesion radius in angstrom and the itemset frequency in the dataset.

Itemset	Cohesion radius	Frequency
LEU_H ARG_H ILE_H	3.27	0.88
LEU_H ARG_H SER_H	3.52	0.98
LEU_H ALA_H ARG_H	3.57	0.98
LEU_H ALA_H SER_H	3.63	0.96
LEU_H ALA_H ILE_H	3.72	0.87
LEU_H VAL_H ARG_H	3.75	0.93
LEU_H GLU_H ILE_H	3.78	0.88
LEU_H LYS_H ARG_H	3.84	1
ALA_H ARG_H ILE_H	3.86	0.87
LEU_H VAL_H SER_H	3.86	0.93
LEU_H VAL_H ALA_H	3.88	0.91
LEU_H ARG_H TYR_H	3.89	0.87
LEU_H ARG_H ASN_H	3.89	0.9
LEU_H VAL_H ILE_H	3.89	0.82
LEU_H GLN_H GLU_H	3.9	0.96
LEU_H PHE_H LYS_H	3.92	0.85
LEU_H VAL_H LYS_H	3.93	0.93
LEU_H ALA_H LYS_H	3.97	0.98
LEU_H ALA_H ASN_H	3.98	0.88
LEU_H LYS_H GLY_H	3.99	0.85
LEU_H PHE_H GLU_H	4	0.85
LEU_H VAL_H ASN_H	4	0.83
LEU_H THR_H ILE_H	4.02	0.85
LEU_H THR_H ARG_H	4.03	0.91
ALA_H ARG_H SER_H	4.04	0.96
GLU_H VAL_H LYS_H	4.05	0.93
LEU_H GLU_H VAL_H	4.06	0.93
LEU_H PHE_H ALA_H	4.07	0.85
LEU_H THR_H ALA_H	4.1	0.9
LEU_H GLU_H SER_H	4.13	0.98

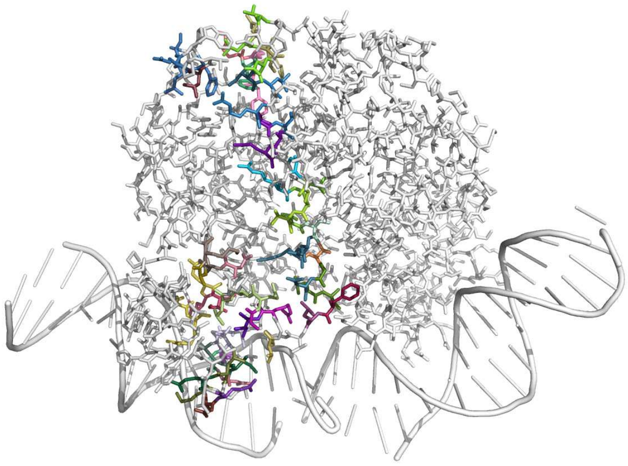

Figure 3: The molecular structure of the *E. coli* CRP transcription factor bound to its operator site (as reported by PDB 1O3T) plotted using the open source version of Pymol. Only one of the two copies forming the protein complex was presented to the cohesive structural itemset miner, namely the one to the left in this figure. The atoms of the protein are presented in the stick representation while those of the DNA molecule are reduced to a cartoon representation. The amino acids matching the patterns extracted for the winged helix domain proteins are provided in a colour corresponding to the amino acid content of the itemset, while amino acids that do not match any pattern are given in light grey. The protein for which no patterns were extracted is presented in white.

Indeed, this corresponds to the results for the CRP protein where the occurrences of the pattern seemed to concern the amino acids responsible for the stabilisation of the dimer structure.

4.3 Kinase Proteins

The first of the larger datasets consists of 2749 proteins displaying kinase activity. These are proteins that catalyse a chemical reaction that transfers a phosphate group to a substrate, a process termed phosphorylation. This substrate is most commonly another protein and phosphorylation may cause conformation change in the substrate protein, for example, causing it to switch from an inactive to an active state. Based on their protein structures and substrates specificity, kinases are divided into the ‘protein kinase-like superfamily’ and then a set of ‘atypical kinases’ whose structures greatly differ and can be further subdivided according to common domains [17]. The typical protein kinases share a common catalytic segment consisting of an N-terminal subdomain of mostly β-sheets and a C-terminal subdomain with mostly α-helices. Using a support threshold of 80%, *max_rad* equal to 4, *min_size* equal to 3, and *max_size* equal to 6, the cohesion-based structural miner resulted in a set of 60 patterns consisting of three amino acids in close proximity. The majority of the patterns consist of residues within α-helices. Furthermore, in several proteins, these patterns could be directly related to the catalytic regions of the kinase.

An example of a typical protein kinase within our dataset is the *Saccharomyces cerevisiae* MAP kinase, Fus3, which forms an essential part of the mating signalling pathway in yeast. The protein structure contains a C-terminal and an N-terminal region connected by a short hinge section. The catalytic loop containing the functional amino acids for the phosphorylation is contained within the N-terminal region [16]. Several patterns were found to describe residues within the catalytic loop of Fus3. These include a pattern describing the amino acids SER_{141} and LYS_{139} within the catalytic loop, and LEU_{100}, which is part of a neighbouring α-helix. The SER_{141} and LEU_{100} residues occur together in these patterns as the spatial distance between their Cα only spanned 5.8 angstroms (according to the structure contained within PDB 2F49) which is found to be sufficiently cohesive by our algorithm (note that a distance of 5.8 angstroms easily fits into a ball with a radius smaller than 4).

4.4 Peptidase Proteins

A set of 2558 proteins with peptidase activity makes up the final dataset for this analysis. These proteins catalyse a reaction to break up the covalent bonds between peptides. Many of these proteins are therefore involved in the degradation of cellular proteins. There is a great deal of variety in the molecular structure of these proteins as many types of enzymes display peptidase activity. Using a support threshold of 80%, *max_rad* equal to 4.5, *min_size* equal to 3, and *max_size* equal to 6, a total of 144 patterns were discovered in this dataset and each of these consists of three amino acids. However, in contrast to the previous analyses, the patterns mostly concern amino acids in unstructured regions of the proteins. This is not unsurprising as α-helices are not as prevalent in peptidase proteins as they are in DNA-binding proteins or kinases. Due to the intrinsic diversity of the peptidase dataset, the same patterns are derived from amino acids present in very different domains in different proteins.

An example of a peptidase from this dataset, the *E. coli* PepP is shown in Figure 4. The PepP protein is an exopeptidase that cleaves the N-terminal residue from polypeptides. The centre of the protein contains two metal-binding sites, which catalyse the cleavage reaction [13]. Within the PepP protein, five amino acids are known to function as metal-binding residues and two histidine residues are known to be essential for the catalytic activity [9]. Interestingly, several of the peptidase patterns were found in the neighbourhood of the catalytic site. Similar to the findings in the previous analysis, the patterns do not always contain the known functional residues themselves but instead match the amino acids that make up the strand carrying the residue. This indicates that the cohesive patterns do not consist of the amino acids that provide the target specificity but instead correspond to the common residues that stabilise their location. Indeed, several itemsets are found to span different strands that form the metal-binding region. For example, the amino

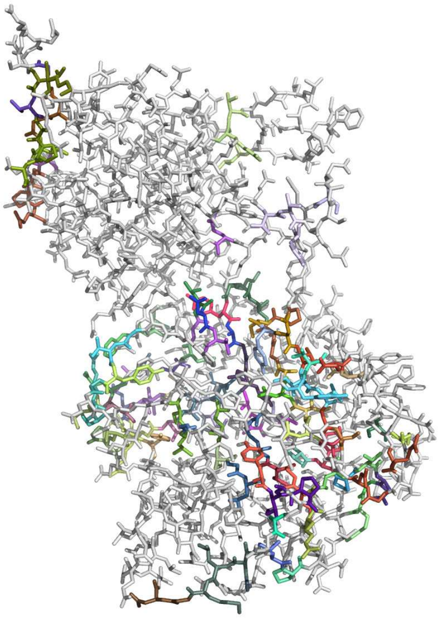

Figure 4: The molecular structure of the *E. coli* PepP aminopeptidase in monomer form (as reported by PDB 1A16) plotted using the open source version of Pymol. The amino acids matching the patterns extracted for the peptidase proteins are provided in a colour corresponding to the amino acid content of the itemset, while amino acids not included in any pattern are given in grey.

acids within the rule SER_U ALA_U GLY_U (i.e., Serine, Alanine and Glycine in unstructured regions) match residues 228, 269 and 270 respectively. This is a distance of more than 40 residues within the sequence, but the protein folding has brought the αC of these residues to within 5 angstroms. Indeed both these strands form a loop along the centre of the metal-binding site. Furthermore, the strand containing ALA_{269} and GLY_{270} also contains the metal-binding residue ASP_{272}.

5. CONCLUSIONS

In this paper, we have presented a novel method to mine frequent cohesive itemsets in multidimensional data. This algorithm was applied to datasets containing the full atomic coordinates of various proteins. We were able to successfully identify sets of amino acids that frequently occur in close proximity to each other throughout the given proteins. Thorough analysis revealed that the patterns did indeed reflect amino acids that could span distances in the primary sequence of the protein but were brought together through the protein folding. Furthermore, the types of patterns that we found in the current setting mostly seem to reflect amino acids with a supporting role to the overall or specific structure of the protein.

6. ACKNOWLEDGMENTS

Cheng Zhou is financially supported by the China Scholarship Council (CSC). This work was supported by the Fund for Scientific Research - Flanders (FWO- Vlaanderen) project "Evolving graph patterns".

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *VLDB'94*, pages 487–499. Morgan Kaufmann Publishers, 1994.

[2] A. Andreeva, D. Howorth, J.-M. Chandonia, S. E. Brenner, T. J. P. Hubbard, C. Chothia, and A. G. Murzin. Data growth and its impact on the SCOP database: new developments. *Nucleic acids research*, 36(Database issue):D419–25, Jan. 2008.

[3] D. N. Arvidson, F. Lu, C. Faber, H. Zalkin, and R. G. Brennan. The structure of PurR mutant L54M shows an alternative route to DNA kinking. *Nature structural biology*, 5(6):436–41, June 1998.

[4] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature genetics*, 25(1):25–9, May 2000.

[5] C. E. Bell, P. Frescura, A. Hochschild, and M. Lewis. Crystal structure of the lambda repressor C-terminal domain provides a model for cooperative operator binding. *Cell*, 101(7):801–11, June 2000.

[6] B. Cule, B. Goethals, and C. Robardet. A new constraint for mining sets in sequences. In *SDM'09*, pages 317–328, 2009.

[7] K. S. Gajiwala and S. K. Burley. Winged helix proteins. *Current Opinion in Structural Biology*, 10(1):110–116, Feb. 2000.

[8] B. Gärtner. Fast and robust smallest enclosing balls. In *Algorithms-ESA'99*, pages 325–338. Springer, 1999.

[9] S. C. Graham, P. E. Lilley, M. Lee, P. M. Schaeffer, A. V. Kralicek, N. E. Dixon, and J. M. Guss. Kinetic and crystallographic analysis of mutant Escherichia coli aminopeptidase P: insights into substrate recognition and the mechanism of catalysis. *Biochemistry*, 45(3):964–75, Jan. 2006.

[10] J. Hu, X. Shen, Y. Shao, C. Bystroff, and M. J. Zaki. Mining protein contact maps. In *2nd BIOKDD workshop on data mining in bioinformatics.*, 2002.

[11] C. G. Kalodimos, R. Boelens, and R. Kaptein. Toward an integrated model of protein-DNA recognition as inferred from NMR studies on the Lac repressor system. *Chemical reviews*, 104(8):3567–86, Aug. 2004.

[12] A. Kouranov, L. Xie, J. de la Cruz, L. Chen, J. Westbrook, P. E. Bourne, and H. M. Berman. The RCSB PDB information portal for structural genomics. *Nucleic acids research*, 34(Database issue):D302–5, Jan. 2006.

[13] W. T. Lowther and B. W. Matthews. Metalloaminopeptidases: Common Functional Themes in Disparate Structural Surroundings. *Chemical Reviews*, 102(12):4581–4608, Dec. 2002.

[14] P. Meysman, K. Marchal, and K. Engelen. Identifying common structural DNA properties in transcription factor binding site sets of the LacI-GalR family. *Current bioinformatics*, 8(4), 2013.

[15] A. Nakamura, C. Wada, and K. Miki. Structural basis for regulation of bifunctional roles in replication initiator protein. *Proceedings of the National Academy of Sciences of the United States of America*, 104(47):18484–9, Nov. 2007.

[16] A. Reményi, M. C. Good, R. P. Bhattacharyya, and W. A. Lim. The role of docking interactions in mediating signaling input, output, and discrimination in the yeast MAPK network. *Molecular cell*, 20(6):951–62, Dec. 2005.

[17] E. D. Scheeff and P. E. Bourne. Structural evolution of the protein kinase-like superfamily. *PLoS computational biology*, 1(5):e49, Oct. 2005.

[18] H. Sharma, S. Yu, J. Kong, J. Wang, and T. A. Steitz. Structure of apo-CAP reveals that large conformational changes are necessary for DNA binding. *Proceedings of the National Academy of Sciences of the United States of America*, 106(39):16604–9, Sept. 2009.

[19] M. Vendruscolo, E. Kussell, and E. Domany. Recovery of protein structure from contact maps. *Folding and Design*, 2(5):295–306, Oct. 1997.