CHEMICAL COMPOUND CLASSIFICATION WITH AUTOMATICALLY MINED STRUCTURE PATTERNS

[1]A. M. SMALTER AND [1]J. HUAN AND [2]G. H. LUSHINGTON
[1]*Department of Electrical Engineering and Computer Science*
[2]*Molecular Graphics and Modeling Laboratory*
University of Kansas, Lawrence, KS 66045, USA
E-mail: {asmalter, jhuan, glushington}@ku.edu

In this paper we propose new methods of chemical structure classification based on the integration of graph database mining from data mining and graph kernel functions from machine learning. In our method, we first identify a set of general graph patterns in chemical structure data. These patterns are then used to augment a graph kernel function that calculates the pairwise similarity between molecules. The obtained similarity matrix is used as input to classify chemical compounds via a kernel machines such as the support vector machine (SVM). Our results indicate that the use of a pattern-based approach to graph similarity yields performance profiles comparable to, and sometimes exceeding that of the existing state-of-the-art approaches. In addition, the identification of highly discriminative patterns for activity classification provides evidence that our methods can make generalizations about a compound's function given its chemical structure. While we evaluated our methods on molecular structures, these methods are designed to operate on general graph data and hence could easily be applied to other domains in bioinformatics.

1. Introduction

The development of accurate models for chemical activity prediction has a range of applications. They are especially useful in the screening of potential drug candidates, currently a difficult and expensive process that can benefit enormously from accurate *in silico* methods. These models have proved difficult to design, due to the complex nature of most biological classification problems. For example, the toxicity of a particular chemical compound is determined by a large variety of factors, as there are innumerable ways that a foreign chemical might interfere with an organism, and the situation is further complicated by the possibility that a benign chemical may be broken down into toxic metabolites *in vivo*. Clearly, there is no single set of chemical features that can be easily applied to to all problems in all situations, and therefore the ability to isolate problem-specific chemical features from broader data collections is a critical issue.

Here we address the problem of identifying structure characteristics that link a chemical compound to its function by integrating graph database mining methods from the data mining community with graph kernel functions from the machine learning community. Graphs are powerful mathematical structures and have been widely used in bioinformatics and other research [1], and are ubiquitous in the representation of chemical compounds. In our method, we identify frequently occurring subgraphs from a group of chemical struc-

tures represented as graphs, and define a graph similarity measure based on the obtained subgraphs. We then build a model to predict the function of a chemical structure based on the previously generated similarity measures.

Traditional approaches to graph similarity rely on the comparison of compounds using a variety of molecular attributes known *a priori* to be involved in the activity of interest. Such methods are problem-specific, however, and provide little assistance when the relevant descriptors are not known in advance. Additionally, these methods lack the ability to provide explanatory information regarding what structural features contribute to the observed chemical activity. Our proposed method alleviates both of these issues through the mining and analysis of structural patterns present in the data in order to identify highly discriminating patterns, which then augment a graph kernel function that computes molecular similarity.

We have applied our methods to three chemical structure-activity benchmarks: predictive toxicology, human intestinal absorption, and virtual screening. Our results indicate that the use of a pattern-based approach to graph similarity yields performance profiles comparable to, and sometimes exceeding that of previous non-pattern-based approaches. In addition, the presence and identification of highly discriminative patterns for chemical activity classification provides evidence that our methods can make generalizations about a compound's function given its chemical structure.

The rest of the paper is organized in the following way. In Section 2, we present an overview of related work on graph kernels and frequent subgraph mining. In Section 3, we present background information about graph representation of chemical structures. In Section 4, we present the algorithmic details of the work and in Section 5, we present our empirical study of the proposed algorithm using several chemical structure benchmarks. We conclude our paper with a short discussion about pros and cons of our proposed methods.

2. Related Work

The term kernel function refers to an operation for computing the inner product between two vectors in a feature space, thus avoiding the explicit computation of coordinates in that feature space. Graph kernel functions are simply kernel functions that have been defined to compute the similarity between two graph structures. In recent years a variety of graph kernel functions have been developed, with promising results as described by Ralaviola et al[2]. Here we review the two methods that are most similar to ours. The first compares graphs using random, linear substructures; and the second is based on matching and aligning the vertices of two graphs. We also review the technique used to identify substructure patterns in our proposed method.

2.1. *Marginalized and Optimal Assignment Graph Kernels*

The work of Kashima et al.[3] is based on the use of shared label sequences in the computation of graph kernels. Their marginalized graph kernel uses a Markov model to randomly

generate walks of a labeled graph. The random walks are created using a transition probability matrix combined with a walk termination probability. These collections of random walks are then compared and the number of shared sequences is used to determine the overall similarity between two molecules.

The optimal assignment kernel, described by Frölich et al[4], differs significantly from the marginalized graph kernel. This kernel function first computes the similarity between all vertices in one graph and all vertices in another. The similarity between the two graphs is then computed by finding the maximal weighted bipartite graph between the two sets of vertices, called the optimal assignment. The authors investigate an extension of this method whereby certain structure patterns defined *a priori* by expert knowledge, are collapsed into single vertices, and this reduced graph is used as input to the optimal assignment kernel.

2.2. *Frequent Subgraph Mining*

Frequent subgraph mining is a technique used to enumerate graph substructures that occur in a graph database with at least some specified frequency. This minimum frequency threshold is termed the *support threshold* by the data mining community. After limiting returned subgraphs by frequency, we can further constrain the types we find by setting upper and lower limits on the number of vertices they can contain. In this paper, we use the FFSM algorithm[11], for fast computation of frequent subgraphs. Figure 2.2, adopted from[11], shows an example of this frequent subgraph enumeration. Some work has been done by Deshpande et al.[5] toward the use of these frequent substructures in the classification of chemical compounds with promising results.

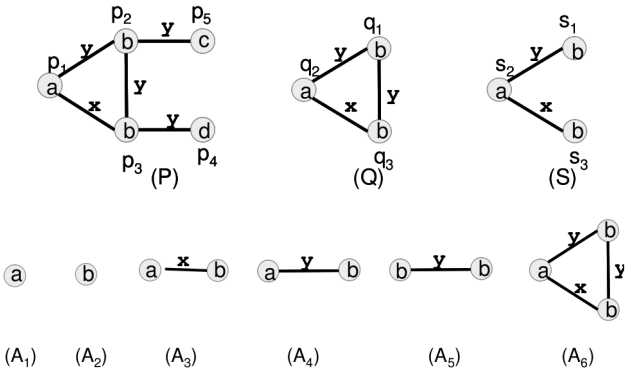

Figure 1. Set of graphs in the top row, and some frequent subgraphs with support threshold 2/3 in the bottom row.

3. Background

Before we proceed to discuss specific methods and other details, let us first provide some general background information regarding both chemical structures and graph mining.

3.1. *Chemical Structure*

Chemical compounds are well-defined structures that are easily encapsulated by a graph representation. Compounds are composed of a number of atoms which are represented as vertices in a graph, and a number of bonds between atoms represented as edges in the graph. Vertices are labeled with the atom element type, and edges are labeled with the bond type. The edges in the graph are undirected, since there is no directionality associated with chemical bonds. Figure 2 shows an example chemical structure.

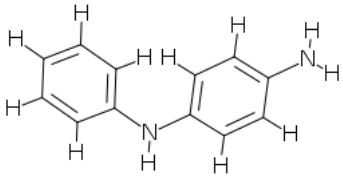

Figure 2. An example chemical structure from the PTC data set. Unlabeled vertices are assumed to be carbon C.

4. Algorithm Design

The following sections outline the algorithm that drives our experimental method. In short, we measure the similarity of graph structures whose vertices and edges have been labeled with various descriptors. These descriptors represent physical and chemical information such as atom and bond types. They are also used to represent the membership of atoms in specific structure patterns that have been mined from the data. To compute the similarity of two graphs, the vertices of one graph are aligned with the vertices of the second graph, such that the total overall similarity is maximized with respect to all possible alignments. Vertex similarity is measured by comparing vertex descriptors, and is computed recursively so that when comparing two vertices, we also compare the neighbors of those vertices, and their neighbors, etc.

4.1. *Structure Pattern Mining*

The frequent subgraph mining problem can be phrased as such: given a set of labeled graphs, the support of an arbitrary subgraph is the fraction of all graphs in the set that contain that subgraph. A subgraph is frequent if its support meets a certain minimum threshold. The goal is to enumerate all the frequent, connected subgraphs in a graph database. The extraction of important subgraph patterns can be controlled by selecting the proper frequency threshold, as well as other parameters such as size and density of subgraph patterns.

4.2. *Optimal Assignment Kernel*

The optimal assignment kernel function computes the similarity between two graph structures. This similarity computation is accomplished by first representing the two sets graph

vertices as a bipartite graph, and then finding the set of weighted edges assigning every vertex in one graph to a vertex in the other. The edge weights are calculated via a recursive vertex similarity function. We present the equations describing this algorithm in detail, as discussed by Frölich et al[4]. The top-level equation describing the similarity of two molecular graphs is:

$$k_A(M_1, M_2) := max_\pi \sum_{h=1}^{m} k_{nei}(v_{\pi(h)}, v_h) \tag{1}$$

Where π denotes a permutation of a subset of graph vertices, and m is the number of vertices in the smaller graph. This is needed since we want to assign all vertices of the smaller graph to vertices in the large graph. The k_{nei} function, which calculates the similarity between two vertices using their local neighbors, is given as follows:

$$k_{nei}(v_1, v_2) := k_v(v_1, v_2) + R_0(v_1, v_2) + S_{nei}(v_1, v_2) \tag{2}$$

$$S_{nei}(v_1, v_2) := \sum_{l=1}^{L} \gamma(l) R_l(v_1, v_2) \tag{3}$$

The functions k_v and k_e compute the similarity between vertices (atoms) and edges (bonds), respectively. These functions could take a variety of forms, but in the OA kernel they are RBF functions between vectors of vertex/edge labels.

The $\gamma(l)$ term is a decay parameter that weights the similarity of neighbors according to their distance from the original vertex. The l parameter controls the topological distance within which to consider neighbors of vertices. The R_l equation, which recursively computes the similarity between two specific vertices is given by the following equation:

$$R_l(v_1, v_2) = \frac{1}{|v_1||v_2|} \sum_{i,j} R_{l-1}(n_i(v_1), n_j(v_2)) \tag{4}$$

Where $|v|$ is the number of neighbors of vertex v, and $n_k(v)$ is the set of neighbors of v. The base case for this equation is R_0, defined by:

$$R_0(v_1, v_2) := \frac{1}{|v_1|} max_\pi \sum_{i=1}^{|v_2|} (k_v(a, b) | k_e(x, y)) \tag{5}$$

$$a = n_{\pi(i)}(v_1),\ b = n_i(v_2) \tag{6}$$

$$x = v_1 \rightarrow n_{\pi(i)}(v_1),\ y = v_2 \rightarrow n_i(v_2) \tag{7}$$

The notation $v \rightarrow n_i(v)$ refers to the edge connecting vertex v with the ith neighboring vertex. The functions k_v and k_e are used to compare vertex and edge descriptors, by counting the total number of descriptor matches.

4.3. *Reduced Graph Representation*

One way in which to utilize the structure patterns that are mined from the graph data is to collapse the specific subgraphs into single vertices in the original graph. This technique

is explored by Frölich et al.[4] with moderate results, although they use predefined structure patterns, so called pharmacophores, identified *a priori* with the help of expert knowledge. Our method ushers these predefined patterns in favor of the structure patterns generated via frequent subgraph mining.

The use of a reduced graph representation does have some advantages. First, by collapsing substructures, we can compare an entire set of vertices at once, reducing the graph complexity and marginally decreasing computation time. Second, by changing the substructure size we can adjust the resolution at which graph structures are compared. The disadvantage of a reduced graph representation is that substructures can only be compared directly to other substructures, and cannot align partial structure matches. As utilized in Frölich et al.[4], this is not as much of a burden since they have defined the best patterns *a priori* using expert knowledge. In our case, however, this is a significant downside, as we have no *a priori* knowledge to guide our pattern generation and we wish to retain as much structural information as possible.

4.4. *Pattern-based Descriptors*

The loss of partial substructure alignment following the use of a reduced graph representation motivated us to find another way of integrating this pattern-based information. Instead of collapsing graph substructures, we simple annotate vertices with additional descriptor labels indicating the vertex's membership in the structure patterns that were previously mined. These pattern-based descriptors are calculated for each vertex and are used by the optimal assignment kernel in the same way that other vertex descriptors are handled. In this way we are able to capture substructure information in the graph vertices without needing to alter the original graph structure.

5. Experimental Study

We conducted classification experiments on five different biological activity data sets, and measured support vector machine (SVM) classifier prediction accuracy for several different feature generation methods. The data sets and classification methods are described in more detail in the following subsections, along with the associated results. Figure 3 gives a graphical overview of the process.

We performed all of our experiments on a desktop computer with a 3Ghz Pertium 4 processor and 1 GB of RAM. Generating a set of frequent subgraphs is very quick, generally a few seconds. Optimal assignment requires significantly more computation time, but not intractable, at less than half an hour for the largest data set.

5.1. *Data Sets*

We have selected five data sets used in various problem areas to evaluate our classifier performance. The Predictive Toxicology Challenge data set, discussed by Helma et al[6], contains a set of chemical compounds classified according to their toxicity in male rats (PTC-MR), female rats (PTC-FR), male mice (PTC-MM), and female mice (PTC-FM).

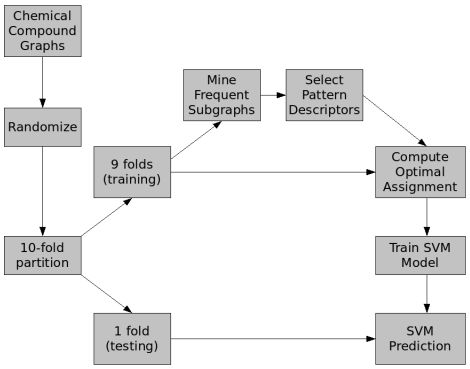

Figure 3. Experimental workflow for a single cross-validation trial.

The Human Intestinal Absorption (HIA) data set (Wessel et al.[7]) contains chemical compounds classified by intestinal absorption activity. We also included two different virtual screening data sets (VS-1,VS-2) used to predict various binding inhibitors from Fontaine et al.[8] and Jorissen et al[9]. The final data set (MD) is from Patterson et al[10], and was used to validate certain molecule descriptors. Various statistics for these data sets can be found in Table 1.

Table 1. Data set statistics.

Dataset	Number of Compounds	Number of Positives	Number of Negatives	Average Compound Size
HIA	86	47	39	22.45
MD	310	148	162	10.38
VS-1	435	279	156	59.81
VS-2	1071	125	946	39.93
PTC-MR	344	152	192	25.56
PTC-MM	336	129	207	25.05
PTC-FR	351	121	230	26.08
PTC-FM	349	143	206	25.25

5.2. *Methods*

We evaluated the performance of the SVM classifier when trained using several different feature sets. The first set of features (FSM) consists only of frequent subgraphs. Those subgraphs are mined using the FFSM software [11] with minimum subgraph frequency of 50%. Each chemical compound is represented by a binary vector with length equal to the number of mined subgraphs. Each subgraph is mapped to a specific vector index, and if a chemical compound contains a subgraph then the bit at the corresponding index is set to one, otherwise it is set to zero.

The second feature set (OA) consists of the similarity values computed by the optimal assignment kernel, as proposed by Frölich et al[4]. Each compound is represented as a real-valued vector containing the computed similarity between it and all other molecules in the data set.

The third feature set (OARG) is computed using the optimal assignment kernel as well, except that we embed the frequent subgraph patterns as a reduced graph representation before computing the optimal assignment. The reduced graph representation is described by Frölich et al.[4] as well, but they use *a priori* patterns instead of frequently mined ones.

Finally, the fourth feature set (OAPD) also consists of the subgraph patterns combined with the optimal assignment kernel, however in this case we do not derive a reduced graph, and instead annotate vertices in a graph with additional descriptors indicating its membership in specific subgraph patterns.

In our experiments, we used the support vector machine (SVM) classifier in order to generate activity predictions. The use of SVM has recently become quite popular for a variety of biological machine learning applications because of its efficiency and ability to operate on high-dimensional data sets. We used the SMO SVM classifier implemented by Platt[13] and included in the Weka data-mining software package by Witten et al[14]. The SVM parameters were fixed, and we used a linear kernel with C = 1. Classifier performance was averaged over a ten-fold cross-validation set.

We perform some feature selection in order to identify the most discriminating frequent patterns. Using a simple statistical formula, known as the Pearson correlation coefficient (PCC), we measure the correlation between a set of feature samples (in our case, the occurrences of a particular subgraph in each of the data samples) and the corresponding class labels. Frequent patterns are ranking according to correlation strength, and the top patterns are selected.

5.3. *Results*

Table 2 contains results reporting the average and standard deviation of the prediction accuracy over the 10 cross-validation trials. With the table, we have the following observations.

Table 2. Average and standard deviation of 10-fold cross-validation accuracy for each data set.

Dataset	Method			
	FSM	OA	OARG	OAPD
HIA	57.36 ±19.11	63.33 ±20.82	62.92 ±22.56	65.28 ±15.44
MD	68.39 ±7.26	70.00 ±6.28	69.35 ±6.5	70.32 ±5.65
VS-1	60.00 ±5.23	64.14 ±3.07	62.07 ±4.06	63.91 ±4.37
VS-2	90.29 ±2.3	94.96 ±1.88	93.18 ±2.68	94.77 ±2.17
PTC-FM	54.16 ±5.82	61.35 ±9.53	59.03 ±6.46	59.29 ±8.86
PTC-FR	63.28 ±5.32	60.10 ±9.21	64.68 ±3.96	64.39 ±3.6
PTC-MM	60.45 ±3.87	62.16 ±6.43	62.75 ±7.69	63.05 ±5.24
PTC-MR	58.42 ±4.43	56.41 ±6	54.07 ±7.52	60.76 ±7.32

First, we notice that OAPD (and OARG) outperforms FSM methods in all of the tried

data sets except one (FSM is better than OARG on the PTC-MR data set). This results indicate that if we use frequent subgraph alone without using the optimal alignment kernel, we do not have a good classifier. Although the conclusion is generally true, interestingly, we found that for the PTC-MR data set, the FSM method outperforms both the OA and OARG methods, while the OAPD method outperforms FSM. This seems to suggest that important information is encoded in the frequent subgraphs, and is being lost in the OARG, but is still preserved in the OAPD method.

Second, we notice that OAPD (or OARG) method outperforms the original OA method in 5 of the tried 8 data sets: HIA, MD, PTC-FR, PTC-MM, PTC-MR. OAPD has a very close performance to that of OA in the rest of the three data sets. The results indicate that our OAPD method provides good performance for diverse data sets which involve tasks such as predicting chemical's toxicology, predicting human intestinal absorption of chemicals, and virtual screening of drugs.

Table 3. Top five highest ranked frequent subgraph patterns for each data set, expressed as SMARTS strings that encode a specific subgraph.

HIA	MD	VS-1	VS-2
[NH3+]C(C)C	C(=CC)(C)S	C(=CC=C)C=C	C(=CCC)C
C(=C)(C)C	C(=CC=CC)(C)S	C(=CC)CNC	C=CCC
C(=CC)(C)C	C(=C)(C=CC=C)S	C(=C)CNC	[NH2+](CC=C)CC
C(=CC)(C=C)C	C(=CCC)C=C	CC(=CC)N	[NH2+](CCC)CC
C(=CC=C)(C=C)C	C(=CS)C=C	CNCC=CC	[NH3+]CC(=CC)C

PTC-MR	PTC-MM	PTC-FR	PTC-FM
[NH2+]C(=C)C=C	[NH3+]CC	[NH2+]C(=CC)C=C	OCC=C
[NH2+]C=CC	c1ccccc1	[NH2+]C(=C)C=C	C(=CC)C(=C)C
[NH3+]CC	C(=CC)C(=C)C	[NH3+]CC	CCC=CC
CC=C	C(=CC=C)C	CC=C	C(=C)(C)C
C(CC)C	C(=C)C(=C)C	C(CC)C	c1ccccc1

In addition to outperforming the previous methods, our new method also reports the specific subgraph patterns that were mined from the training data and used to augment the optimal assignment kernel function. By identifying highly discriminating patterns, our method can offer additional insight into the structural features that contribute to a compound's chemical function. Table 3 contains the five highest ranked (using Pearson correlation coefficient) subgraph patterns for each data set, expressed as SMARTS strings that encode the specific pattern. Many of the patterns in all sets denote various carbon chains (C(CC)C, C=CC, etc.), however there seem to be some unique patterns as well. The MD data set contains carbon chain patterns with some sulfur atoms mixed in, while the VS-1 data set has carbon chains with nitrogen mixed in. The [NH2+] and [NH3+] patterns appear to be important in the VS-2 data set, as well as some of the PTC data sets.

6. Conclusions

Graph structures are a powerful and expressive representation for chemical compounds. In this paper we present a new method, termed OAPD, for computing the similarity of

chemical compounds, based on the use of an optimal assignment graph kernel function augmented with pattern-based descriptors that have been mined from a set of molecular graphs. Our experimental study demonstrate that our OAPD method integrates the structural alignment capabilities of the existing optimal alignment kernel method with the substructure discovery capabilities of the frequent subgraph mining method and delivers better performance in most of the tried benchmarks. In the future, we plan to involve domain experts to evaluate the performance of our algorithm, including the prediction accuracy and the capability of identifying structure important features, in diverse chemical structure data sets.

Acknowledgments

This work has been supported by the Kansas IDeA Network for Biomedical Research Excellence (NIH/NCRR award #P20 RR016475) and the KU Center of Excellence for Chemical Methodology and Library Development (NIH/NIGM award #P50 GM069663)

References

1. M. E. J. Newman. The structure and function of complex networks. *SCIAM Rev.*, 45(2):167-256, 2003.
2. L. Ravaliola, S. J. Swamidass, H. Saigo. Graph Kernels for Chemical Informatics. *Neural Networks*, 18(8):1093-1110, September 2005.
3. H. Kashima, K. Tsuda, A. Inokuchi. Marginalized Kernels Between Labeled Graphs. *Proc. of the Twentieth Int. Conf. on Machine Learning (ICML-03)*, 2003.
4. H. Fröohlich, J. Wegner, F. Sieker, A. Zell. Kernel Functions for Attriubted Molecular Graphs - A new Similarity-Based Approach to ADME Prediction in Classification. *QSAR & Combinatorial Science*, 25(4):317-326, 2006.
5. M. Deshpande, M. Kuramochi, G. Karypis. Frequent Sub-Structure-Based Approaches for Classifying Chemical Compounds. *IEEE Transactions on Knowledge and Data Engineering*, 17(8):1036-1050, August 2005.
6. C. Helma, R. King, S. Kramer. The predictive toxicology challenge 2000-2001. *Bioinformatics*, 17(1):107-108, 2001.
7. M. Wessel, P. Jurs, J. Tolan, S. Muskal. Prediction of Human Intestinal Absorption of Drug Compounds from Molecular Structure. *J. Chem. Inf. Comput. Sci.*, 38(4):726-735, 1998.
8. F. Fontaine, M. Pastor, I. Zamora, and F. Sanz. Anchor-GRIND: Filling the Gap between Standard 3D QSAR and the GRid-INdependent Descriptors. *J. Med. Chem.*, 48(7):2687-2694, 2005.
9. R. Jorissen and M. Gilson. Virtual Screening of Molecular Databases Using a Support Vector Machine. *J. Chem. Inf. Model.*, 45(3):549-561, 2005.
10. D. Patterson, R. Cramer, A. Ferguson, R. Clark, L. Weinberger. Neighbourhood Behaviour: A Useful Concept for Validation of "Molecular Diversity" Descriptors. *J. Med. Chem.*, 39:3049-3059, 1996.
11. J. Huan, W. Wang, J. Prins. Efficient Mining of Frequent Subgraphs in the Presence of Isomorphism. *Proc. of the 3rd IEEE Int. Conf. on Data Mining (ICDM-03)*, 549-552, 2003.
12. V. Vapnik. *Statistical Learning Theory*. John Wiley, New York, NY, 1998.
13. J. Platt. Fast Training of Support Vector Machines using Sequential Minimal Optimization. *Advances in Kernel Methods - Support Vector Learning*, MIT Press, Cambridge, MA, 1998.
14. I. Witten, E. Frank. *Data Mining: Practical machine learning tools and techniques*, 2nd Edition. Morgan Kaufmann, San Francisco, CA, 2005.