# Frequent Substructure-Based Approaches for Classifying Chemical Compounds

Mukund Deshpande, Michihiro Kuramochi, Nikil Wale, and George Karypis, *Senior Member*, *IEEE Computer Society*

**Abstract**—Computational techniques that build models to correctly assign chemical compounds to various classes of interest have many applications in pharmaceutical research and are used extensively at various phases during the drug development process. These techniques are used to solve a number of classification problems such as predicting whether or not a chemical compound has the desired biological activity, is toxic or nontoxic, and filtering out drug-like compounds from large compound libraries. This paper presents a substructure-based classification algorithm that decouples the substructure discovery process from the classification model construction and uses frequent subgraph discovery algorithms to find all topological and geometric substructures present in the data set. The advantage of this approach is that during classification model construction, all relevant substructures are available allowing the classifier to intelligently select the most discriminating ones. The computational scalability is ensured by the use of highly efficient frequent subgraph discovery algorithms coupled with aggressive feature selection. Experimental evaluation on eight different classification problems shows that our approach is computationally scalable and, on average, outperforms existing schemes by 7 percent to 35 percent.

**Index Terms**—Classification, chemical compounds, virtual screening, graphs, SVM.

## 1 Introduction

Discovering new drugs is an expensive and challenging process. Any new drug should not only produce the desired response to the disease, but should do so with minimal side effects and be superior to existing drugs. One of the key steps in the drug design process is the identification of the chemical compounds (*hit* compounds) that display the desired and reproducible behavior against the specific biomolecular target [53] and represents a significant hurdle in the early stages of drug discovery. The 1990s saw the widespread adoption of high-throughput screening (HTS) and ultra HTS [13], [35], which use highly automated techniques to conduct the biological assays and can be used to screen a large number of compounds. Although the number of compounds that can be evaluated by these methods is very large, these numbers are small in comparison to the millions of drug-like compounds that exist or can be synthesized by combinatorial chemistry methods. Moreover, in most cases, it is hard to find all desirable properties in a single compound and medicinal chemists are interested in not just identifying the hits, but studying what part of the chemical compound leads to the desirable behavior, so that new compounds can be rationally synthesized (*lead* development).

Computational techniques that build models to correctly assign chemical compounds to various classes of interest can address these limitations, have many applications in pharmaceutical research, and are used extensively to replace or supplement HTS-based approaches. These techniques are designed to computationally search large compound databases to select a limited number of candidate molecules for testing in order to identify novel chemical entities that have the desired biological activity. The combination of HTS with these *virtual screening* methods allows a move away from purely random-based testing, toward more meaningful and directed iterative rapid-feedback searches of subsets and focused libraries. However, the challenge in developing practical virtual screening methods is to develop chemical compound classification algorithms that can be applied fast enough to rapidly evaluate potentially millions of compounds while achieving sufficient accuracy to successfully identify a subset of compounds that is significantly enriched in hits.

In recent years, two classes of techniques have been developed for solving the chemical compound classification problem. The first class, corresponding to the traditional quantitative structure-activity relationships (QSAR) approaches [8], [36], [37], [80], contains methods that represent the chemical compounds using various descriptors (e.g., physicochemical properties, topological and/or geometric indices, fingerprints, etc.) and then apply various statistical or machine learning approaches to learn the classification models. The second class operates directly on the structure of the chemical compound and try to automatically identify a small number of chemical substructures that can be used to discriminate between the different classes [19], [24], [40], [50], [82]. A number of comparative studies [44], [77] have shown that techniques based on the automatic discovery of chemical substructures are superior to those based on QSAR and require limited user intervention and domain knowledge. However, despite their success, a key limitation of these techniques is that they rely on heuristic search methods to discover these substructures. Even though such approaches reduce the inherently high computational complexity associated with these schemes, they may lead to suboptimal classifiers in cases in which the heuristic

• *The authors are with the Department of Computer Science & Engineering, University of Minnesota, 4-192 EE/CS Building, 200 Union St. SE, Minneapolis, MN 55455.*
*E-mail: {deshpand, kuram, nwale, karypis}@cs.umn.edu.*

*Manuscript received 23 July 2004; revised 27 Oct. 2004; accepted 10 Feb 2005; published online 17 June 2005.*
*For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDESI-0271-0704.*

search failed to uncover substructures that are critical for the classification task.

In this paper, we present a substructure-based classifier that overcomes the limitations associated with existing algorithms. One of the key ideas of this approach is to decouple the substructure discovery process from the classification model construction step and use frequent subgraph discovery algorithms to find all chemical substructures that occur a sufficiently large number of times. Once the complete set of these substructures has been identified, the algorithm then proceeds to build a classification model based on them. The advantage of such an approach is that during classification model construction, all relevant substructures are available allowing the classifier to intelligently select the most discriminating ones. To ensure that such an approach is computationally scalable, we use recently developed [47], [50] highly efficient frequent subgraph discovery algorithms coupled with aggressive feature selection to reduce both the amount of time required to build as well as to apply the classification model. In addition, we present a substructure discovery algorithm that finds a set of substructures whose geometry is conserved, further improving the classification performance of the algorithm.

We experimentally evaluated the performance of these algorithms on eight different problems derived from three publicly available data sets and compared their performance against that of traditional QSAR and fingerprint-based classifiers and existing substructure classifiers based on SUBDUE [20] and SubdueCL [32]. Our results show that these algorithms, on the average, outperform QSAR and fingerprint-based schemes by 7 percent to 35 percent and SUBDUE-based schemes by 10 percent. Portions of these results were first published in a short paper that appears in the International Conference on Data Mining 2003 [26].

The rest of the paper is organized as follows: Section 2 provides some background information related to chemical compounds, their activity, and their representation. Section 3 provides a survey on the related research in this area. Section 4 provides the details of the chemical compound classification approach. Section 5 experimentally evaluates its performance and compares it against other approaches. Finally, Section 6 provides outlines directions of future research and provides some concluding remarks.

## 2 Background

The activity of a compound largely depends on its chemical structure and the arrangement of different atoms in 3D space. As a result, effective classification algorithms must be able to directly take into account the structural nature of these data sets. In this paper, we represent each compound by its corresponding chemical graph [43]. The vertices of these graphs correspond to the various atoms (e.g., carbon, nitrogen, oxygen, etc.), and the edges correspond to the bonds between the atoms (e.g., single, double, etc.). Each of the vertices and edges has a label associated with it. The labels on the vertices correspond to the type of atoms and the labels on the edges correspond to the type of bonds. We will refer to this representation as the ***topological graph*** representation of a chemical compound.

To capture the 3D structural information of a chemical compound, each vertex of the graph has a 3D-coordinate indicating the position of the corresponding atom in 3D space. However, there are two key issues that need to be considered when working with the compound's 3D structure. First, the number of experimentally determined molecular geometries is limited (about 270,000 $X$-ray structures in the Cambridge Crystallographic Database compared to 15 millions known compounds). As a result, the 3D geometry of a compound needs to be computationally determined, which may introduce a certain amount of error. To address this problem, we use the Corina [31] software package to compute the 3D coordinates for all the chemical compounds in our data sets. Corina is a rule and data-based system that has been experimentally shown to predict the 3D structure of compounds with high accuracy. Second, each compound can have multiple low-energy conformations (i.e., multiple 3D structures) that need to be taken into account in order to achieve the highest possible classification performance. In this study, we do not take into account these multiple conformations but, instead, use the single low-energy conformation that is returned by Corina's default settings. However, as discussed in Section 4.1.2, the presented approach for extracting geometric substructures can be easily extended to cases in which multiple conformations are considered as well. Nevertheless, despite this simplification, as our experiments in Section 5 will show, incorporating 3D structure information leads to measurable improvements in the overall classification performance. We will refer to this representation as the ***geometric graph*** representation of a chemical compound.

Finally, for both topological and geometric graphs, we apply two commonly used structure normalization transformations [53]. First, we label all bonds in aromatic rings as *aromatic* (i.e., a different edge-label), and second, we remove the hydrogen atoms that are connected to carbon atoms (i.e., hydrogen-suppressed chemical graphs).

## 3 Related Research

Many approaches have been developed for building classification models for chemical compounds. These approaches can be grouped into two broad categories. The first contains methods that represent the chemical compounds using various descriptors and then apply various statistical or machine learning approaches to learn the classification models. The second category contains methods that automatically analyze the structure of the chemical compounds involved in the problem to identify a set of substructure-based rules, which are then used for classification. A survey of some of the key methods in both categories and a discussion on their relative advantages and disadvantages is provided in the remaining of this section.

### 3.1 Approaches Based on Descriptors

A number of different types of descriptors have been developed that are based on frequency, physicochemical property, topological, and geometric descriptors [8], [80]. In general, the quality of the representation derived from these descriptors tends to improve as we move from frequency, to property, to topology, to geometry-based descriptors. Specifically, a number of studies have shown that topological and geometric descriptors are often superior to those based on simple physicochemical properties, and geometric descriptors tend to outperform their topological counterparts [7], [12], [75]. However, the relative advantage of one class of descriptors over another is not universal. For example, the study in [15] showed that in the context of

ligand-receptor binding, topological descriptors outperform their geometric counterparts.

The types of properties that are captured/measured by these descriptors are identified a priori in a data set independent fashion and rely on extensive domain knowledge. Frequency descriptors are counts that measure basic characteristics of the compounds and include the number of individual atoms, bonds, degrees of connectivity, rings, etc. Physicochemical descriptors correspond to various molecular properties that can be computed directly from the compounds structure. This includes properties such as molecular weight, number of aromatic bonds, molecular connectivity index, $\log P$, total energy, dipole moment, solvent accessible surface area, molar refractivity, ionization potential, atomic electron densities, van der Waals volume, etc. [7], [14], [58]. Topological descriptors are used to measure various aspects of the compounds two-dimensional structure, i.e., the connectivity pattern of the compound's atoms, and include a wide-range of descriptors that are based on topological indices and 2D fragments. Topological indices are similar to physicochemical properties in the sense that they characterize some aspect of molecular data by a single value. These indices encode information about the shape, size, bonding, and branching pattern [9], [34]. 2D fragment descriptors correspond to certain chemical substructures that are present in the chemical compound. This includes various atom-centered, bond-centered, ring-centered fragments [3], fragments based on atom-pairs [17], topological torsions [66], and fragments that are derived by performing a rule-based compound segmentation [10], [11], [54]. Geometric descriptors measure various aspects of the compounds 3D structure that has been either experimentally or computationally determined. These descriptors are usually based on pharmacophores [14]. Pharmacophores are based on the types of interaction observed to be important in ligand-protein binding interactions. Pharmacophore descriptors consist of three or four points separated by well-defined distance ranges and are derived by considering all combinations of three or four atoms over all conformations of a given molecule [6], [22], [33], [68], [75]. Note that information about the 2D fragments and the pharmacophores present in a compound are usually stored in the form of a fingerprint, which is fixed-length string of bits each representing the presence or absence of a particular descriptor.

The actual classification model is learned by transforming each chemical compound into a vector of numerical or binary values whose dimensions correspond to the various descriptors that are used. Within this representation, any classification technique capable of handling numerical or binary features can be used for the classification task. Early research on building these classification models focused primarily on regression-based techniques [14]. This work was pioneered by Hansch et al. [36], [37], which demonstrated that the biological activity of a chemical compound is a function of its physicochemical properties. This led to the development of the quantitative structure-activity relationship (QSAR) methods in which the statistical techniques (i.e., classification model) enable this relationship to be expressed mathematically. However, besides regression-based approaches, other classification techniques have been used that are, in general, more powerful and lead to improved accuracies. This includes techniques based on principle component regression and partial least squares [81], neural networks [5], [27], [57], [85], recursive partitioning [4], [18], [72], phylogenetic-like trees [64], [78], binary QSAR [30], [51], linear discriminant analysis [67], and support vector machines [16].

Descriptor-based approaches are very popular in the pharmaceutical industry and are used extensively to solve various chemical compound classification problems. However, their key limitation stems from the fact that, to a large extent, the classification performance depends on the successful identification of the relevant descriptors that capture the structure-activity relationships for the particular classification problem.

### 3.2 Approaches Based on Substructure Rules

The pioneering work in this field was done by King et al. in the early 1990s [44], [45]. They applied an inductive logic programming (ILP) system [62], Golem [63], to study the behavior of 44 trimethoprin analogues and their observed inhibition of Escherichia coli dihydrofolate reductase and reported a considerable improvement in classification accuracy over the traditional QSAR-based models. In this approach, the chemical compound is expressed using first order logic. Each atom is represented as a predicate consisting of atomID and the element, and a bond is represented as a predicate consisting of two atomIDs. Using this representation, an ILP system discovers rules (i.e., conjunction of predicates) that are good for discriminating the different classes. Since these rules consist of predicates describing atoms and bonds, they essentially correspond to substructures that are present in the chemical compounds. Srinivasan et al. [77] present a detailed comparison of the features generated by ILP with the traditional QSAR properties used for classifying chemical compounds and show that for some applications features discovered by ILP approaches lead to a significant lift in the performance.

Though ILP-based approaches are quite powerful, the high computational complexity of the underlying rule-induction system limits the size of the data set for which they can be applied. Furthermore, they tend to produce rules consisting of relatively small substructures (usually, three to four atoms [21], [23]), limiting the size of structural constraints that are being discovered and hence affecting the classification performance. Another drawback of these approaches is that in order to reduce their computational complexity they employ various heuristics to prune the explored search-space [61], potentially missing substructures that are important for the classification task. One exception is the WARMR system [21], [23] that is specifically developed for chemical compounds and discovers all possible substructures above a certain frequency threshold. However, WARMR's computational complexity is very high and can only be used to discover substructures that occur with relatively high frequency.

One of the fundamental reasons limiting the scalability of ILP-based approaches is the first-order logic-based representation that they use. This representation is much more powerful than what is needed to model chemical compounds and discover substructures. For this reason, a number of researchers have explored the much simpler graph-based representation of the chemical compound's topology and transformed the problem of finding chemical substructures to that of finding subgraphs in this graph-based representation [19], [40], [82]. The best-known approach is the SUBDUE system [20], [38]. SUBDUE finds patterns which can effectively compress the original input data based on the minimum description length (MDL) principle, by substituting those patterns with a single

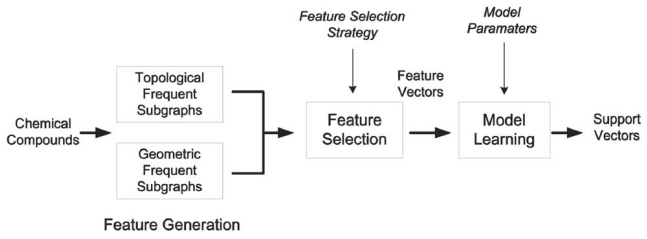


Fig. 1. Frequent subgraph-based classification framework.

vertex. To narrow the search-space and improve its computational efficiency, SUBDUE uses a heuristic beam search approach, which quite often results in failing to find subgraphs that are frequent. The SUBDUE system was also later extended to classify graphs and was referred as SubdueCL [32]. In SubdueCL, instead of using minimum description length as a heuristic, a measure similar to confidence of a subgraph is used as a heuristic. Finally, another heuristic-based scheme is MOLFEA [46] that takes advantage of the compound's SMILES string representation and identifies substructures corresponding to frequently occurring subsequences.

## 4 Classification Based on Frequent Subgraphs

The previous research on classifying chemical compounds (discussed in Section 3) has shown that techniques based on the automatic discovery of chemical substructures are in general superior to traditional descriptor-based approaches and require limited user intervention and domain knowledge. However, despite their success, a key limitation of both the ILP and the subgraph-based techniques, is that they rely on heuristic search methods to discover the substructures to be used for classification. As discussed in Section 3, even though such approaches reduce the inherently high computational complexity associated with these schemes, they may lead to suboptimal classifiers in cases in which the heuristic search fails to uncover substructures that are critical for the classification task.

To overcome this problem, we developed a classification algorithm for chemical compounds that uses the graph-based representation and limits the number of substructures that are pruned a priori. The key idea of our approach is to decouple the substructure discovery process from the classification model construction step, and use frequent subgraph discovery algorithms to find all chemical substructures that occur a sufficiently large number of times. Once the complete set of such substructures has been identified, our algorithm then proceeds to build a classification model based on them. To a large extent, this approach is similar in spirit to the recently developed frequent-itemset-based classification algorithms [25], [55], [56] that have been shown to outperform traditional classifiers that rely on heuristic search methods to discover the classification rules.

The overall outline of our classification methodology is shown in Fig. 1. It consists of three distinct steps: 1) feature generation, 2) feature selection, and 3) classification model construction. During the feature generation step, the chemical compounds are mined to discover the frequently occurring substructures that correspond to either topological or geometric subgraphs. These substructures are then used as the features by which the compounds are represented in the subsequent steps. During the second step, a small set of features is selected such that the selected features can correctly discriminate between the different classes present in the data set. Finally, in the last step, each chemical compound is represented using these set of features and a classification model is learned.

This methodology, by following the above three-step framework, is designed to overcome the limitations of existing approaches. By using computationally efficient subgraph discovery algorithms to find all chemical substructures (topological or geometric) that occur a sufficiently large number of times in the compounds, they can discover substructures that are both specific to the particular classification problem being solved and at the same time involve arbitrarily complex substructures. By discovering the complete set of frequent subgraphs and decoupling the substructure discovery process from the feature generation step, they can proceed to select and synthesize the most discriminating descriptors for the particular classification problem that take into account all relevant information. Finally, by employing advanced machine learning techniques, they can account for the relationships between these features at different levels of granularity and complexity leading to high classification accuracy.

### 4.1 Feature Generation

Our classification algorithm finds substructures in a chemical compound database using two different methods. The first method uses the topological graph representation of each compound, whereas the second method is based on the corresponding geometric graph representation (discussed in Section 2). In both of these methods, our algorithm uses the topological or geometric connected subgraphs that occur in at least $\sigma$ percent of the compounds to define the substructures.

There are two important restrictions on the type of the substructures that are discovered by our approach. The first has to do with the fact that we are only interested in substructures that are connected and is motivated by the fact that connectivity is a natural property of such patterns. The second has to do with the fact that we are only interested in frequent substructures (as determined by the value of $\sigma$) as this ensures that we do not discover spurious substructures that will in general not be statistically significant. Furthermore, this minimum support constraint also helps in making the problem of frequent subgraph discovery computationally tractable.

#### 4.1.1 Frequent Topological Subgraphs

Developing frequent subgraph discovery algorithms is particularly challenging and computationally intensive as graph and/or subgraph isomorphisms play a key role throughout the computations. Despite that, in recent years, a number of algorithms have been developed capable of finding all frequently occurring subgraphs with reasonable computational efficiency. These are the AGM algorithm developed by Inokuchi et al. [40], the FSG algorithm developed by members of our group [47], [50], the chemical substructure discovery algorithm developed by Borgelt and Berthold [19], the gSpan algorithm developed by Yan and Han [82], the FFSM by Huan et al. [39], and, more recently, the algorithm by Nijssen and Kok [65]. The enabling factors to the computational efficiency of these schemes have been

1) the development of efficient candidate subgraph generation schemes that reduce the number of times the same candidate subgraph is being generated, 2) the use of efficient canonical labeling schemes to represent the various subgraphs, and 3) the use of various techniques developed by the data mining community to reduce the number of times subgraph isomorphism computations need to be performed.

In our classification algorithm, we find the frequently occurring subgraphs using the FSG algorithm. FSG takes as input a database $D$ of graphs and a minimum support $\sigma$, and finds all connected subgraphs that occur in at least $\sigma$ percent of the transactions. FSG, initially presented in [47], with subsequent improvements presented in [50], uses a breadth-first approach to discover the lattice of frequent subgraphs. It starts by enumerating small frequent graphs consisting of one and two edges and then proceeds to find larger subgraphs by joining previously discovered smaller frequent subgraphs. The size of these subgraphs is grown by adding one-edge-at-a-time. The lattice of frequent patterns is used to prune the set of candidate patterns and it only explicitly computes the frequency of the patterns which survive this downward closure pruning. Despite the inherent complexity of the problem, FSG employs a number of sophisticated techniques to achieve high computational performance. It uses a canonical labeling algorithm that fully makes use of edge and vertex labels for fast processing, and various vertex invariants to reduce the complexity of determining the canonical label of a graph. These canonical labels are then used to establish the identity and total order of the frequent and candidate subgraphs, a critical step of redundant candidate elimination and downward closure testing. It uses a sophisticated scheme for candidate generation [50] that minimizes the number of times each candidate subgraph gets generated and also dramatically reduces the generation of subgraphs that fail the downward closure test. Finally, for determining the actual frequency of each subgraph, FSG reduces the number of subgraph isomorphism operations by using TID-lists [29], [74], [83], [84] to keep track of the set of transactions that supported the frequent patterns discovered at the previous level of the lattice. For every candidate, FSG takes the intersection of TID-lists of its parents, and performs the subgraph isomorphism only on the transactions contained in the resulting TID-list. The experiments presented in [50] show that FSG is able to scale to large data sets and low support values. For example, it can mine a data set containing 200,000 chemical compounds at 1 percent minimum support level in about one hour.

#### 4.1.2 Frequent Geometric Subgraphs

Topological substructures capture the connectivity of atoms in the chemical compound, but they ignore the 3D shape (3D arrangement of atoms) of the substructures. For certain classification problems, the 3D shape of the substructure might be essential for determining the chemical activity of a compound. For instance, the geometric configuration of atoms in a substructure is crucial for its ability to bind to a particular target [53]. For this reason, we developed an algorithm that find all frequent substructures whose topology as well as geometry is conserved.

There are two important aspects specific to the geometric subgraphs that need to be considered. First, since the coordinates of the vertices depend on a particular reference coordinate axes, we would like the discovered geometric subgraphs to be independent of these coordinate axes, i.e., we are interested in geometric subgraphs whose occurrences are translation and rotation invariant. This dramatically increases the overall complexity of the geometric subgraph discovery process because we may need to consider all possible geometric configurations of a single pattern. Second, while determining if a geometric subgraph is contained in a bigger geometric graph we would like to allow some tolerance when we establish a match between coordinates, ensuring that slight deviations in coordinates between two identical topological subgraphs do not lead to the creation of two geometric subgraphs. The amount of tolerance ($r$) should be a user-specified parameter. The task of discovering such $r$-tolerant frequent geometric subgraphs dramatically changes the nature of the problem. In traditional pattern discovery problems such as finding frequent itemsets, sequential patterns, and/or frequent topological graphs there is a clear definition of what a pattern is, given its set of supporting transactions. On the other hand, in the case of $r$-tolerant geometric subgraphs, there are many different geometric representations of the same pattern (all of which will be $r$-tolerant isomorphic to each other). The problem becomes not only that of finding a pattern and its support, but also finding the right representative for this pattern. The selection of the right representative can have a serious impact on correctly computing the support of the pattern. For example, given a set of subgraphs that are $r$-tolerant isomorphic to each other, the one that corresponds to an *outlier* will tend to have a lower support than the one corresponding to the *center*. These two aspects of geometric subgraphs makes the task of discovering the full fledged geometric subgraphs extremely hard [48], [49].

To overcome this problem we developed a simpler, albeit less discriminatory, representation for geometric subgraphs. We use a property of a geometric graph called the ***average interatomic distance*** that is defined as the average Euclidean distance between all pairs of atoms in the molecule. Note that the average interatomic distance is computed between all pairs of atoms irrespective of whether a bonds connects the atoms or not. The average interatomic distance can be thought of as a geometric signature of a topological subgraph. The geometric subgraph consists of two components, a topological subgraph and an interval of average interatomic distance associated with it. A geometric graph contains this geometric subgraph if it contains the topological subgraph and the average interatomic distance of the embedding (of the topological subgraph) is within the interval associated with the geometric subgraph. Note that this geometric representation is also translation and rotation invariant, and the width of the interval determines the tolerance displayed by the geometric subgraph. We are interested in discovering such geometric subgraphs that occur above $\sigma$ percent of the transactions and the interval of average interatomic distance is bound by $r$.

Since a geometric subgraph contains a topological subgraph, for the geometric subgraph to be frequent the corresponding topological subgraph has to be frequent, as well. This allows us to take advantage of the existing approach to discover topological subgraphs. We modify the frequency counting stage of the FSG algorithm as follows: If a subgraph $g$ is contained in a transaction $t$, then all possible embeddings of $g$ in $t$ are found and the average interatomic distance for each of these embeddings is computed. As a

result, at the end of the frequent subgraph discovery each topological subgraph has a list of average interatomic distances associated with it. Each one of the average interatomic distances corresponds to one of the embeddings, i.e., a geometric configuration of the topological subgraph. This algorithm can be easily extended to cases in which there are multiple 3D conformations associated with each chemical compound (as discussed in Section 2), by simply treating each distinct conformation as a different chemical compound.

The task of discovering geometric subgraphs now reduces to identifying those geometric configurations that are frequent enough, i.e., identify intervals of average interatomic distances such that each interval contains the minimum number geometric configurations (it occurs in $\sigma$ percent of the transactions) and the width of the interval is smaller than the tolerance threshold ($r$). This task can be thought of as 1D clustering on the vector of average interatomic distances such that each cluster contains items above the minimum support and the spread of each cluster is bounded by the tolerance $r$. Note that not all items will belong to a valid cluster as some of them will be infrequent. In our experiments, we set the value of $r$ to be equal to half of the minimum distance between any two pairs of atoms in the compounds.

To find such clusters, we perform agglomerative clustering on the vector of average interatomic distance values. The distance between any two average interatomic distance values is defined as the difference in their numeric values. To ensure that we get the largest possible clusters, we use the maximum-link criterion function for deciding which two clusters should be merged [42]. The process of agglomeration is continued until the interval containing all the items in the cluster is below the tolerance threshold ($r$). When we reach a stage where further agglomeration would increase the spread of the cluster beyond the tolerance threshold, we check the number of items contained in the cluster. If the number of items is above the support threshold, then the interval associated with this cluster is considered as a geometric feature. Since we are clustering one-dimensional data sets, the clustering complexity is low. Some examples of the distribution of the average interatomic distance values and the associated clusters are shown in Fig. 2. Note that the average interatomic distance values of the third example are uniformly spread and lead to no geometric subgraph.

Note that this algorithm for computing geometric subgraphs is approximate in nature for two reasons. First, the average interatomic distance may map two different geometric subgraphs to the same average interatomic distance value. Second, the clustering algorithm may not find the complete set of geometric subgraphs that satisfy the $r$ tolerance. Nevertheless, as our experiments in Section 5 show the geometric subgraphs discovered by this approach improve the classification accuracy of the algorithm.

#### 4.1.3 Additional Considerations

Even though FSG provides the general functionality required to find all frequently occurring substructures in chemical data sets, there are a number of issues that need to be addressed before it can be applied as a black-box tool for feature discovery in the context of classification. One issue deals with the selecting the right value for the $\sigma$, the support constraint used for discovering frequent substructures. The value of $\sigma$ controls the number of subgraphs discovered by FSG. Choosing a good value of $\sigma$ is especially important for the data set containing classes of significantly different sizes. In such cases, in order to ensure that FSG is able to find features that are meaningful for all the classes, the minimum support should be small enough so that the corresponding absolute frequency can capture the size of the smaller class.

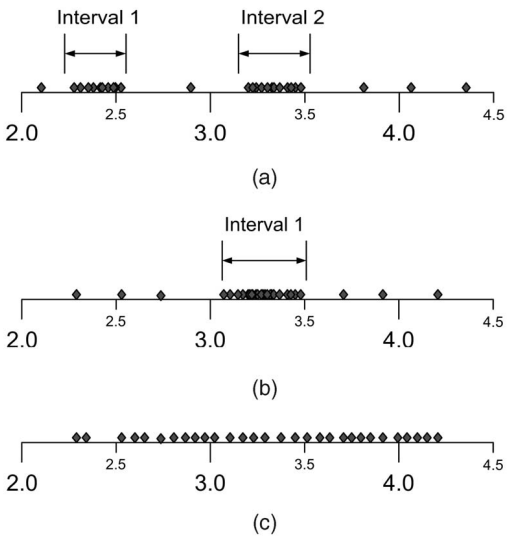


Fig. 2. Some examples of the one-dimensional clustering of average interatomic distance values.

For this reason, we first partition the complete data set, using the class label of the examples, into specific class specific data sets. We then run FSG on each of these *class data sets*. This partitioning of the data set ensures that sufficient subgraphs are discovered for those class labels which occur rarely in the data set. Next, we combine subgraphs discovered from each of the *class data set*. After this step each subgraph has a vector that contains the frequency with which it occurs in each class.

### 4.2 Feature Selection

The frequent subgraph discovery algorithms described in Section 4.1 discovers all the substructures (topological or geometric) that occur above a certain support constraint ($\sigma$) in the data set. Though the discovery algorithm is computationally efficient, the algorithm can generate a large number of features. A large number of features is detrimental for two reasons. First, it could increase the time required to build the model. But, more importantly, a large number of features can increase the time required to classify a chemical compound, as we need to first identify which of the discovered features it contains before we can apply the classification model. Determining whether a compound contains a particular feature or not can be computationally expensive as it may require a subgraph isomorphism operation. This problem is especially critical in the drug discovery process where the classification model is learned on a small set of chemical compounds and it is then applied on large chemical compound libraries containing millions of compounds.

One way of solving this problem is to follow a heuristic subgraph discovery approach (similar in spirit to previously developed methods [20], [32]) in which during the subgraph discovery phase itself, the discriminatory ability of a particular subgraph is determined, and the discovery

process is terminated as soon as a subgraph is generated that is less discriminatory than any of its subgraphs. By following this approach, the total number of features will be substantially reduced, achieving the desired objective. However, the limitation with such an approach is that it may fail to discover and use highly discriminatory subgraphs. This is because the discriminatory ability of a subgraph does not (in general) consistently increase as a function of its size, and subgraphs that appear to be poor discriminators may become very discriminatory by growing their size. For this reason, in order to develop an effective feature selection method, we use a scheme that first finds all frequent subgraphs and then selects among them a small set of discriminatory features. The advantage of this approach is that during feature selection all frequent subgraphs are considered irrespective of when they were generated and whether or not they contain less or more discriminatory subgraphs.

The feature selection scheme is based on the ***sequential covering paradigm*** used to learn rule sets [59]. To apply this algorithm, we assume that each discovered substructure corresponds to a rule, with the class label of the substructure as the *target attribute*, such rules are referred to as *class-rules* in [56]. The sequential covering algorithm takes as input a set of examples and the features discovered from these examples, and iteratively applies the feature selection step. In this step, the algorithm selects the feature that has the highest estimated accuracy. After selecting this feature, all the examples containing this feature are eliminated and the feature is marked as selected. In the next iteration of the algorithm, the same step is applied, but on a smaller set of examples. The algorithm continues in an iterative fashion until either all the features are selected or all the examples are eliminated.

In this paper, we use a computationally efficient implementation of sequential covering algorithm known as CBA [56], this algorithm proceeds by first sorting the features based on confidence and then applying the sequential covering algorithm on this sorted set of features. One of the advantages of this approach is that it requires a minimal number of passes on the data set, hence is very scalable. To obtain a better control over the number of selected features, we use an extension of the sequential covering scheme known as *Classification based on Multiple Rules* (CMAR) [55]. In this scheme, instead of removing the example after it is covered by the selected feature, the example is removed only if that example is covered by $\delta$ selected features. The number of selected rules increases as the value of $\delta$ increases, an increase in the number of features usually translates into an improvement in the accuracy as more features are used to classify a particular example. The value of $\delta$ is specified by the user and provides a means to the user to control the number of features used for classification.

### 4.3 Classification Model Construction

Given the frequent subgraphs discovered in the previous step, our algorithm treats each of these subgraphs as a feature and represents the chemical compound as a frequency vector. The $i$th entry of this vector is equal to the number of times (frequency) that feature occurs in the compound's graph. This mapping into the feature space of frequent subgraphs is performed both for the training and the test data set. Note that the frequent subgraphs were identified by mining *only* the graphs of the chemical compounds in the training set. However, the mapping of the test set requires that we check each frequent subgraph against the graph of the test compound using subgraph isomorphism. Fortunately, the overall process can be substantially accelerated by taking into account the frequent subgraph lattice that is also generated by FSG. In this case, we traverse the lattice from top to bottom and only visit the child nodes of a subgraph if that subgraph is isomorphic to the chemical compound.

Once the feature vectors for each chemical compound have been built, any one of the existing classification algorithms can potentially be used for classification. However, the characteristics of the transformed data set and the nature of the classification problem itself tends to limit the applicability of certain classes of classification algorithms. In particular, the transformed data set will most likely be high-dimensional and, second, it will be sparse, in the sense that each compound will have only a few of these features, and each feature will be present in only a few of the compounds. Moreover, in most cases, the positive class will be much smaller than the negative class, making it unsuitable for classifiers that primarily focus on optimizing the overall classification accuracy.

In our study, we built the classification models using support vector machines (SVM) [79], as they are well-suited for operating in such sparse and high-dimensional data sets. Furthermore, an additional advantage of SVM is that it allows us to directly control the cost associated with the misclassification of examples from the different classes [60]. This allows us to associate a higher cost for the misclassification of positive instances; thus, biasing the classifier to learn a model that tries to increase the true-positive rate, at the expense of increasing the false positive rate.

## 5 Experimental Evaluation

We experimentally evaluated the performance of our classification algorithm and compared it against that achieved by earlier approaches on a variety of chemical compound data sets. The data sets, experimental methodology, and results are described in subsequent sections.

### 5.1 Data Sets

We evaluated the performance of our classification algorithm on eight classification problems derived from three different chemical compound data sets. The first data set that was used as a part of the Predictive Toxicology Evaluation Challenge [76] contains data published by the US National Institute for Environmental Health Sciences and consists of bio-assays of different chemical compounds on rodents to study the carcinogenicity properties of the compounds. Each compound is evaluated on male mice, female mice, male rats, and female rats, and is assigned four class labels each indicating the toxicity of the compound for that animal. There are four classification problems one corresponding to each of the rodents and will be referred as *P1*, *P2*, *P3*, and *P4*. The second data set is obtained from the National Cancer Institute's DTP AIDS Antiviral Screen program [28], [46]. Each compound in the data set is evaluated for evidence of anti-HIV activity. Compounds that provided at least 50 percent protection were listed as *confirmed moderately active* (CM). Compounds that reproducibly provided 100 percent protection were listed as *confirmed active* (CA). Compounds neither active nor moderately active were listed as *confirmed inactive* (CI). We

TABLE 1
The Characteristics of the Various Data Sets

| | *Toxic.* | *Aids* | *Anthrax* | *Class Dist.* (% +ve class) | |
|---|---|---|---|---|---|
| $N$ | 417 | 42,687 | 34,836 | **Toxicology** | |
| $\bar{N}_A$ | 25 | 46 | 25 | P1: Male Mice | 38.3% |
| $\bar{N}_B$ | 26 | 48 | 25 | P2: Female Mice | 40.9% |
| $\bar{L}_A$ | 40 | 82 | 25 | P3: Male Rats | 44.2% |
| $\bar{L}_B$ | 4 | 4 | 4 | P4: Female Rats | 34.4% |
| $\max N_A$ | 106 | 438 | 41 | **AIDS** | |
| $\min N_A$ | 2 | 2 | 12 | H1: CA/CM | 28.1% |
| $\max N_B$ | 1 | 276 | 44 | H2: (CA+CM)/CI | 3.5% |
| $\min N_B$ | 85 | 1 | 12 | H3: CA/CI | 1.0% |
| | | | | **Anthrax** | |
| | | | | A1: active/inactive | 35% |

$N$ is the number of compounds in the database. $\bar{N}_A$ and $\bar{N}_B$ are the average number of atoms and bonds in each compound. $\bar{L}_A$ and $\bar{L}_B$ are the average number of atom- and bond-types in each dataset. $\max N_A/\min N_A$ and $\max N_B/\min N_B$ are the maximum/minimum number of atoms and bonds over all the compounds in each dataset.

formulated three classification problems. The first problem was designed to classify between CA and CM; the second between CM + CA and CI, and the third between CA and CI. We will refer to these problems as *H1*, *H2*, and *H3*, respectively. The third data set was obtained from the Center of Computational Drug Discovery's anthrax project at the University of Oxford [71]. The goal of this project was to discover small molecules that would bind with the heptameric protective antigen component of the anthrax toxin, and prevent it from spreading its toxic effects. The screen identified a set of 12,376 compounds that could potentially bind to the anthrax toxin and a set of 22,460 compounds that were unlikely to bind to the toxin. The classification problem for this data set was given a chemical compound classify it in to one of these two classes, i.e, will the compound bind the anthrax toxin or not. This classification problem is referred as *A1*.

Some important characteristics of these data sets are summarized in Table 1. The right-hand side of the table displays the class distribution for different classification problems, for each problem the table displays the percentage of positive class found in the data set for that classification problem. Note that both the DTP-AIDS and the Anthrax data sets are quite large containing 42,687 and 34,836 compounds, respectively. Moreover, in the case of DTP-AIDS, each compound is also quite large having on an average 46 atoms and 48 bonds.

## 5.2 Experimental Methodology and Metrics

The classifications results were obtained by performing 5-way cross validation on the data set, ensuring that the class distribution in each fold is identical to the original data set. In each one of the cross validation experiments, the test set was never considered and our algorithm used only the training-set to find the frequent substructures, perform feature selection, and build the classification model. For the SVM classifier, we used SVMLight library [41]. All the experiments were conducted on a 1500MHz Athlon MP processors having a 2GB of memory.

Since the size of the positive class is significantly smaller than the negative class, using *accuracy* to judge a classifier would be incorrect. To get a better understanding of the classifier performance for different cost settings, we obtain the ROC curve [69] for each classifier. ROC curve plots the false positive rate ($X$-axis) versus the true positive rate ($Y$-axis) of a classier; it displays the performance of the classifier regardless of class distribution or error cost. Two classifiers are evaluated by comparing the area under their respective ROC curves, a larger area under ROC curve indicating better performance. The area under the ROC curve will be referred by the parameter $A$.

TABLE 2
Varying Minimum Support Threshold ($\sigma$)

| $D$ | $\sigma$=10.0% | | | | $\sigma$ = 15.0% | | | | $\sigma$ = 20.0% | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Topo.* | | *Geom.* | | *Topo.* | | *Geom.* | | *Topo.* | | *Geom.* | |
| | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ |
| P1 | 66.0 | 1211 | 65.5 | 1317 | 66.0 | 513 | 64.1 | 478 | 64.4 | 254 | 60.2 | 268 |
| P2 | 65.0 | 967 | 64.0 | 1165 | 65.1 | 380 | 63.3 | 395 | 64.2 | 217 | 63.1 | 235 |
| P3 | 60.5 | 597 | 60.7 | 808 | 59.4 | 248 | 61.3 | 302 | 59.9 | 168 | 60.9 | 204 |
| P4 | 54.3 | 275 | 55.4 | 394 | 56.2 | 173 | 57.4 | 240 | 57.3 | 84 | 58.3 | 104 |
| H1 | 81.0 | 27034 | 82.1 | 29554 | 77.4 | 13531 | 79.2 | 8247 | 78.4 | 7479 | 79.5 | 7700 |
| H2 | 70.1 | 1797 | 76.0 | 3739 | 63.6 | 307 | 62.2 | 953 | 59.0 | 139 | 58.1 | 493 |
| H3 | 83.9 | 27019 | 89.5 | 30525 | 83.6 | 13557 | 88.8 | 11240 | 84.6 | 7482 | 87.7 | 7494 |
| A1 | 78.2 | 476 | 79.0 | 492 | 78.2 | 484 | 77.6 | 332 | 77.1 | 312 | 76.1 | 193 |

"$A$" denotes the area under the ROC curve and "$N_f$" denotes the number of discovered frequent subgraphs.

## 5.3 Results

### 5.3.1 Varying Minimum Support

The key parameter of the proposed frequent substructure-based classification algorithm is the choice of the minimum support ($\sigma$) used to discover the frequent substructures (either topological or geometric). To evaluate the sensitivity of the algorithm on this parameter. we performed a set of experiments in which we varied $\sigma$ from 10 percent to 20 percent in 5 percent increments. The results of these experiments are shown in the left subtable of Table 2 for both topological and geometric substructures.

From Table 2, we observe that as we increase $\sigma$, the classification performance for most data sets tends to degrade. However, in most cases, this degradation is gradual and correlates well with the decrease on the number of substructures that were discovered by the frequent subgraph discovery algorithms. The only exception is the H2 problem for which the classification performance (as measured by ROC) degrades substantially as we increase the minimum support from 10 percent to 20 percent. Specifically, in the case of topological subgraphs, the performance drops from 70.1 down to 59.0, and in the case of geometric subgraphs it drops from 76.0 to 58.1.

These results suggest that lower values of support are in general better as they lead to better classification performance. However, as the support decreases, the number of discovered substructures and the amount of time required also increases. Moreover, models derived from an extremely large number of features, some of which have very small occurrence frequency run the risk of overfitting the training set (i.e., they produce high accuracies on the training set, but fail to generalize on the test set). Thus, depending on the data set, some experimentation may be required to select the proper values of support that balances these conflicting requirements.

In our study, we performed such experimentation. For each data set, we kept on decreasing the value of support down to the point after which the number of features that were generated was too large to be efficiently processed by the SVM library. The resulting support values, number of features, and associated classification performance are shown in Table 3. Note that for each problem, two different

TABLE 3
Optimized Minimum Support Threshold ($\sigma$)

| $D$ | *Topo.* | | *Geom.* | | Per class | $Time_p$ |
|---|---|---|---|---|---|---|
| | $A$ | $N_f$ | $A$ | $N_f$ | $\sigma$ | (sec) |
| P1 | 65.5 | 24510 | 65.0 | 23612 | 3.0, 3.0 | 211 |
| P2 | 67.3 | 7875 | 69.9 | 12673 | 3.0, 3.0 | 72 |
| P3 | 62.6 | 7504 | 64.8 | 10857 | 3.0, 3.0 | 66 |
| P4 | 63.4 | 25790 | 63.7 | 31402 | 3.0, 3.0 | 231 |
| H1 | 81.0 | 27034 | 82.1 | 29554 | 10.0, 10.0 | 137 |
| H2 | 76.5 | 18542 | 79.1 | 29024 | 10.0, 5.0 | 1016 |
| H3 | 83.9 | 27019 | 89.5 | 30525 | 10.0, 10.0 | 392 |
| A1 | 81.7 | 3054 | 82.6 | 3186 | 5.0, 3.0 | 145 |

"$A$" denotes the area under the ROC curve and "$N_f$" denotes the number of discovered frequent subgraphs.

TABLE 4
The Area under the ROC Curve Obtained by Varying the Misclassification Cost

| *Dataset* | Topo | | Geom | |
|---|---|---|---|---|
| | $\beta = 1.0$ | $\beta = EqCost$ | $\beta = 1.0$ | $\beta = EqCost$ |
| P1 | 65.5 | 65.3 | 65.0 | 66.7 |
| P2 | 67.3 | 66.8 | 69.9 | 69.2 |
| P3 | 62.6 | 62.6 | 64.8 | 64.6 |
| P4 | 63.4 | 65.2 | 63.7 | 66.1 |
| H1 | 81.0 | 79.2 | 82.1 | 81.1 |
| H2 | 76.5 | 79.4 | 79.1 | 81.9 |
| H3 | 83.9 | 90.8 | 89.5 | 94.0 |
| A1 | 81.7 | 82.1 | 82.6 | 83.0 |

"$\beta = 1.0$" indicates the experiments in which each positive and negative example had a weight of one, and "$\beta = EqCost$" indicates the experiments in which the misclassification cost of the positive examples was increased to match the number of negative examples.

support values are displayed corresponding to the supports that were used to mine the positive and negative class, respectively. The last column shows the amount of time required by FSG to find the frequent subgraphs and provides a good indication of the computational complexity at the feature discovery phase of our classification algorithm. Finally, Fig. 3 shows the distribution of the size of the features discovered by FSG for the optimal values of $\sigma$ for the H3 and A1 data sets. From these histograms, we can see that the majority of the subgraphs discovered by FSG are actually quite large.

Comparing the ROC values obtained in these experiments with those obtained for $\sigma = 10$ percent, we can see that as before, the lower support values tend to improve the results, with measurable improvements for problems in which the number of discovered substructures increased substantially. In the rest of our experimental evaluation, we will be using the frequent subgraphs that were generated using these values of support.

### 5.3.2 Varying Misclassification Costs

Since the number of positive examples is in general much smaller than the number of negative examples, we performed a set of experiments in which the misclassification cost associated with each positive example was increased to match the number of negative examples. That is, if $n^+$ and $n^-$ is the number of positive and negative examples, respectively, the misclassification cost $\beta$ was set equal to $(n^-/n^+ - 1)$ (so that $n^- = \beta n^+$). We refer to this value of $\beta$ as the "*EqCost*" value. The classification performance achieved by our algorithm using either topological or geometric subgraphs for $\beta = 1.0$ and $\beta = EqCost$ is shown in Table 4. Note that the $\beta = 1.0$ results are the same with those presented in the right subtable of Table 2.

From the results in this table, we can see that, in general, increasing the misclassification cost so that it balances the size of positive and negative class tends to improve the classification accuracy. When $\beta = EqCost$, the classification performance improves for four and five problems for the topological and geometric subgraphs, respectively. Moreover, in the cases in which the performance decreased, that decrease was quite small, whereas the improvements achieved for some problem instances (e.g., P4, H1, and H2) was significant. In the rest of our experiments, we will focus only on the results obtained by setting $\beta = EqCost$.

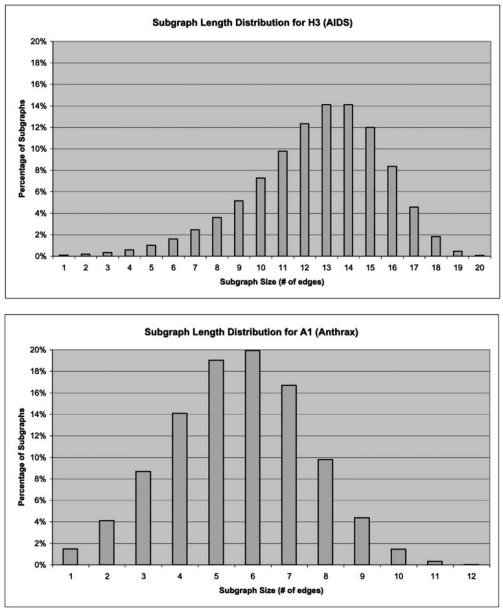


Fig. 3. The size distribution of the various discovered subgraphs for H3 and A1 data sets.

### 5.3.3 Feature Selection

We evaluated the performance of the feature selection scheme based on sequential covering (described in Section 4.2) by performing a set of experiments in which we varied the parameter $\delta$ that controls the number of times an example must be covered by a feature, before it is removed from the set of yet to be covered examples. Table 5 displays the results of these experiments. The results under the column labeled "Original" shows the performance of the classifier without any feature selection. These results are identical to those shown in Table 4 for $\beta = EqCost$ and are included here to make comparisons easier.

Two key observations can be made by studying the results in this table. First, as expected, the feature selection scheme is able to substantially reduce the number of features. In some cases, the number of features that was selected decreased by almost two orders of magnitude. Also, as $\delta$ increases, the number of retained features increases; however, this increase is gradual. Second, the overall classification performance achieved by the feature selection scheme when $\delta \geq 5$ is quite comparable to that achieved with no feature selection. The actual performance depends on the problem instance and whether or not we use topological or geometric subgraphs. In particular, for

TABLE 5
Results Obtained Using Feature Selection Based on Sequential Rule Covering

| | *Topological Features* | | | | | | | | | | *Geometric Features* | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Original* | | $\delta = 1$ | | $\delta = 5$ | | $\delta = 10$ | | $\delta = 15$ | | *Original* | | $\delta = 1$ | | $\delta = 5$ | | $\delta = 10$ | | $\delta = 15$ | |
| *Dataset* | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ | $A$ | $N_f$ |
| P1 | 65.3 | 24510 | 65.4 | 143 | 66.4 | 85 | 66.5 | 598 | 66.7 | 811 | 66.7 | 23612 | 68.3 | 161 | 68.1 | 381 | 67.4 | 613 | 68.7 | 267 |
| P2 | 66.8 | 7875 | 69.5 | 160 | 69.6 | 436 | 68.0 | 718 | 67.5 | 927 | 69.2 | 12673 | 72.2 | 169 | 73.9 | 398 | 73.1 | 646 | 73.0 | 265 |
| P3 | 62.6 | 7504 | 68.0 | 171 | 65.2 | 455 | 64.2 | 730 | 64.5 | 948 | 64.6 | 10857 | 71.1 | 175 | 70.0 | 456 | 71.0 | 241 | 66.7 | 951 |
| P4 | 65.2 | 25790 | 66.3 | 156 | 66.0 | 379 | 64.5 | 580 | 64.1 | 775 | 66.1 | 31402 | 68.8 | 164 | 69.7 | 220 | 67.4 | 609 | 66.2 | 819 |
| H1 | 79.2 | 27034 | 78.4 | 108 | 79.2 | 345 | 79.1 | 571 | 79.5 | 796 | 81.1 | 29554 | 80.8 | 128 | 81.6 | 396 | 81.9 | 650 | 82.1 | 885 |
| H2 | 79.4 | 18542 | 77.1 | 370 | 78.0 | 1197 | 78.5 | 1904 | 78.5 | 2460 | 81.9 | 29024 | 80.0 | 525 | 80.4 | 1523 | 80.6 | 2467 | 81.2 | 3249 |
| H3 | 90.8 | 27019 | 88.4 | 111 | 89.6 | 377 | 90.0 | 638 | 90.5 | 869 | 94.0 | 30525 | 91.3 | 177 | 92.2 | 496 | 93.1 | 831 | 93.2 | 1119 |
| A1 | 82.1 | 3054 | 80.6 | 620 | 81.4 | 1395 | 81.5 | 1798 | 81.8 | 2065 | 83.0 | 3186 | 81.0 | 631 | 82.0 | 1411 | 82.4 | 1827 | 82.7 | 2106 |

"$\delta$" specifies the number of times each example needs to be covered before it is removed, "$A$" denotes the area under the ROC curve and "$N_f$" denotes the number of features that were used for classification.

TABLE 6
Physicochemical Property Descriptors

| *P*roperty | *D*im. | *P*roperty | *D*im. | *P*roperty | *D*im. |
|---|---|---|---|---|---|
| Solvent accessible area | $\mathring{A}^2$ | Moment of Inertia | $none$ | Total accessible area | $\mathring{A}^2$ |
| Total energy | $kcal/mol$ | Total accessible volume | $\mathring{A}^3$ | Bend energy | $kcal/mol$ |
| Total Van der Waal's area | $\mathring{A}^2$ | Hbond energy | $kcal/mol$ | Total Van der Waal's volume | $\mathring{A}^3$ |
| Stretch energy | $kcal/mol$ | Dipole moment | $Debye$ | Nonbond energy | $kcal/mol$ |
| Dipole moment comp. (X, Y, Z) | $Debye$ | Estatic energy | $kcal/mol$ | Heat of formation | $Debye$ |
| Torsion energy | $kcal/mol$ | Multiplicity | $Kcal$ | Quantum total charge | $eV$ |

the first four problems (P1, P2, P3, and P4) derived from the PTC data set, the performance actually improves with feature selection. Such improvements are possible because models learned on lower-dimensional spaces will tend to have better generalization ability [25]. Also note that for some data sets, the number of features decreases as $\delta$ increases. Even though this is counter-intuitive, it can happen in the cases in which due to a higher value of $\delta$, a feature that would have been skipped is now included into the set. If this newly included feature has a relatively high support, it will contribute to the coverage of many other features. As a result, the desired level of coverage can be achieved without the inclusion of other lower-support features. Our analysis of the selected feature sets showed that for the instances in which the number of features decreases as $\delta$ increases, the selected features have indeed higher average support.

#### 5.3.4 Topological versus Geometric Subgraphs

The various results shown in Tables 2, 3, 4, and 5 also provide an indication on the relative performance of topological versus geometric subgraphs. In almost all cases, the classifier that is based on geometric subgraphs outperforms that based on topological subgraphs. For some problems, the performance advantage is marginal whereas for other problems, geometric subgraphs lead to measurable improvements in the area under the ROC curve. For example, if we consider the results shown in Table 4 for $\beta = EqCost$, we can see the geometric subgraphs lead to improvements that are at least 3 percent or higher for P2, P3, and H3, and the average improvement over all eight problems is 2.6 percent. As discussed in Section 4.1.2, these performance gains are due to the fact that a conserved geometric structure is a better indicator of a chemical compounds activity than just its topology.

### 5.4 Comparison with Other Approaches

We compared the performance of our classification algorithm against the performance achieved by three existing approaches. The first builds a traditional QSAR model based on physicochemical properties, the second uses a set of features that were derived by combining the 166 MACCS keys from MDL Inc. [2] and the Daylight fingerprints [1], and the third uses a set of substructure features that were identified by SUBDUE [20] and SubdueCL [32].

#### 5.4.1 Comparison with Physicochemical Property Descriptors

There is a wide variety of physicochemical properties that capture certain aspects of a compounds chemical activity. For our study, we have chosen a set of 18 properties that are good descriptors of the chemical activity of a compound, and most of them have been previously used for classification purposes [4]. A brief description of these properties is shown in Table 6. We used two programs to compute these attributes; the geometric attributes like solvent accessible area, total accessible area/vol, total Van der Waal's accessible area/vol were computed using the programs SASA [52], the remaining attributes were computed using Hyperchem software.

We used two different algorithms to build classification models based on these properties. The first is the C4.5 decision tree algorithm [70] that has been shown to produce good models for chemical compound classification based on physicochemical properties [4], and the second is the SVM algorithm that was used to build the classification models in our frequent substructure-based approach. Since the range of values of the different physicochemical properties can be significantly different, we first scaled them to be in the range of $[0, 1]$ prior to building the SVM model. We found that this scaling resulted in some improvements in the overall classification results. Note that C4.5 is not affected by such scaling.

Table 7 shows the results obtained by these methods for the different data sets. The values shown for SVM correspond to the area under the ROC curve and can be directly compared with the corresponding values obtained

TABLE 7
Performance of the Physicochemical Properties-Based Classifier

| *Dataset* | *SVM* | *C4.5* | | *Freq. Sub. Prec.* | |
|---|---|---|---|---|---|
| | *A* | Precision | Recall | *Topo* | *Geom* |
| P1 | 60.2 | 0.4366 | 0.1419 | 0.6972 | 0.6348 |
| P2 | 59.3 | 0.3603 | 0.0938 | 0.8913 | 0.8923 |
| P3 | 55.0 | 0.6627 | 0.1275 | 0.7420 | 0.7427 |
| P4 | 45.4 | 0.2045 | 0.0547 | 0.6750 | 0.8800 |
| H1 | 64.5 | 0.5759 | 0.1375 | 0.7347 | 0.7316 |
| H2 | 47.3 | 0.6282 | 0.4071 | 0.7960 | 0.7711 |
| H3 | 61.7 | 0.5677 | 0.2722 | 0.7827 | 0.7630 |
| A1 | 49.4 | 0.5564 | 0.3816 | 0.7676 | 0.7798 |

by our approaches (Tables 2, 3, 4, and 5). Unfortunately, since C4.5 does not produce a ranking of the training set based on its likelihood of being in the positive class, it is quite hard to obtain the ROC curve. For this reason, the values shown for C4.5 correspond to the precision and recall of the positive class for the different data sets. Also, to make the comparisons between C4.5 and our approach easier, we also computed the precision of our classifier at the same value of recall as that achieved by C4.5. These results are shown under the columns labeled "*Freq. Sub. Prec.*" for both topological and geometric features and were obtained from the results shown in Table 4 for $\beta = EqCost$. Note that the results in Table 7 for both SVM and C4.5 were obtained using the same cost-sensitive learning approach.

Comparing both the SVM-based ROC results and the precision/recall values of C4.5 we can see that our approach substantially outperforms the physicochemical properties-based classifier. In particular, our topological subgraph-based algorithm does 35 percent better compared to the SVM-based approach and 72 percent better in terms of the C4.5 precision at the same recall values. Similar results hold for the geometric subgraph based algorithm. These results are consistent with those observed by other researchers [77], [44] that showed that substructure based approaches outperform those based on physicochemical properties.

### 5.4.2 Comparison with Descriptor-Based Methods

Among the best-performing methods used by the Pharmaceutical industry to classify chemical compound data sets are those based on various topological and geometric descriptors (Section 3). To evaluate the effectiveness of these approaches and compare them against our frequent subgraph-based features, we represented each chemical compound as a feature-vector using the set of descriptors that were derived by combining the 166 MACCS keys from MDL and the Daylight fingerprints. Due to data format incompatibilities, we were only able to obtain these descriptors for the AIDS and Anthrax data sets, and we are currently investigating how to obtain them for the toxicology data set as well.

The results obtained by using the SVM classifier on this descriptor-based representation for the AIDS and Anthrax data sets are shown in Table 8. These results show the area under the ROC curve and were obtained using the same cost-sensitive learning used by our scheme. Comparing these results against those obtained by our algorithm we see that our algorithms based on either topological or geometric substructures outperform the descriptor-based approach for all four classification problems. Specifically, our topological and geometric substructure-based algorithms (Table 4 for $\beta = EqCost$) achieved ROC values that, on average, are 7.2 percent and 11.2 percent better than the descriptor-based approaches, respectively.

TABLE 8
Performance of the SVM Classifier Using MACCS Keys and Daylight Fingerprints

| Dataset | | | |
|---|---|---|---|
| H1 | H2 | H3 | A1 |
| 77.2 | 72.1 | 85.9 | 75.2 |

### 5.4.3 Comparison with SUBDUE and SubdueCL

Finally, to evaluate the advantage of using the complete set of frequent substructures over existing schemes that are based on heuristic substructure discovery, we performed a series of experiments in which we used the SUBDUE system to find the substructures and then used them for classification. Specifically, we performed two sets of experiments. In the first set, we obtain a set of substructures using the standard MDL-based heuristic substructure discovery approach of SUBDUE [38]. In the second set, we used the substructures discovered by the more recent SubdueCL algorithm [32] that guides the heuristic beam search using a scheme that measures how well a subgraph describes the positive examples in the data set without describing the negative examples.

Even though there are a number of parameters controlling SUBDUE's heuristic search algorithm, the most critical among them are the width of the beam search, the maximum size of the discovered subgraph, and the total number of subgraphs to be discovered. In our experiments, we spent a considerable amount of time experimenting with these parameters to ensure that SUBDUE was able to find a reasonable number of substructures. Specifically, we changed the width of the beam search from 4 to 50 and set the other two parameters to high numeric values. Note that in the case of the SubdueCL, in order to ensure that the subgraphs were discovered that described all the positive examples, the subgraph discovery process was repeated by increasing the value of beam-width at each iteration and removing the positive examples that were covered by subgraphs.

Table 9 shows the performance achieved by SUBDUE and SubdueCL on the eight different classification problems along with the number of subgraphs that it generated and the amount of time that it required to find these subgraphs. These results were obtained by using the subgraphs discovered by either SUBDUE or SubdueCL as features in

TABLE 9
Performance of the SUBDUE and SubdueCL-Based Approaches

| *Dataset* | *SUBDUE* | | | *SubdueCL* | | |
|---|---|---|---|---|---|---|
| | *A* | $N_f$ | $Time_p$ | *A* | $N_f$ | $Time_p$ |
| P1 | 61.9 | 1288 | 303sec | 63.5 | 2103 | 301sec |
| P2 | 64.2 | 1374 | 310sec | 63.3 | 2745 | 339sec |
| P3 | 57.4 | 1291 | 310sec | 59.6 | 1772 | 301sec |
| P4 | 58.5 | 1248 | 310sec | 60.8 | 2678 | 324sec |
| H1 | 74.2 | 1450 | 1,608sec | 73.8 | 960 | 1002sec |
| H2 | 58.5 | 901 | 232,006sec | 65.2 | 2999 | 476,426sec |
| H3 | 71.3 | 905 | 178,343sec | 77.5 | 2151 | 440,416sec |
| A1 | 75.3 | 983 | 56,056sec | 75.9 | 1094 | 31,177sec |

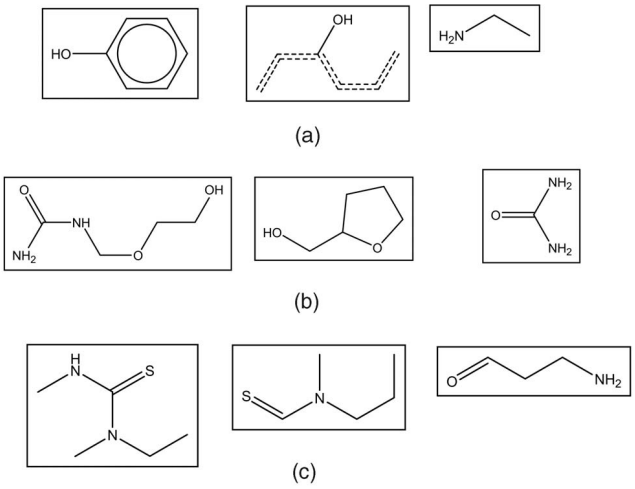


Fig. 4. The three most discriminating substructures for the (a) PTC, (b) AIDS, and (c) Anthrax data sets.

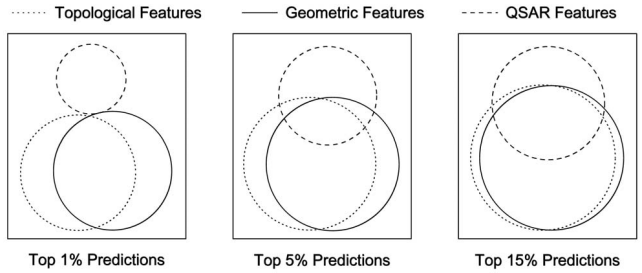


Fig. 5. Venn diagrams displaying the relation between the positive examples that were correctly classified by the three approaches at different cutoff values for the Anthrax data set. The different cutoffs were obtained by looking at only the top 1 percent, 5 percent, and 15 percent of the ranked predictions. Each circle in the Venn diagram corresponds to one of the three classification schemes and the size of the circle indicates the number of positive examples correctly identified. The overlap between two circles indicates the number of common correct predictions.

an SVM-based classification model. Essentially, our SUBDUE and SubdueCL classifiers have the same structure as our frequent subgraph-based classifiers with the only difference being that the features now correspond to the subgraphs discovered by SUBDUE and SubdueCL. Moreover, to make the comparisons as fair as possible we used $\beta = EqCost$ as the misclassification cost. We also performed another set of experiments in which we used the rule-based classifier produced by SubdueCL. The results of this scheme was inferior to those produced by the SVM-based approach and we are not reporting them here.

Comparing SUBDUE against SubdueCL, we can see that the latter achieves better classification performance, consistent with the observations made by other researchers [32]. Comparing the SUBDUE and SubdueCL-based results with those obtained by our approach (Tables 2, 3, 4, and 5), we can see that in almost all cases both our topological and geometric frequent subgraph-based algorithms lead to substantially better performance. This is true both in the cases in which we performed no feature selection as well as in the cases in which we used the sequential covering-based feature selection scheme. In particular, comparing the SubdueCL results against the results shown in Table 5 without any feature selection we can see that on the average, our topological and geometric subgraph based algorithms do 9.3 percent and 12.2 percent better, respectively. Moreover, even after feature selection with $\delta = 15$ that result in a scheme that have comparable number of features as those used by SubdueCL, our algorithms are still better by 9.7 percent and 13.7 percent, respectively. Finally, if we compare the amount of time required by either SUBDUE or SubdueCL to that required by the FSG algorithm to find all frequent subgraphs (last column of Table 2), we can see that despite the fact that we are finding the complete set of frequent subgraphs our approach requires substantially less time.

## 6 Conclusions and Directions for Future Research

In this paper, we presented a highly effective algorithm for classifying chemical compounds based on frequent substructure discovery that can scale to large data sets. Our experimental evaluation showed that our algorithm leads to substantially better results than those obtained by existing descriptor and substructure-based methods. Moreover, besides this improved classification performance, the substructure-based nature of this scheme provides to the chemists valuable information as to which substructures are most critical for the classification problem at hand. For example, Fig. 4 shows the three most discriminating substructures for the PTC, DTP AIDS, and Anthrax data sets that were obtained by analyzing the decision hyperplane produced by the SVM classifier.[1] A chemist can then use this information to understand the models and potentially use it to design better compounds.

The classification algorithms presented in this paper can be improved along three different directions. First, as already discussed in Section 2, our current geometric graph representation utilizes a single conformation of the chemical compound and we believe the overall classification performance can be improved by using all possible low-energy conformations. Such conformations can be obtained from existing 3D coordinate prediction software and as discussed in Section 4.1.2 can be easily incorporated in our existing framework. Second, our current feature selection algorithms only focus on whether or not a particular substructure is contained in a compound and they do not take into account how these fragments are distributed over different parts of the molecule. Better feature selection algorithms can be developed by taking this information into account so that to ensure that the entire (or most of) molecule is covered by the selected features. Third, even though the proposed approaches significantly outperformed that based on physicochemical property descriptors, our analysis showed that there is a significant difference as to which compounds are correctly classified by these two approaches. For example, Fig. 5 shows the overlap among the different correct predictions produced by the geometric, topological, and QSAR-based (using the various physicochemical property descriptors) methods at different cutoff values for the Anthrax data set. From these results, we can see that there is a great agreement between the substructure-based approaches, but there is a large difference among the compounds that are correctly predicted by the QSAR approach, especially at the top 1 percent

1. These features correspond to the highest-weight dimensions of the decision hyperplane produced by a linear SVM model. Since each compound is a vector in $\mathcal{R}^+$, the highest-weight dimensions of the decision hyperplane correlate well with the dimensions of the underlying data set that contribute the most to its assignment in the positive class [73].

and 5 percent. These results suggest that better results can be potentially obtained by combining the substructure and QSAR-based approaches.

## ACKNOWLEDGMENTS

The authors will like to thank Dr. Ian Watson from Lilly Research Laboratories and Dr. Peter Henstock from Pfizer Inc. for providing them with the various fingerprints used in the experimental evaluation and for the numerous discussions on the practical aspects of virtual screening. This work was supported by the US National Science Foundation EIA-9986042, ACI-9982274, ACI-0133464, ACI-0312828, IIS-0431135, the Army High Performance Computing Research Center contract number DAAD19-01-2-0014, and by the Digital Technology Center at the University of Minnesota.

## REFERENCES

[1] Daylight Inc., Mission Viejo, Calif., http://www.daylight.com, 2005.
[2] MDL Information Systems Inc., San Leandro, Calif., http://www.mdl.com, 2005.
[3] G.W. Adamson, J. Cowell, M.F. Lynch, A.H. McLure, W.G. Town, and A.M. Yapp, "Strategic Considerations in the Design of a Screening System for Substructure Searches of Chemical Structure File," *J. Chemical Documentation,* 1973.
[4] A. An and Y. Wang, "Comparisons of Classification Methods for Screening Potential Compounds," *Proc. Int'l Conf. Data Mining,* 2001.
[5] T.A. Andrea and H. Kalayeh, "Applications of Neural Networks in Quantitative Structure-Activity Relationships of Dihydrofolate Reductase Inhibitors," *J. Medicinal Chemistry,* vol. 34, pp. 2824-2836, 1991.
[6] M.J. Ashton, M.C. Jaye, and J.S. Mason, "New Perspectives in Lead Generation II: Evaluating Molecular Diversity," *Drug Discovery Today,* 1996.
[7] J. Bajorath, "Integration of Virtual and High Throughput Screening," *Nature Rev. Drug Discovery,* 2002.
[8] J.M. Barnard, G.M. Downs, and P. Willet, "Descriptor-Based Similarity Measures for Screening Chemical Databases," *Virtual Screening for Bioactive Molecules,* H.J. Bohm and G. Schneider, eds., Wiley-VCH, 2000.
[9] S.C. Basak, V.R. Magnuson, J.G. Niemi, and R.R. Regal, "Determining Structural Similarity of Chemicals Using Graph Theoretic Indices," *Discrete Applied Math.,* 1988.
[10] G.W. Bemis and M.A. Murcko, "The Properties of Known Drugs. 1. Molecular Frameworks," *J. Medicinal Chemistry,* vol. 39, no. 15, pp. 2887-2893, 1996.
[11] G.W. Bemis and M.A. Murcko, "The Properties of Known Drugs. 2. Side Chains," *J. Medicinal Chemistry,* vol. 42, no. 25, pp. 5095-5099, 1999.
[12] K.H. Bleicher, H.-J. Bohm, K. Muller, and A.I. Alanine, "Hit and Lead Generation: Beyond High Throughput Screening," *Nature Rev. Drug Discovery,* 2003.
[13] H.J. Bohm and G. Schneider, *Virtual Screening for Bioactive Molecules.* Wiley-VCH, 2000.
[14] G. Bravi, E. Gancia, D. Green, V.S. Hann, and M. Mike, "Modelling Structure-Activity Relationship," *Virtual Screening for Bioactive Molecules,* H.J. Bohm and G. Schneider, eds., Wiley-VCH, 2000.
[15] R. Brown and Y. Martin, "The Information Content of 2D and 3D Structural Descriptors Relevant to Ligand-Receptor Binding," *J. Chemical Information and Computer Science,* vol. 37, no. 1, pp. 1-9, 1997.
[16] E. Byvatov, U. Fechner, J. Sadowski, and G. Schneider, "Comparison of Support Vector Machine and Artificial Neural Network Systems for Drug/Nondrug Classification," *J. Chemical Information and Computer Science,* vol. 43, no. 6, pp. 1882-1889, 2003.
[17] R.E. CarHart, D.H Smith, and R. Venkataraghavan, "Atom Pairs as Molecular Features in Structure-Activity Studies: Definition and Applications," *J. Chemical Information and Computer Science,* 1985.
[18] X. Chen, A. Rusinko, and S.S. Young, "Recursive Partitioning Analysis of a Large Structure-Activity Data Set Using Three-Dimensional Descriptors," *J. Chemical information and Computer Science,* 1998.
[19] M.R. Berthold and C. Borgelt, "Mining Molecular Fragments: Finding Relevant Substructures of Molecules," *Proc. Int'l Conf. Data Mining,* 2002.
[20] D.J. Cook and L.B. Holder, "Graph-Based Data Mining," *IEEE Intelligent Systems,* vol. 15, no. 2, pp. 32-41, 2000.
[21] R.D King, A. Srinivasan, and L. Dehaspe, "Warmr: A Data Mining Tool for Chemical Data," *J. Computer Aided Molecular Design,* vol. 15, pp. 173-181, 2001.
[22] E.K. Davies, "Molecular Diversity and Combinatorial Chemistry: Libraries and Drug Discovery," *Am. Chemical Soc.,* 1996.
[23] L. Dehaspe, H. Toivonen, and R.D. King, "Finding Frequent Substructures in Chemical Compounds," *Proc. Fourth Int'l Conf. Knowledge Discovery and Data Mining,* pp. 30-36, 1998.
[24] M. Deshpande and G. Karypis, "Automated Approaches for Classifying Structure," *Proc. Second ACM SIGKDD Workshop Data Mining in Bioinformatics,* 2002.
[25] M. Deshpande and G. Karypis, "Using Conjunction of Attribute Values for Classification," *Proc. 11th ACM Conf. Information and Knowledge Management,* pp. 356-364, 2002.
[26] M. Deshpande, M. Kuramochi, and G. Karypis, "Frequent Sub-Structure Based Approaches for Classifying Chemical Compounds," *Proc. 2003 IEEE Int'l Conf. Data Mining (Int'l Conf. Data Mining),* pp. 35-42, 2003.
[27] J. Devillers, *Neural Networks in QSAR and Drug Design.* London: Acamedic Press, 1996.
[28] dtp.nci.nih.gov., DTP AIDS Antiviral Screen Data Set, 2005.
[29] B. Dunkel and N. Soparkar, "Data Organizatinon and Access for Efficient Data Mining," *Proc. 15th IEEE Int'l Conf. Data Eng.,* Mar. 1999.
[30] H. Gao, C. Williams, P. Labute, and J. Bajorath, "Binary Quantitative Structure-Activity Relationship (QSAR) Analysis of Estrogen Receptor Ligands," *J. Chemical Information and Computer Science,* 1999.
[31] J. Gasteiger, C. Rudolph, and J. Sadowski, "Automatic Generation of 3D-Atomic Coordinates for Organic Molecules," *Tetrahedron Comp. Method,* vol. 3, pp. 537-547, 1990.
[32] J. Gonzalez, L. Holder, and D. Cook, "Application of Graph Based Concept Learning to the Predictive Toxicology Domain," *Proc. Pacific Telecomm. Conf., Workshop at the Fifth Principles and Practice of Knowledge Discovery in Databases Conf.,* 2001.
[33] A.C. Good, J.S. Mason, and S.D. Pickett, "Pharmacophore Pattern Application in Virtual Screening, Library Design and QSAR," *Virtual Screening for Bioactive Molecules,* H.J. Bohm and G. Schneider, eds., Wiley-VCH, 2000.
[34] L.H. Hall and L.B. Kier, "Electrotopological State Indices for Atom Types: A Novel Combination of Electronic, Topological, and Valence State Information," *J. Chemical Information and Computer Science,* 1995.
[35] J.S. Handen, "The Industrialization of Drug Discovery," *Drug Discovery Today,* vol. 7, no. 2, pp. 83-85, Jan. 2002.
[36] C. Hansch, P.P. Maolney, T. Fujita, and R.M. Muir, "Correlation of Biological Activity of Phenoxyacetic Acids with Hammett Substituent Constants and Partition Coefficients," *Nature,* vol. 194, pp. 178-180, 1962.
[37] C. Hansch, R.M. Muir, T. Fujita, C.F. Maloney, and M. Streich, "The Correlation of Biological Activity of Plant Growth-Regulators and Chloromycetin Derivatives with Hammett Constants and Partition Coefficients," *J. Am. Chemical Soc.,* vol. 85, pp. 2817-1824, 1963.
[38] L. Holder, D. Cook, and S. Djoko, "Substructure Discovery in the Subdue System," *Proc. AAAI Workshop Knowledge Discovery in Databases,* pp. 169-180, 1994.
[39] J. Huan, W. Wang, and J. Prins, "Efficient Mining of Frequent Subgraph in the Presence of Isomophism," *Proc. 2003 IEEE Int'l Conf. Data Mining (Int'l Conf. Data Mining '03),* 2003.
[40] A. Inokuchi, T. Washio, and H. Motoda, "An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data," *Proc. Fourth European Conf. Principles and Practice of Knowledge Discovery in Databases (PKDD '00),* pp. 13-23, Sept. 2000.
[41] T. Joachims, *Advances in Kernel Methods: Support Vector Learning.* MIT-Press, 1999.
[42] G. Karypis, "CLUTO a Clustering Toolkit," Technical Report 02-017, Dept. of Computer Science, Univ. of Minnesota, http://www.cs.umn.edu/cluto, 2002.

[43] L. Kier and L. Hall, *Molecular Structure Description.* Academic Press, 1999.
[44] R.D. King, S.H. Muggleton, A. Srinivasan, and M.J.E. Sternberg, "Strucutre-Activity Relationships Derived by Machine Learning: The Use of Atoms and Their Bond Connectivities to Predict Mutagenecity BYD Inductive Logic Programming," *Proc. Nat'l Academy of Science,* vol. 93, pp. 438-442, Jan. 1996.
[45] R.D. King, S. Muggleton, R.A. Lewis, and J.E. Sternberg, "Drug Design by Machine Learning: The Use of Inductive Logic Programming to Model the Sturcture-Activity Relationships of Trimethoprim Analogues Binding to Dihydrofolate Reductase," *Proc. Nat'l Academy of Science,* vol. 89, pp. 11322-11326, Dec. 1992.
[46] S. Kramer, L. De Raedt, and C. Helma, "Molecular Feature Mining in HIV Data," *Proc. Seventh Int'l Conf. Knowledge Discovery and Data Mining,* 2001.
[47] M. Kuramochi and G. Karypis, "Frequent Subgraph Discovery," *Proc. IEEE Int'l Conf. Data Mining,* 2001, also available as a UMN-CS Technical Report, TR# 01-028.
[48] M. Kuramochi and G. Karypis, "Discovering Geometric Frequent Subgraph," *Proc. IEEE Int'l Conf. Data Mining,* 2002, also available as a UMN-CS Technical Report, TR# 02-024.
[49] M. Kuramochi and G. Karypis, "Discovering Frequent Geometric Subgraphs," Technical Report 04-039, Dept. of Computer Science, Univ. of Minnesota, 2004.
[50] M. Kuramochi and G. Karypis, "An Efficient Algorithm for Discovering Frequent Subgraphs," *IEEE Trans. Knowledge and Data Eng.,* vol. 16, no. 9, pp. 1038-1051, Sept. 2004.
[51] P. Labute, "Binary QSAR: A New Method for the Determination of Quantitative Structure Activity Relationships," *Proc. Pacific Symp. Biocomputing,* 1999.
[52] S.M. Le Grand and J.K.M. Merz, "Rapid Approximation to Molecular Surface Area via the Use of Boolean Logic Look-Up Tables," *J. Computational Chemistry,* vol. 14, pp. 349-352, 1993.
[53] A.R. Leach, *Molecular Modeling: Principles and Applications.* Englewood Cliffs, N.J.: Prentice Hall, 2001.
[54] X.Q. Lewell, D.B. Judd, S.P. Watson, and M.M. Hann, "RECAP Retrosynthetic Combinatorial Analysis Procedure: A Powerful New Technique for Identifying Privileged Molecular Fragments with Useful Applications in Combinatorial Chemistry," *J. Chemical Information and Computer Science,* vol. 38, no. 3, pp. 511-522, 1998.
[55] W. Li, J. Han, and J. Pei, "CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules," *Proc. IEEE Int'l Conf. Data Mining,* 2001.
[56] B. Liu, W. Hsu, and Y. Ma, "Integrating Classification and Association Rule Mining," *Proc. Fourth Int'l Conf. Knowledge Discovery and Data Mining,* 1998.
[57] D.J. Livingstone, *Neural Networks in QSAR and Drug Design.* London: Acamedic Press, 1996.
[58] D.J. Livingstone, "The Characterization of Chemical Structures Using Molecular Properties. A Survey," *J. Chemical Information and Computer Science,* 2000.
[59] T.M. Mitchell, *Machine Learning.* McGraw Hill, 1997.
[60] K. Morik, P. Brockhausen, and T. Joachims, "Combining Statistical Learning with a Knowledge-Based Approach—A Case Study in Intensive Care Monitoring," *Proc. Int'l Conf. Machine Learning,* 1999.
[61] S. Muggleton, "Inverse Entailment and Progol," *New Generation Computing,* vol. 13, pp. 245-286, 1995.
[62] S. Muggleton and L. De Raedt, "Inductive Logic Programming: Theory and Methods," *J. Logic Programming,* vol. 19, no. 20, pp. 629-679, 1994.
[63] S.H. Muggleton and C. Feng, "Efficient Induction of Logic Programs," *Inductive Logic Programming,* S. Muggleton, ed., pp. 281-298, London: Academic Press, 1992.
[64] C.A. Nicalaou, S.Y. Tamura, B.P. Kelley, S.I. Bassett, and R.F. Nutt, "Analysis of Large Screening Data Sets via Adaptively Grown Phylogenetic-Like Trees," *J. Chemical Information and Computer Science,* 2002.
[65] S. Nijssen and J.N. Kok, "A Quickstart in Frequent Structure Mining Can Make a Difference," *Proc. 10th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD-2004),* pp. 647-652, Aug. 2004.
[66] R. Nilakantan, N. Bauman, S. Dixon, and R. Venkataraghavan, "Topological Torsion: A New Molecular Descriptor for SAR Applications. Comparison with Other Descriptors," *J. Chemical Information and Computer Science,* 1987.
[67] M. Otto, *Chemometrics.* Wiley-VCH, 1999.
[68] S.D. Pickett, J.S. Mason, and I.M. McLay, "Diversity Profiling and Design Using 3D Pharmacophores: Pharmacophore-Derived Queries (PDG)," *J. Chemical information and Computer Science,* 1996.
[69] F. Provost and T. Fawcett, "Robust Classification for Imprecise Environments," *Machine Learning,* vol. 42, no. 3, 2001.
[70] J. Ross Quinlan, *C4.5: Programs for Machine Learning.* San Mateo, Calif.: Morgan Kaufmann, 1993.
[71] G.W. Richards, "Virtual Screening Using Grid Computing: The Screensaver Project," *Nature Rev.: Drug Discovery,* vol. 1, pp. 551-554, July 2002.
[72] A. Rusinko, M.W. Farmen, C.G. Lambert, P.L. Brown, and S.S. Young, "Analysis of a Large Structure/Biological Activity Data Set Using Recursive Partitioning," *J. Chemical Information and Computer Science,* 1999.
[73] B. Scholkopf and A. Smola, *Learning with Kernels.* Boston, Mass.: MIT Press, 2002.
[74] P. Shenoy, J.R. Haritsa, S. Sundarshan, G. Bhalotia, M. Bawa, and D. Shah, "Turbo-Charging Vertical Mining of Large Databases," *Proc. ACM SIGMOD Int'l Conf. Management of Data,* pp. 22-33, May 2000.
[75] R.P. Sheridan, M.D. Miller, D.J. Underwood, and S.J. Kearsley, "Chemical Similarity Using Geometric Atom Pair Descriptors," *J. Chemical Information and Computer Science,* 1996.
[76] A. Srinivasan, R.D. King, S.H. Muggleton, and M. Sternberg, "The Predictive Toxicology Evaluation Challenge," *Proc. 15th Int'l Joint Conf. Artificial Intelligence (IJCAI-97),* pp. 1-6, 1997.
[77] A. Sriniviasan and R. King, "Feature Construction with Inductive Logic Programming: A Study of Quantitative Predictions of Biological Activity Aided by Structural Attributes," *Knowledge Discovery and Data Mining J.,* vol. 3, pp. 37-57, 1999.
[78] S.Y. Tamura, P.A. Bacha, H.S. Gruver, and R.F. Nutt, "Data Analysis of High-Throughput Screening Results: Application of Multidomain Clustering to the NCI Anti-HIV Data Set," *J. Medicinal Chemistry,* 2002.
[79] V. Vapnik, *Statistical Learning Theory.* New York: John Wiley, 1998.
[80] P. Willett, "Chemical Similarity Searching," *J. Chemical Information and Computer Science,* vol. 38, no. 6, pp. 983-996, 1998.
[81] S. Wold, E. Johansson, and M. Cocchi, *3D QSAR in Drug Design: Theory, Methods and Application.* ESCOM Science Publishers B.V., 1993.
[82] X. Yan and J. Han, "gSpan: Graph-Based Substructure Pattern Mining," *Proc. Int'l Conf. Data Mining,* 2002.
[83] M.J. Zaki and K. Gouda, "Fast Vertical Mining Using Diffsets," Technical Report 01-1, Dept. of Computer Science, Rensselaer Polytechnic Inst., 2001.
[84] M. Javeed Zaki, "Scalable Algorithms for Association Mining," *Knowledge and Data Eng.,* vol. 12, no. 2, pp. 372-390, 2000.
[85] J. Zupan and J. Gasteiger, *Neural Networks for Chemists.* VCH Publisher, 1993.

**Mukund Deshpande** received the masters degree in engineering (ME) in system science and automation from the Indian Institute of Science, Bangalore, India in 1997. He received the PhD degree from the Department of Computer Science at the University of Minnesota in 2004, and is currently working at Oracle Corporation.

**Michihiro Kuramochi** received the BEng and MEng degrees from the University of Tokyo, and the MS degree from Yale University. He is a graduate student at the University of Minnesota, Twin Cities.

**Nikil Wale** received the bachelor's degree in electrical engineering from the National Institute of Technology, Warangal, in 2001. He is pursuing the PhD degree in computer science at the University of Minnesota, Twin Cities. His research interests include data mining, machine learning, and bioinformatics. Currently, he is working on the various aspects of virtual screening in drug discovery.

**George Karypis** received the PhD degree in computer science from the University of Minnesota and he is currently an associate professor in the Department of Computer Science and Engineering at the University of Minnesota. His research interests spans the areas of parallel algorithm design, data mining, bioinformatics, information retrieval, applications of parallel processing in scientific computing and optimization, sparse matrix computations, parallel preconditioners, and parallel programming languages and libraries. His research has resulted in the development of software libraries for serial and parallel graph partitioning (METIS and ParMETIS), hypergraph partitioning (hMETIS), for parallel Cholesky factorization (PSPASES), for collaborative filtering-based recommendation algorithms (SUGGEST), clustering high-dimensional data sets (CLUTO), and finding frequent patterns in diverse data sets (PAFI). He has coauthored more than 90 journal and conference papers on these topics and a book titled *Introduction to Parallel Computing* (Addison Wesley, 2003, second edition). In addition, he is serving on the program committees of many conferences and workshops on these topics and is an associate editor of the *IEEE Transactions on Parallel and Distributed Systems*. He is a senior member of the IEEE Computer Society.