Training Similarity Measures for Specific Activities: Application to Reduced Graphs

Kristian Birchall,[†] Valerie J. Gillet,*[,†] Gavin Harper,[‡] and Stephen D. Pickett[‡]

Department of Information Studies, University of Sheffield, Western Bank, Sheffield S10 2TN, United Kingdom, and GlaxoSmithKline, Gunnels Wood Road, Stevenage, SG1 2NY, United Kingdom

Received October 20, 2005

Reduced graph representations of chemical structures have been shown to be effective in similarity searching applications where they offer comparable performance to other 2D descriptors in terms of recall experiments. They have also been shown to complement existing descriptors and to offer potential to scaffold hop from one chemical series to another. Various methods have been developed for quantifying the similarity between reduced graphs including fingerprint approaches, graph matching, and an edit distance method. The edit distance approach quantifies the degree of similarity of two reduced graphs based on the number and type of operations required to convert one graph to the other. An attractive feature of the edit distance method is the ability to assign different weights to different operations. For example, the mutation of an aromatic ring node to an acyclic node may be assigned a higher weight than the mutation of an aromatic ring to an aliphatic ring node. In this paper, we describe a genetic algorithm (GA) for training the weights of the different edit distance operations. The method is applied to specific activity classes extracted from the MDDR database to derive activity-class specific weights. The GA-derived weights give substantially improved results in recall experiments as compared to using weights assigned on intuition. Furthermore, such activity specific weights may provide useful structure−activity information for subsequent design efforts. In a virtual screening setting when few active compounds are known, it may be more useful to have weights that perform well across a variety of different activity classes. Thus, the GA is also trained on multiple activity classes simultaneously to derive a generalized set of weights. These more generally applicable weights also represent a substantial improvement on previous work.

INTRODUCTION

Similarity methods are widely used in drug discovery programs in applications such as similarity searching and clustering.[1−5] The rationale for similarity methods stems from the similar property principle which states that structurally similar molecules are likely to have similar properties.[6] Calculation of pairwise molecular similarity requires the use of molecular descriptors to represent the compounds and a method for quantifying the similarity based on the descriptors.[7] The wide variety of similarity methods that has been developed reflects the complex relationships that exist between structural characteristics and biological activity.[8]

Reduced graphs are one of a number of different types of descriptor that have been developed recently for similarity searching applications. In previous work, they have been shown to provide comparable performance to other 2D descriptors measured in recall experiments.[9−11] They have also been shown to complement the more traditional 2D descriptors and to offer potential in scaffold hopping applications[9,12−15] that have become a recent focus for similarity methods. Scaffold hopping refers to the identification of compounds that share the same biological activity as a query compound but that belong to different chemical series;[16] they therefore provide patent opportunities and also allow multiple series to be explored in parallel. Reduced graphs are summary representations of structures in which groups of connected atoms that share some characteristic are collapsed into a single node with the topology between the resulting nodes being retained. When used in the context of similarity searching, the aim of the graph reduction process is to represent the features of a structure that may be important in forming interactions with a receptor. Therefore, nodes are generated to represent features such as hydrogen bond donors, hydrogen bond acceptors, charged groups, ring systems, etc. The reduced graph can therefore be considered as a topological pharmacophore.

Various methods have been devised for measuring the similarity of two structures based on their reduced graphs including mapping the reduced graphs to a fingerprint,[9,10] using graph matching techniques to compare the reduced graphs directly,[14,17,18] and using a method based on edit distances.[11] The use of conventional fingerprints to represent chemical structures, such as Daylight fingerprints, is well-known, and similar techniques can also be applied to generate fingerprint representations of reduced graphs. Whereas in the former case the fingerprints represent paths consisting of atoms and bonds, in the latter case, they represent paths consisting of nodes and edges in the reduced graph. Such fingerprints have been used successfully for comparing reduced graphs.[9] However, the different characteristics of reduced graphs, relative to the structures from which they are derived, suggest that fingerprinting methods derived for handling chemical structures may not be ideal when applied to reduced graphs. By definition, reduced graphs consist of fewer nodes than the chemical structures from which they

* To whom correspondence should be sent. E-mail: v.gillet@sheffield.ac.uk. Tel: +44 1142 222 652.
[†] University of Sheffield.
[‡] GlaxoSmithKline.

10.1021/ci050465e CCC: $33.50
Published on Web 01/27/2006

are derived with the result that the fingerprints can be quite sparse. The small number of node types present in reduced graphs can result in repeated patterns occurring, and multiple occurrences are not usually represented in a conventional binary fingerprint. Perhaps more significantly, small changes in a chemical structure can result in large changes in the paths that are present in a reduced graph; for example, the insertion of a heteroatom in an acyclic chain can result in the insertion of a node into a reduced graph. As a result, the fingerprint representations of the two structures may be quite different. Finally, the node definitions themselves may cause problems. For example, in the definitions described by Gillet et al.,[9] a functional group that can act as both hydrogen bond donor and acceptor is assigned a distinct node type from either a donor node or an acceptor node. The relationship between these nodes would not be recognized in a simple fingerprinting scheme.

In recognition of these limitations, different fingerprinting methods have been devised that are more appropriate to reduced graphs; for example, binary vectors have been used to represent the presence/absence of node pairs and also to take account of the frequency of occurrence of reduced graph fragments.[10] In addition, the small size of the reduced graphs relative to their parent structures means that it is possible to use other more computationally expensive techniques with greater discriminatory power.[18] Thus, Harper et al. developed a method based on edit distances.[11] The edit distance technique is well-known in computational biology where it is used for sequence comparisons with the similarity between two sequences being related to the number of operations required to change one sequence to another. The allowed operations include mutation, deletion, and insertion of the objects in a sequence. Harper et al. developed an edit distance method to calculate the similarity of reduced graphs with the similarity between two reduced graphs being related to the cost of converting one reduced graph to the other. The operations performed on a reduced graph include mutation of one node type to another and insertion and deletion of a node. The edit distance method therefore copes naturally with the insertion problem in reduced graphs. Furthermore, different weights can be assigned to different node operations to reflect similarities in node types. For example, the cost associated with converting a donor to a donor and acceptor node was assigned a low weight; whereas, the cost of converting a donor to a negatively ionizable group was assigned a high cost. Harper showed that the edit distance similarity measure combined with a new fingerprint method was more effective at identifying similarities between reduced graphs than the earlier fingerprinting methods.

The edit distance weights used in Harper's method were assigned by intuition and it was recognized that they may be suboptimal. Here, we have developed a method for optimizing weights based on maximizing recalls using training data. Weights can be optimized on individual activity classes to derive activity-class specific weights and also on several classes simultaneously to derive generalized weights that may be more appropriate when little is known about the desired activity. We show that the optimized weights do indeed lead to enhanced performance of the edit distance similarity measure. Furthermore, the weights may also give clues as to the structure−activity relationship that exists within an activity class.

Table 1. Key to Node Types[a]

node type	heavy atom label	symbol
	Aromatic	
none	Sc	
donor	Ti	
acceptor	V	ar
donor & acceptor	Cr	
+ ve ionizable	Mn	
− ve ionizable	Fe	
	Aliphatic	
none	Hf	
donor	Ta	
acceptor	W	
donor & acceptor	Re	al
+ ve ionizable	Y	
− ve ionizable	Zr	
	Acyclic	
donor	Co	d
acceptor	Ni	a
donor & acceptor	Cu	a&d
+ ve ionizable	Nb	+ ve
− ve ionizable	Mo	− ve
linker	Zn	li
single bond	−	
double bond	=	fusion

[a] A double bond indicates ring fusion. The reduced graph nodes are represented by heavy atom labels. The symbols listed in the final column are used later in the paper to identify node type or a group of nodes.

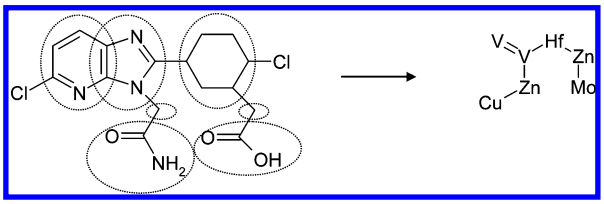

Figure 1. Example reduced graph. The node definitions are given in Table 1.

The paper is organized as follows. First, the edit distance method developed by Harper is described. Next we outline the genetic algorithm (GA) developed to optimize the weights together with some experiments that were designed to establish its robustness and to develop a protocol for subsequent runs. We then used the GA to derive activity-class specific weights. Finally, we present results aimed at optimizing weights over all the activity classes.

METHODS

Reduced Graph Definition. The reduced graphs used in this work are as described by Harper et al. and consist of the following basic node types: hydrogen bond acceptor, hydrogen bond donor, positively ionizable groups, negatively ionizable groups, aromatic rings, and aliphatic rings. Adjacent donor and acceptor features are merged into a distinct joint donor and acceptor node, and ring nodes are also combined with feature type as shown in Table 1. Positively ionizable groups and negatively ionizable groups take precedence over donors and acceptors. (The heavy atom codes used in the table enable the reduced graphs to be handled by software designed to handle chemical structures and are retained here to enable the current work to be related to Harper's work.) An example reduced graph is shown in Figure 1.

Edit Distance Calculation. Calculation of the edit distance between a query reduced graph (Q) and a database reduced

Table 2. Weights Used by Harper et al.[11] [a]

Section A: Mutations

	aromatic ring {Cr, Mn, Fe}	aliphatic ring {Hf, Ta, W, Re, Y, Zr}	Nb	Mo	Co	Ni	Cu	Zn	—	=
aromatic ring {Sc, Ti, V, Cr, Mn, Fe}	**1**	**2**	**2**	**2**	**2**	**2**	**2**	**2**	2	**3**
aliphatic ring {Hf, Ta,W, Re, Y, Zr}	2	**2**	**2**	**2**	**2**	**2**	**2**	**2**	2	**3**
Nb	2	2	0	**2**	**2**	**2**	**2**	**2**	2	**3**
Mo	2	2	2	0	**2**	**2**	**2**	**2**	2	**3**
Co	2	2	2	2	0	**2**	**1**	**2**	2	**3**
Ni	2	2	2	2	2	0	**1**	**2**	2	**3**
Cu	2	2	2	2	1	1	0	**2**	2	**3**
Zn	2	2	2	2	2	2	2	0	2	**3**
— (single bond)	2	2	2	2	2	2	2	2	0	3
= (double bond)	3	3	3	3	3	3	3	3	3	0

Section B: Insertions/Deletions

feature	cost	feature	cost	feature	cost
aromatic ring {Sc, Ti, V, Cr, Mn, Fe}	**2**	Co	**2**	— (single bond)	0
aliphatic ring {Hf, Ta, W, Re, Y, Zr}	**2**	Ni	**2**	= (double bond)	**3**
Nb	**2**	Cu	**2**		
Mo	**2**	Zn	**1**		

[a] The key to the node types is given in Table 1. Section A indicates mutation weights. For example, the weight incurred when an acyclic donor (Co) is substituted for an acyclic donor and acceptor feature (Cu) is 1. Weights for insertions and deletions are given in Section B. For example, the weight for the insertion or deletion of an aromatic ring node (of any subtype) is 2. Note that the weights consist of integers in the range 1 ... 3 and that most of the weights are assigned a value of 2. The boldface type indicates the weights to be optimized by the GA as referred to later in the text.

graph (S) involves comparing each path in Q with each path in S. The weighted edit distance between two paths extracted from the reduced graphs is based on the minimum number of operations (node mutations, insertions, and deletions) required to convert one path to the other. Each of the operations is assigned a weight that depends on the node types involved and the specific operation carried out. The weighted edit distance is calculated using dynamic programming techniques.[19]

The set of paths for a reduced graph is defined as all linear shortest paths between nodes of degree one (these are referred as a maximal paths). The cost of each path in Q is assigned as the minimum weighted edit distance over all paths in S. Similarly, the cost of each path in S is defined as the minimum weighted edit distance when compared with all paths in Q. The edit distance between the two reduced graphs Q and S is then defined as the maximum cost path considering all paths in Q and S.

The edit distance is transformed into a similarity measure by first normalizing the distance to be in the range 0 ... 1 followed by subtraction from 1. Normalization is required since the magnitude of the edit distance is dependent on the number of nodes in the reduced graphs, with larger reduced graphs being more likely to have greater numbers of mismatches and hence larger edit distances. Thus, the edit distance is divided by the number of nodes in the smaller of the two reduced graphs. The magnitude of the weights also affects the value of the edit distance. Since most of the weights used by Harper are 2, the edit distance is divided by 2 to normalize for this effect. The distance measure is then converted to a more conventional measure of similarity by subtracting it from 1. Negative values, resulting from a normalized edit distance >1 are assigned zero to ensure all similarities are in the range 0 ... 1.

The node types and weights for insertions, deletions and mutations used by Harper et al. are given in Tables 1 and 2. The procedure for calculating the reduced graph edit distance is illustrated in Figure 2. Note that the edit distance method is based on identifying paths and is therefore unable to process reduced graphs that contain cycles which can arise from structures that contain complex fused rings.

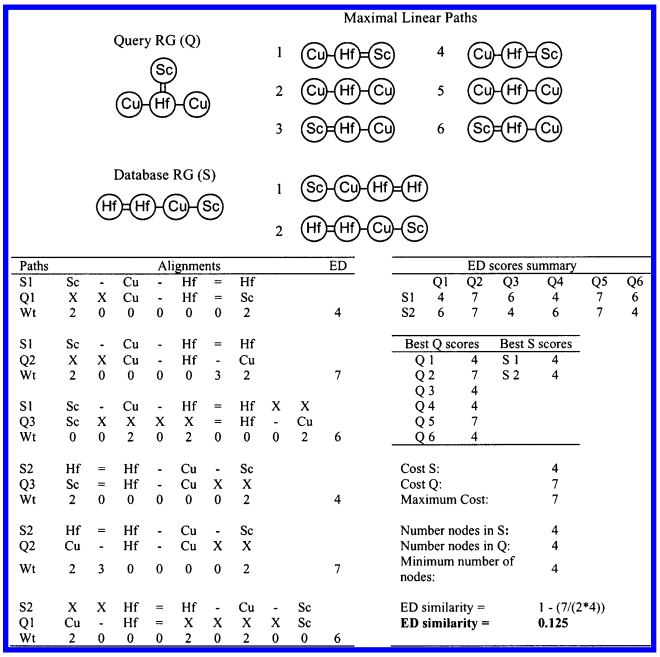

Figure 2. See text for details.

Optimizing Penalty Weights. A GA has been developed to optimize the weights for insertions, deletions, and mutations used in the edit distance method. The GA is based on a previous method for deriving bioactivity profiles.[20] Each chromosome of the GA consists of a set of penalty weights defined as an array of integers. The aromatic node types and the aliphatic node types are grouped as shown in Table 1 with all nodes within each group being represented by a single weight. For example, a single weight is defined for the mutation of any aromatic ring node (that is, with either

donor, acceptor, donor and acceptor, or no h-bonding character) to any other aromatic ring node (except itself which is assigned a weight of zero). A total of 47 weights are encoded in a chromosome, shown by the elements in bold in Table 2.

There are nine weights for insertions/deletions: one to represent the aromatic node types, one to represent the aliphatic node types; six to represent the individual acyclic node types, and one to represent the double bond. Single bonds are no longer considered explicitly, in contrast to Harper's method. When coded explicitly, the only valid operation that can be applied to a single bond is to mutate it to a double bond (this effectively changes two rings joined by a single bond to a fused ring system and therefore is only valid when the connecting nodes are ring nodes). If single bonds are made implicit and removed from the paths, the mutation of two singly bonded rings to a fused ring system can be effected by insertion of a double bond into the path; conversely, the mutation of a fused ring system to two nonfused rings (i.e., a double bond to a single bond) can be represented by the deletion of a double bond. The implicit representation of single bonds simplifies the edit distance calculation by reducing the lengths of the paths considerably and also by reducing the number of weights that are considered by the GA. The nine nodes types give rise to 36 unique node pairs ($n(n-1)/2$ possible pairs, where $n = 9$). Two additional weights arise for mutation of one aromatic node type for another (i.e., within group), and for mutation of one aliphatic node type for another (i.e., within group). Note that the insertion and deletion operations applied to a given node are assigned the same weight and that mutations are treated symmetrically (i.e., mutation of node type *X* to *Y* is assigned the same weight as mutation of *Y* to *X*).

The fitness function of the GA is based on the recall of actives in similarity searches carried out in the MDL Drug Data Report (MDDR) database[21] where similarity is calculated using the reduced graph edit distance similarity measure and the weights held in a chromosome. The similarity searches are carried out for multiple query structures, and the fitness of the chromosome is taken as the mean recall over all queries. The GA attempts to find the set of weights that gives maximum mean recall.

The fitness function requires repeated application of similarity searches with the same query compounds over the same database compounds. The efficiency of this process is increased significantly through pre-processing of the database. Each edit distance calculation between a given pair of compounds requires that the paths are enumerated for both the query compound and the database compound. The data set is pre-processed so that the paths are enumerated once only (rather than each time the fitness function is called) and are held in memory. The total number of path comparisons required for calculating the similarity between one query compound and all compounds in the data set is $m \times N$ where m is the number of maximal paths in the query and N is the total number of maximal paths in the data set compounds. However, inspection of the paths generated across subsets of the MDDR revealed that many of the paths are duplicated, either within a single reduced graph or, more commonly, across different reduced graphs. Figure 3 shows a plot of the total number of paths generated for compounds selected at random from the MDDR as compared to the number of unique paths. As can be seen, the total number of paths increases much more rapidly than the number of unique paths, which appears to be reaching a plateau. Further substantial efficiencies are gained by storing the paths for the entire data set as an inverted index. Here each unique path in the data set is stored once only together with a list of the reduced graphs in which it occurs. The number of path comparisons required to calculate the edit distance similarities is now $m \times n$, where n is the number of unique paths in the data set.

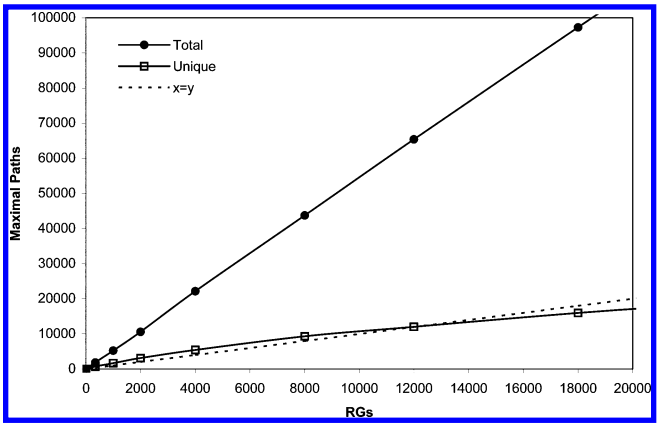

Figure 3. Maximal reduced graph paths in MDDR. The total number of maximal paths generated from reduced graphs that represent structures extracted from the MDDR is given together with the number that are unique. The total number of paths increases much more rapidly than the number of unique paths that appears to be reaching a plateau.

Parametrizing the GA. Preliminary experiments were carried out on a set of 5HT-reuptake inhibitors (hereon referred to as the 5HT-RT data set) extracted from the MDDR in order to determine various parameters of the GA including the optimum weight range (referred to as the weight granularity) and appropriate data set sizes.

A subset of 200 actives was selected at random from the 5HT-RT data set and mixed with 3800 molecules randomly selected from the remainder of the MDDR excluding the activity class. The GA was run using varying numbers of the actives as queries, and the average recall (percentage of the 200 most similar molecules correctly predicted as active in the top 200 positions in the ranked list) over all the queries was calculated. In each case, the results are taken over three different GA runs, each using a different randomly selected set of queries. Since Harper's weights are limited to integer values 1, 2, and 3, the GA was restricted to selecting weights in the same range.

Table 3 shows the mean recalls for the GA optimized weights, Harper's weights, and randomly assigned weights. The number of generations and time to convergence for the GA are also reported. The results show that the GA-evolved weights give significantly better recalls than Harper's weights, which in turn are better than the randomly assigned weights. Note that when the weights are assigned at random, the edit distance is still related to the number of operations required to convert one reduced graph to another—it is the relative costs associated with the different operations that are assigned at random. As the number of queries used increases, in general, the mean recall also increases (although recall is reduced for 100 queries); however, the run time also increases as expected. It was decided to use approximately

Table 3. Performance of the GA on 5HT-RTs[a]

		GA-derived wt				Harper's wt		random wt	
no. of queries	fraction of acty	mean recall	SD	time (min)	convergence (generations)	mean recall	SD	mean recall	SD
10	1/20	18.7	1.5	45	560	15.3	1.0	13.6	1.9
20	1/10	19.6	2.6	80	572	16.5	2.8	15.1	2.1
50	1/4	19.9	1.6	200	562	17.5	1.2	15.8	1.1
100	1/2	18.9	0.2	380	556	16.6	0.2	15.0	0.4

[a] Mean recalls and standard deviations calculated over three independently and randomly selected sets of 5HT-RT inhibitors as queries using GA optimized weights, Harper's weights, and randomly assigned weights. The numbers of generations to convergence and the processing times of the GA are also reported. Times are reported for running on a standard Linux PC (2.8GHz Pentium 4 running Red Hat Enterprise 3). The code is written in the C programming language.

Table 4. GA Parameters

parameter	value	parameter	value
population size	100	mutation chance (%)	50
max generations	1000	mutation rate (%)	10
replacement (%)	10	one-point crossover	75
convergence	200	two-point crossover	50

one-sixth of the activities as queries in the subsequent experiments as this represents a good compromise between run time, quality, and consistency in the results.

GA Parameters. The GA parameters used throughout the experiments are shown in Table 4.

Protocol. The 342 5HT-RT inhibitors present in the MDDR (following removal of structures that could not be processed by the edit distance method due to the presence of cycles in the reduced graphs) were divided at random into three subsets of 114 activities (*A*, *B*, and *C*) to provide training, test, and validation sets, respectively. Each subset of actives was then mixed with a set of 2000 inactives randomly selected from the MDDR after the removal of the 5HT-RTs. A subset of 19 actives was selected at random from the training, test, and validation sets for use as queries. The GA was run on the training set with the aim of maximizing mean recall (that is, the percentage of actives occurring in the top 114 positions in the ranked list) averaged over the 19 queries. Once the GA had reached convergence, the best chromosomes produced over all generations were scored using the test set, and the set of weights that gives maximum mean recall on the test set was selected. The mean recalls that are reported in all subsequent tables are then based on applying these weights to the validation sets. This process was repeated using three different randomly selected sets of queries. The test and validation sets were then exchanged and the runs repeated to give a total of six GA runs. Finally, the entire process was repeated with *B* and *C* as the training sets, respectively, giving a total of 18 runs.

Table 5. Mean and Standard Deviation of the Percentage Recall Values Are Reported for 18 Validation Runs Using the 5HT-RTs[a]

weight range	1−3	1−4	1−6	1−9
mean recall	16.9	17.1	18.4	18.8
SD	3.7	3.8	3.7	3.8

[a] The penalty weight ranges indicate the integer values that are used during training of the GA.

The protocol was based on training, test, and validation sets in an attempt to prevent overtraining of the weights. Figure 4 shows the typical performance of the GA with the black line showing the best recall as determined using the training set, whereas the gray line shows best performance based on the test set. As expected, performance on the test set is slightly worse than on the training set, however, the reduction in performance is small (close to 1%). Similar plots were found for the data sets described later.

Determining the optimum granularity. Next, the optimum granularity of the weights was determined. The ranges of penalty weights investigated were: 1−3; 1−4; 1−6; and 1−9. For each weight range, the edit distance measure was normalized by the mean of the weights over all operations (in Harper's method the edit distance is normalized by the value 2 since most of the weights are 2). Mean percentage recalls are shown in Table 5 where it can be seen that the

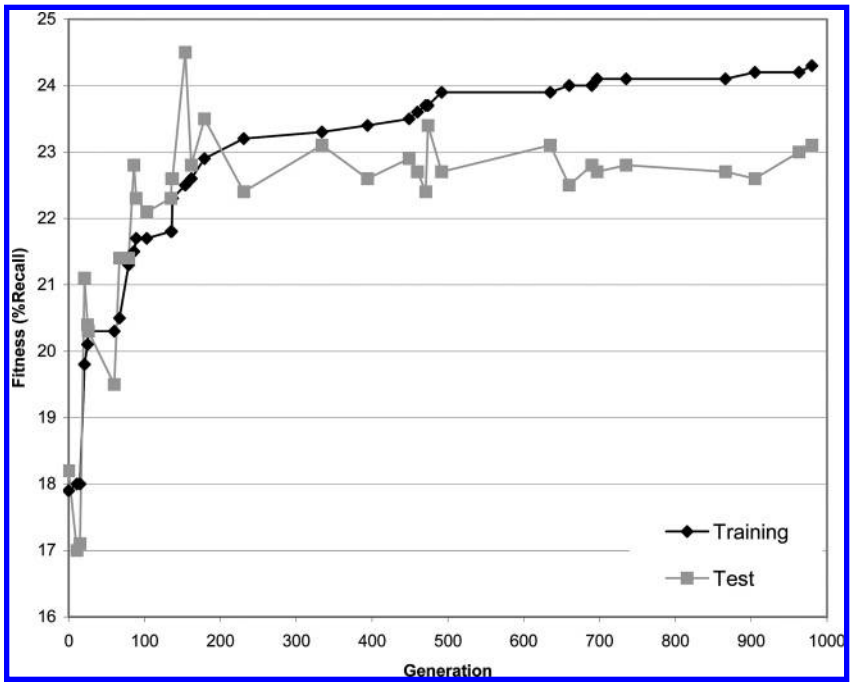

Figure 4. Progress of the GA. The black line shows the fitness of the training set, and the gray line shows the fitness of the test set.

Table 6. Activity-Class Specific Results[a]

	GA		Harper's		random	
data set	mean	SD	mean	SD	mean	SD
5HT1A	34.4	2.3	23.7	1.9	20.9	2.2
5HT-RT	18.4	3.6	15.4	3.1	13.4	3.2
Renin	81.2	1.5	73.6	1.9	70.2	4.2
COX	18.6	1.8	13.6	1.3	14.6	1.5

[a] Mean and standard deviations for recall values calculated over the 18 validation runs carried out after training the GA on different activity classes. Recalls are also reported for applying Harper's and randomly assigned weights as described in the text. The weights are activity specific (i.e., the GA is trained on each class independently).

mean recall increases with increasing range. The 1 −3 range, which is the range used by Harper, gives the worst performance. The weight range of 1−6 was chosen for the subsequent experiments due to its relatively high recall and relatively low standard deviation.

Data Sets. Four of the activity classes described by Hert et al.[22,23] were selected from MDDR. These are 711 5HT1 agonists (5HT1A); the 342 5HT reuptake inhibitors (5HT-RT) described earlier; 1015 renin inhibitors (Renin); and 620 cyclooxygenase inhibitors (COX) (following the removal of structures that could not be processed by the edit distance method). Each data set was divided into training, test, and validation sets and runs were carried out using the protocol identified above with the values reported being based on the validation sets. We acknowledge the limitations of the MDDR for this kind of study, namely, the occurrence of close analogues which favors 2D methods, and the lack of explicit information about the inactivity of compounds such that inactives identified as false positives could in fact be true positives. However, the data set has the significant advantages of being publicly available and of being widely used in the literature and thus allows the relative performance of different similarity methods to be compared. The four activity classes were selected to represent different degrees of homogeneity within the actives measured by the mean intraset pairwise similarity (MPS) calculated using UNITY fingerprints and the Tanimoto coefficient, as reported by Hert et al. According to this measure, the 5HT1As (MPS = 0.343) are of similar homogeneity to the 5HT-RTs (MPS = 0.345) used to parametrize the GA; the Renins (MPS = 0.573) represent a structurally homogeneous set; and the COX inhibitors (MPS = 0.268) represent a class with low structural similarity. The rationale for choosing these particular data sets is as follows. Since the 5HT1As are similar in diversity to the 5HT-RTs and both are CNS targets, we were interested to see if the performance of the GA was similar. The other two activity classes were chosen as they represent extremes: the Renins are structurally homogeneous and therefore represent an easy set for similarity methods; in contrast, the COX inhibitors are structurally diverse and therefore represent a difficult case.

RESULTS

Deriving Activity Specific Weights. Table 6 shows the means and standard deviations for recalls obtained after training the GA on each of the four activity classes, where in each case the results are averaged over 18 runs. For comparison, recalls obtained using Harper's weights and sets of randomly generated weights (1−6 integer values) are also shown, averaged over the same three random sets of queries selected from subsets *A*, *B*, and *C* used in training by the GA. Considerable variation in the recall values is seen across the different activity classes; however, for all classes the GA-derived weights outperform both Harper's and the randomly generated weights. The percentage improvement ranges from 12% for the Renins to 51% for the 5HT1As. The degree of improvement reflects the absolute recall values seen. For example, the homogeneous nature of the Renins is such that many methods are able to achieve very high recalls of 70% or higher, and it is hard to improve on already excellent performance. The COX inhibitors show the worst recall of the four activity classes when using Harper's weights. This is perhaps not surprising due to the high heterogeneity of the structures, however, considerable improvement is seen using the GA-optimized weights so that the performance is now comparable to that of the 5HT-RTs. Harper's weights outperform the randomly generated weights in three out of the four classes; however, they are slightly worse for the COX data set.

Each run of the GA generates a different set of weights so that a total of 18 sets of weights are evolved for each activity class. The mean weights for selected operations are shown in Figure 5 across all four activity classes where considerable variation across the different activity classes can be seen. For example, the mean weight for substitution between different aromatic node types for the 5HT-RTs, 5HT1As, and COX data sets is around one, whereas, for the Renins the mean weight is >2. The standard deviations of weights within a class (not shown) also vary considerably (from 0 to 2.2). The variation in weights over different runs within the same activity class may be due to differences in the queries used in each run or to the low occurrence of a particular node or combination of nodes within the activity class.

Figure 6 shows the frequency of occurrence of the various node types within the activity classes and the inactive compounds and can be used to help interpret the weights seen. It can be seen that the frequency of occurrence of different node types varies considerably between the activity classes. For example, the COX inhibitors have a high incidence of aromatic nodes (taking all the aromatics subtypes together) relative to the other activity classes and the inactive compounds, with 44% of all nodes being aromatic; and the Renins have high occurrence of acyclic donor and acceptor nodes and linker nodes relative to the other classes.

An operation assigned a low mean weight suggests that it does not have a significant impact on similarity. For example, the COX inhibitors have a low weight (equal to 1) assigned to mutation between any of the aromatic node types. The frequency plot shows that aromatic nodes are by far the most prevalent node types in these compounds; therefore, it seems reasonable that mutation between the different aromatic subtypes has less impact on activity than does an operation that changes an aromatic node to a nonaromatic node or that removes it.

On the other hand, an operation assigned a high weight suggests that it has a high impact on activity. For example, all mutation operations involving an acyclic negative node in the Renins are assigned high weights. The Renins have a

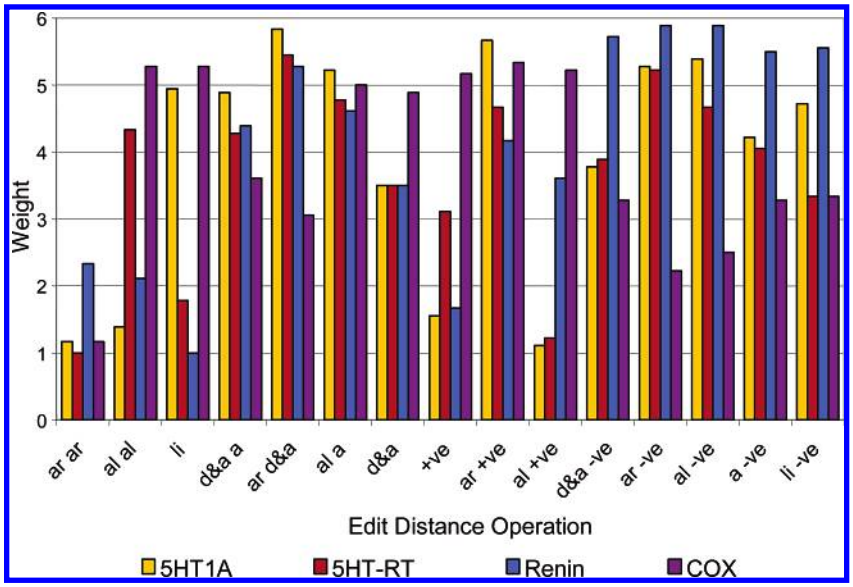

Figure 5. Activity-class specific weights. The mean values of selected weights over the 18 runs are shown for each of the data sets. The *x*-axis indicates the operation, a weight labeled by two node types represents mutation between the node types; a weight represented by a single node type indicates insertion/deletion of the node.

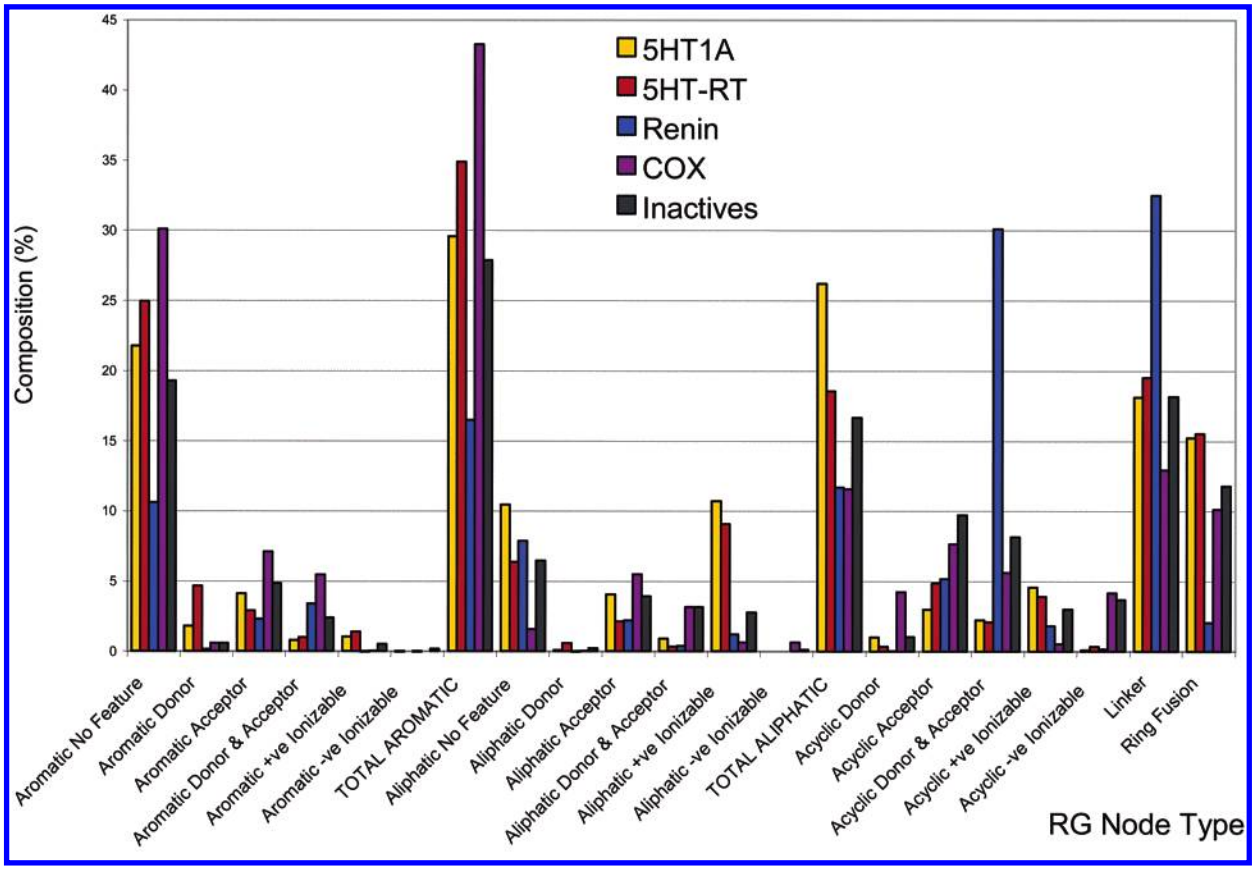

Figure 6. Frequency of occurrence of node types.

very low incidence of acyclic negative nodes relative to the inactive compounds, and the introduction of such a node would suggest loss of activity.

In some cases, the standard deviations are small, indicating little or no variation across the runs. These operations can therefore be considered more significant than operations where the weights vary considerably from one run to another. For example, the insertion/deletion of a linker node in the Renins, the mutation between different aromatic node types in the 5HT1As, and the mutation between aromatic and aliphatic node types in the 5HT-RTs all have standard deviations of 0 and are all assigned the lowest weight of 1, indicating that these operations have a low impact on activity. The low weights indicate that these modifications can be made to the reduced graphs without loss of activity.

While some general trends in the weights have been observed it is not possible to interpret them fully due to a number of factors. For example, the significance that can be attributed to the weights is dependent on their frequency of occurrence in the data sets. Furthermore, there are usually

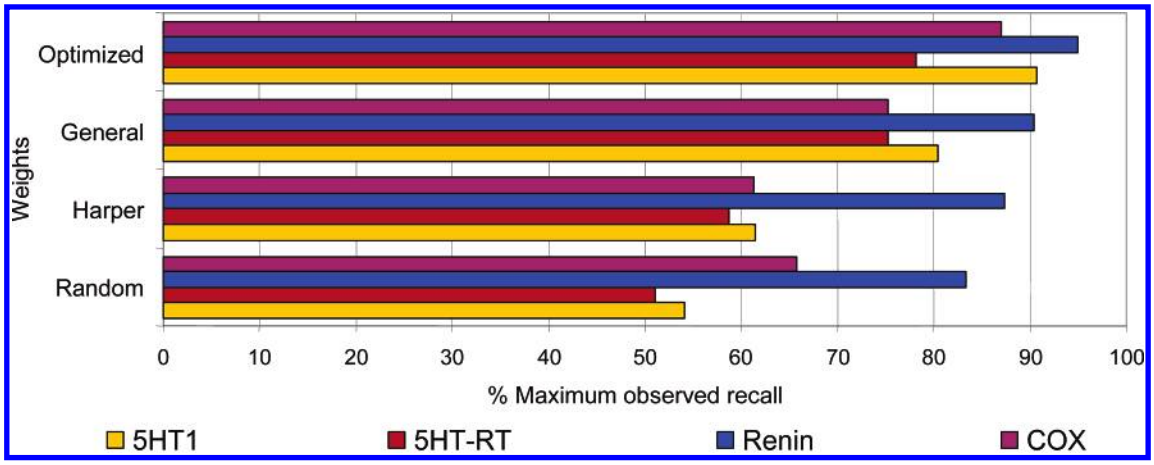

Figure 7. Peformance when training on multiple activity classes. Improvements in recall seen when the weights are optimized across all classes simultaneously (optimized) relative to the general weights, Harper's weights, and randomly assigned weights. The recalls are shown as a percentage of the maximum recall seen in each class when optimizing on the classes independently.

Table 7. Results of Training the GA on Multiple Activity Classes[a]

	5HT1A	5HT-RT	Renin	COX
upper bound	38.6	26.2	84.3	22.2
optimized	35.0 (2.2)	20.5 (4.3)	80.1 (1.5)	19.3 (1.9)
general	31.0 (1.6)	19.7 (4.3)	76.2 (2.1)	16.7 (1.3)
Harper	23.7 (1.9)	15.4 (3.1)	73.6 (1.9)	13.6 (1.3)
random	20.9 (2.2)	13.4 (3.2)	70.2 (4.2)	14.6 (1.5)

[a] Recall values and standard deviations (where applicable) are shown for weights trained over all activity classes simultaneously (optimized). The upper bound represents the maximum recall achieved for a given activity class when optimizing the weights on each class independently; hence, a different set of weights is applied to each of the activity classes. In all cases except upper bound, the results are reported for a single set of weights that is applied across all of the activity classes.

several operations involved when two reduced graphs are compared so that the operations are not independent. Finally, the weights do not indicate the direction in which an operation is applied so that both the insertion and deletion of a given node are assigned the same weight and mutation between two nodes is treated symmetrically. While theoretically it would be possible to impose direction on the operations this would increase the number of variables in the GA considerably with a consequent increase in the size of the search space and was not considered to be computationally feasible.

Deriving Generalized Weights. The previous experiments demonstrate that improved performance in the edit distance similarity method can be achieved when the weights are trained on a particular activity class; however, it is expected that the resulting weights are activity-class specific. This was investigated below.

The sets of weights represented in the final populations of all of the GA runs were pooled across all of the activity classes. Each set of weights was then used in similarity searches across all of the activity classes (not just the class that the weights were trained on). To select the set of weights that performs best over all activity classes, the recalls were first normalized. Normalization is required to account for the variation in recall values observed between the activity classes, for example, the Renins typically give recalls in excess of 70% whereas the COX inhibitors give recalls of less than 20%. For each activity class, the set of weights in the pooled set that gave maximum recall for the class was found, and the recall achieved was recorded. This was considered to be an upper bound for the activity class. Note that the upper bound for each activity class is achieved using a different set of weights in each case. To find the best generalizing set of weights, that is, a single set of weights that performs well across all activity classes, each recall value was then reported as the percentage of the upper bound for that class. The best set of weights was identified using a max−min criterion applied across all classes. The max−min criterion identifies the set of weights that gives maximum improvement in recall in the class with worst performance overall. This criterion was chosen to ensure a reasonable level of performance across all activity classes. This set of weights is referred to as general weights. The mean recall for each activity class using the general weights is reported in Table 7, along with the upper bound for each activity class.

The GA was then trained on all four activity classes simultaneously. Thus the data set consisted of training, test, and validation sets for all four activity classes. A chromosome was scored by first calculating the recall in each activity class using the weights stored in the chromosome. The recalls were then normalized using the upper bounds identified previously, and the max−min criterion was applied as described earlier, so that the fitness was given as the maximum normalized recall in the worst performing activity class.

Results for the best set of weights optimized over all activity classes are presented in Table 7 (optimized). The best generalized weights found when optimizing on the activity classes independently (general) are shown for comparison, as are the upper bounds on recall for each activity class. In the latter case, the weights are activity-class specific, with each activity class being represented by a different set of weights. Figure 7 shows the same results graphically as percentages of the upper bounds on recall seen in each activity class. It can be seen that optimizing the weights using the GA represents a significant improvement (up to 12%) on the best generalizing weights chosen from the independent optimization runs. The optimized weights also represent a significant improvement on Harper's weights. Thus we have been able to derive a set of weights that gives substantial improvements over all of the activity classes as compared to the original weights devised by Harper.

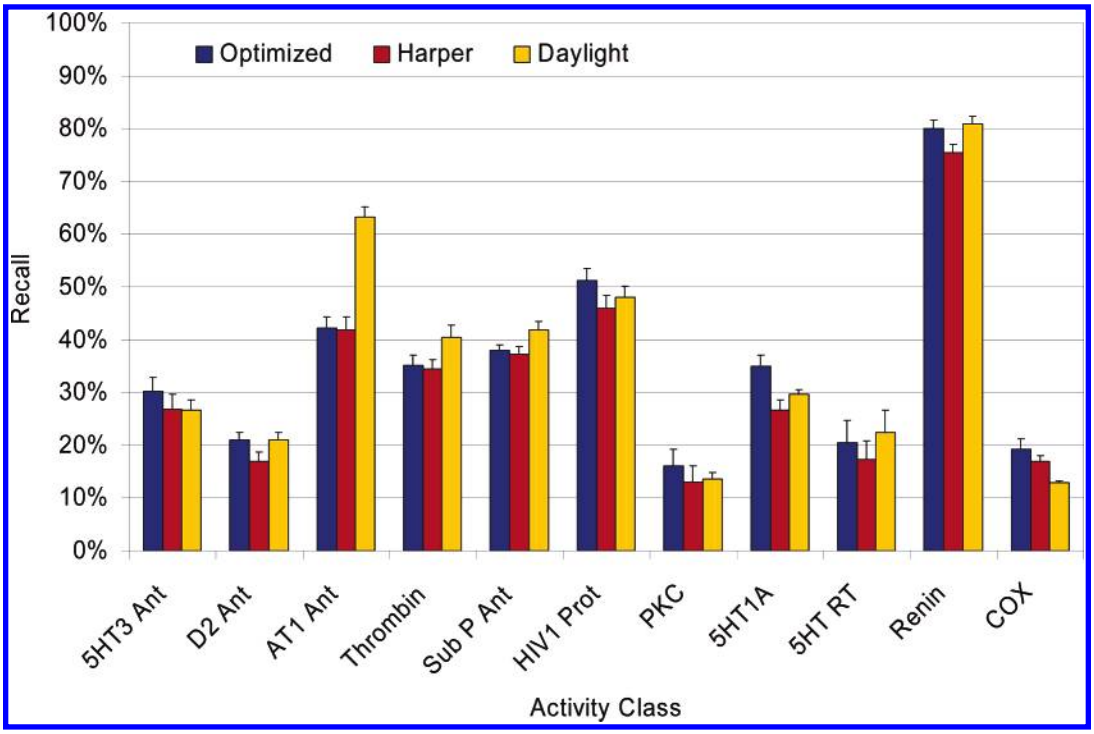

Figure 8. Performance of optimized weights on previously unseen activity classes. Recalls obtained using the optimized weights on all activity classes in Hert's data set are shown together with recalls obtained for Harper's weights and when using a more conventional similarity method, namely, Daylight fingerprints.

Table 8. Weights Derived by Optimizing on Multiple Activity Classes[a]

operation	5HT1A	5HT-RT	Renin	COX	general	optimized	Harper
ar ar	1.2	1.0	2.3	1.2	1	1	2
al al	1.4	4.3	2.1	5.3	3	3	4
li	4.9	1.8	1.0	5.3	2	2	2
d&a a	4.9	4.3	4.4	3.6	4	6	2
al ac	5.2	4.8	4.6	5	6	6	4
+ve	1.6	3.1	1.7	5.2	5	6	4
al +ve	1.1	1.2	3.6	5.2	1	1	4
ar −ve	5.3	5.2	5.9	2.2	6	6	4
RF	1.7	2.2	1.4	4.1	1	1	6

[a] A single node in the first column indicates an insertion/deletion operation, whereas two node types indicate the mutation operator. RF refers to the insertion/deletion of a double bond (a double bond represents a ring fusion). The columns labeled by activity classes indicate weights derived when the GA is trained on each class independently. The remaining columns indicate weights chosen based on performance over all of the activity classes.

Moreover, the optimized weights give performance that is close to the upper bounds seen when optimizing on each class independently with the performance ranging from 78 to 95% of the upper bound, depending on the activity class.

Weights for a subset of the operations are shown in Table 8. Harper's weights (in the range 1−3) have been multiplied by two to allow comparison with the GA-derived weights (in the range 1−6). The weights for the two GA-derived sets (general and optimized) are fairly similar, and in fact 16 weights are identical with a further 17 weights differing by a value of 1 only. Some of Harper's weights, which were based on intuition, agree with the weights evolved by the GA; however, in other cases there are significant differences. For example, insertion/deletion of a double bond (equivalent to fusing two rings) was given a high weight by Harper; however, the GA suggests that a low weight should be attributed to this operation. Similarly, the mutation between an aliphatic ring node and an acyclic positively charged node is given a high weight by Harper; whereas, the GA has evolved a low weight.

Table 9. Additional Activity Classes in Hert Data Set Used To Validate the Optimized Weights

activity class	no. of activities
5HT3 antagonists (5HT3 Ant)	752
dopamine D2 antagonists (D2 Ant)	827
angiotensin II AT1 antagonists (AT1 Ant)	943
thrombin inhibitors (Thrombin)	797
substance P antagonists (Sub P Ant)	1246
HIV 1 protease inhibitors (HIV 1 Prot)	750
protein kinase C inhibitors (PKC)	453

While it is clear that training the GA against multiple activity classes simultaneously has resulted in a more effective set of weights, the extent to which they are generalizable to activity classes not included in the training is not yet known. Therefore, a final set of experiments was performed in which the optimized weights were applied to previously unseen activity classes to form an independent validation of the weights. The additional data sets are the remaining seven activity classes in the Hert data set shown in Table 9 (following removal of those that could not be processed). Three random subsets were selected from each activity class and mixed with the 2000 inactives. Approximately one-sixth of the actives in each subset were selected at random as queries for similarity searches, as before. The selection of actives as queries was repeated three times for each of the three random subsets. The average recall values found using the optimized weights are compared with those obtained using Harper's weights in Figure 8. The results for the data sets used in training the weights are also shown for ease of comparison. It can be seen that the optimized weights result in improved recall for all but one of the data sets (AT1 Ant), demonstrating that they are also

effective for activity classes that were not included in training. It remains to be seen whether it is possible to derive a truly global set of weights that can be applied across all biological activity classes. Finally, the recalls obtained using Daylight fingerprints are also shown for all data sets to allow comparison of the reduced graph method with a more conventional searching method. With the exception of the AT1 Ant data set, the overall performance of the reduced graphs is comparable to that of Daylight fingerprints which is consistent with previous studies using reduced graphs.[9,10]

CONCLUSIONS

A similarity measure based on the use of edit distances was previously shown to provide an effective way of comparing reduced graphs. The method overcomes some of the limitations inherent in the representation such as the insertion problem whereby small changes in chemical structure can lead to significant changes in reduced graph representation due to the insertion or deletion of nodes. It also allows relationships between different node types to be preserved by assigning different weights to different node mutations. In the original method, weights for the different node operations applied in the edit distance measure were assigned using intuition. Here we have developed a GA that optimizes the weights based on a training set of known actives and (presumed) inactives. Significant improvements in performance were seen when the GA was trained on four different activity classes. The result, in each case, is a set of weights specific to the activity class. While these activity-specific weights may provide useful structure−activity information, in a virtual screening setting when few active compounds are known it is more useful to have weights that perform well across a variety of different activity classes. By training the GA on all the activity classes simultaneously, we were able to derive a single set of weights that gives substantial improvements over Harper's original weights across the activity classes. Moreover, the generalized weights give performance that is close to the upper bounds seen when optimizing on each class independently.

ACKNOWLEDGMENT

The authors acknowledge Eleanor Gardiner for helpful comments on this manuscript, Daylight Chemical Information for software support, and MDL Information Systems Inc. for the provision of the MDDR database. The work was funded by GlaxoSmithKline and BBSRC via an industrial CASE studentship.

REFERENCES AND NOTES

(1) Bender, A.; Glen, R. C. Molecular similarity: a key technique in molecular informatics. *Org. Biomol. Chem.* **2004**, *2*, 3204−3218.

(2) Nikolova, N.; Jaworska, J. Approaches to measure chemical similarity—a review. *QSAR Comb. Sci.* **2003**, *22*, 1006−1026.

(3) Willett, P. The evaluation of molecular similarity and molecular diversity methods using biological activity data. *Methods Mol. Biol.* **2004**, *275*, 51−63.

(4) Lajiness, M. S. Molecular similarity-based methods for selecting compounds for screening. In *Computational Chemical Graph Theory*; Rouvray, D. H., Ed.; Nova Science Publishers: New York, 1990; pp 299−316.

(5) Willett, P. *Similarity and Clustering in Chemical Information Systems*; Research Studies Press: Letchworth, 1987.

(6) Johnson, M. A.; Maggiora, G. M. *Concepts and Applications of Molecular Similarity*; John Wiley: New York, 1990.

(7) Willett, P.; Barnard, J. M.; Downs, G. M. Chemical similarity searching. *J. Chem. Inf. Comput. Sci.* **1998**, *38* (6), 983−996.

(8) Sheridan, R. P.; Kearsley, S. K. Why do we need so many chemical similarity search methods? *Drug Discovery Today* **2002**, *7* (17), 903−911.

(9) Gillet, V. J.; Willett, P.; Bradshaw, J. Similarity searching using reduced graphs. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (2), 338−345.

(10) Barker, E. J.; Gardiner, E. J.; Gillet, V. J.; Kitts, P.; Morris, J. Further development of reduced graphs for identifying bioactive compounds. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (2), 346−356.

(11) Harper, G.; Bravi, G. S.; Pickett, S. D.; Hussain, J.; Green, D. V. S. The reduced graph descriptor in virtual screening and data- driven clustering of high-throughput screening data. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (6), 2145−2156.

(12) Schneider, G.; Neidhart, W.; Giller, T.; Schmid, G. "Scaffold-hopping" by topological pharmacophore search: a contribution to virtual screening. *Angew. Chem., Int. Ed. Engl.* **1999**, *38* (19), 2894−2896.

(13) Schuffenhauer, A.; Floersheim, P.; Acklin, P.; Jacoby, E. Similarity metrics for ligands reflecting the similarity of the target proteins. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (2), 391−405.

(14) Rarey, M.; Dixon, J. S. Feature trees: A new molecular similarity measure based on tree matching. *J. Comput.-Aided Mol. Des.* **1998**, *12* (5), 471−490.

(15) Kearsley, S. K.; Sallamack, S.; Fluder, E. M.; Andose, J. D.; Mosley, R. T.; Sheridan, R. P. Chemical similarity using physiochemical property descriptors. *J. Chem. Inf. Comput. Sci.* **1996**, *36* (1), 118−127.

(16) Böhm, H.-J.; Flohr, A.; Stahl, M. Scaffold hopping. *Drug Discovery Today: Technol.* **2004**, *1* (3), 217−224.

(17) Takahashi, Y.; Sukekawa, M.; Sasaki, S. Automatic identification of molecular similarity using reduced-graph representation of chemical-structure. *J. Chem. Inf. Comput. Sci.* **1992**, *32* (6), 639−643.

(18) Barker, E. J.; Cosgrove, D. A.; Gardiner, E. J.; Gillet, V. J.; Kitts, P.; Willett, P. Scaffold-hopping using clique detection applied to reduced graphs. *J. Chem. Inf. Model.* **2006**, *46*, 503−511.

(19) Gusfield, D. *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*; Cambridge University Press: Cambridge, 1997.

(20) Gillet, V. J.; Willett, P.; Bradshaw, J. Identification of biological activity profiles using substructural analysis and genetic algorithms. *J. Chem. Inf. Comput. Sci.* **1998**, *38* (2), 165−179.

(21) MDL. MDL Information Systems, Inc. 14600 Catalina Street, San Leandro, CA. http://www.mdli.com.

(22) Hert, J.; Willett, P.; Wilton, D. J. Comparison of fingerprint-based methods for virtual screening using multiple bioactive reference structures. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (3), 1177−1185.

(23) Hert, J.; Willett, P.; Wilton, D. J.; Acklin, P.; Azzaoui, K.; Jacoby, E.; Schuffenhauer, A. Comparison of topological descriptors for similarity-based virtual screening using multiple bioactive reference structures. *Org. Biomol. Chem.* **2004**, *2* (22), 3256−3266.

CI050465E