Kernel Approach to Molecular Similarity Based on Iterative Graph Similarity

Matthias Rupp,* Ewgenij Proschak, and Gisbert Schneider

Beilstein Endowed Chair for Cheminformatics, Johann Wolfgang Goethe-University, Siesmayerstrasse 70,
60323 Frankfurt am Main, Germany

Received July 26, 2007

Similarity measures for molecules are of basic importance in chemical, biological, and pharmaceutical applications. We introduce a molecular similarity measure defined directly on the annotated molecular graph, based on iterative graph similarity and optimal assignments. We give an iterative algorithm for the computation of the proposed molecular similarity measure, prove its convergence and the uniqueness of the solution, and provide an upper bound on the required number of iterations necessary to achieve a desired precision. Empirical evidence for the positive semidefiniteness of certain parametrizations of our function is presented. We evaluated our molecular similarity measure by using it as a kernel in support vector machine classification and regression applied to several pharmaceutical and toxicological data sets, with encouraging results.

1. INTRODUCTION

Similarity measures for molecules are of basic importance for many computer-based applications in the chemical, biological, and pharmaceutical sciences. Popular applications include the enrichment of bioactive compounds (creation of focused libraries), the selection of promising drug candidates (virtual screening) and the construction of compounds similar to reference compounds (de novo design), the derivation of quantitative structure−activity relationships (QSAR), and the selection of maximally diverse subsets. A common objective is to predict biochemical activity (e.g., toxicity, bioavailability, inhibition) from molecular structure. As comprehensive information about the fundamental chemical and biological processes underlying a specific activity is often unavailable, inferences are instead based on the notion that similar molecules tend to have similar properties.[1]

To measure the similarity of molecules, one needs a representation (e.g., topology, shape, physicochemical descriptors) of the molecules to be compared as well as a method (e.g., metrics, kernels, similarity coefficients) to compare two such representations.[2] Results depend on both method and representation. The abundance of available representations—the handbook of molecular descriptors[3] lists more than 1600 of them—is caused by the necessity to selectively model molecular properties relevant to the specific activity under investigation. [Theoretically, a molecule is completely described by its quantum mechanical wave function. In practice, however, the Schrödinger equation can be solved analytically only for the hydrogen atom, and approximate (ab initio) solutions are presently limited to molecules with no more than a few dozen electrons for computational reasons.] The selection of a good molecular model for a specific task is a problem in itself.

Many molecular models focus on real-valued vector representations. Consequently, popular choices for molecular similarity measures include metrics (e.g., p-norm induced metrics, spherical distance), similarity coefficients (e.g., Tanimoto coefficient, cosine coefficient), and kernels (e.g., polynomial kernel, Gaussian kernel) defined on vector spaces. [Here, a kernel $\kappa : X \times X \rightarrow \mathbb{R}$ is a function corresponding to an inner product in some Hilbert space H, i.e. $\kappa(x,y) = \langle\phi(x),\phi(y)\rangle$, where $\phi: X \rightarrow H$. For continuous κ or finite X, this is equivalent to κ being symmetric and all Gram matrices $(\kappa(x,y))_{x,y\in A}$ for finite $A \subseteq X$ being positive semidefinite.[4]] On the other hand, the molecular graph is an established and intuitive representation of molecules. Consequently, many similarity measures rely on graph features (e.g. subgraph counts, subgraph matching, interatomic edge distance distributions). Often, these are not directly compared; instead, they are first converted into vectors (using, e.g., fingerprints, hashing, binning) and then compared using vector-based methods. Such a conversion is not always chemically motivated, and information may be lost or noise may be added in the process.

An alternative is to directly compare the graphs, avoiding the conversion into vectors. Several approaches from graph theory have been adopted for the comparison of molecules. Recently, kernels have been developed which work directly on the molecular graph.[5−14] Since the computation of inner products in the subgraph feature space, which would allow kernels that separate all nonisomorphic graphs, is NP-hard,[16] these methods trade in separation capability for computational efficiency. [Indeed, Ramon and Gärtner[15] have shown that even approximating a complete graph kernel is as hard as the graph isomorphism problem. In general, one cannot expect to use graph kernels to efficiently learn concepts which are hard to compute.]

We introduce a novel similarity measure on molecular graphs, based on the ideas of iterative graph similarity and optimal assignments, in section 2. We give experimental evidence in the form of a retrospective study in section 3 and conclude in section 4.

2. METHOD

2.1. Optimal Assignments. Our approach is based on the idea of an optimal assignment: A similarity measure between

* Corresponding author e-mail: matthias.rupp@bio.uni-frankfurt.de.

10.1021/ci700274r CCC: $37.00
Published on Web 11/07/2007

the vertices of two graphs is defined, and each vertex of the smaller graph is assigned to a vertex of the larger graph such that the total similarity between the assigned vertices is maximized. Formally, let $G = (V,E)$, $G' = (V', E')$ with $V = \{v_1,..., v_{|V|}\}$, and $V' = \{v'_1,...,v'_{|V'|}\}$ denote the labeled molecular graphs of the compared molecules, let $l(v)$ and $l(\{v,v'\})$ denote the labels of a vertex v and an edge $\{v,v'\}$, and let k_x be a similarity measure defined on $V \times V'$. We then call

$$k_a(G,G') = \begin{cases} \max\limits_{\pi} \sum\limits_{i=1}^{|V|} k_x(v_i,v'_{\pi(i)}) \text{ if } |V| \leq |V'| \\ \max\limits_{\pi} \sum\limits_{i=1}^{|V'|} k_x(v_{\pi(i)},v'_i) \text{ otherwise} \end{cases} \tag{1}$$

the optimal assignment between G and G'. The maximum in the first case of the equation is taken over all possible assignments of the vertices in V to vertices in V', i.e. all prefixes of length $|V|$ of permutations of V'; in the second case, the roles of $|V|$ and $|V'|$ are exchanged. Since $k_a(G,G') = k_a(G',G)$, we assume from now on without loss of generality that $|V| < |V'|$. Equation 1 then becomes

$$k_a(G,G') = \max_{\pi} \sum_{i=1}^{|V|} k_x(v_i,v'_{\pi(i)})$$

In the context of machine learning and cheminformatics, this approach was proposed by Fröhlich et al.[11,17] as "optimal assignment kernels". However, their proof that k_a is positive semidefinite is wrong (the inequality at the end of their proof is bounded in the wrong direction). [The positive semidefiniteness of a kernel is important for use with support vector machines, as it guarantees that the underlying quadratic optimization problem is convex, thus ensuring the existence of a global minimum.] Furthermore, their proof idea (induction over matrix size based on properties of 2 × 2 matrices) fails as there are 3 × 3 matrices which are not positive semidefinite, although all their 2 × 2 submatrices are.

Algorithmically, k_a can be computed in two steps: First, the matrix of pairwise vertex similarities

$$X = (k_x(v_i,v'_j))_{\substack{i=1,..., |V| \\ j=1,..., |V'|}}$$

is calculated. Then an optimal assignment (one column assigned to each row) is computed using, e.g., the Kuhn-Munkres assignment algorithm[18-20] (also known as the Hungarian algorithm).

Let k_v denote a kernel defined on vertices (which correspond to atoms) and k_e denote a kernel defined on edges (which correspond to bonds). Default choices for k_v and k_e are the Dirac kernel for discrete labels and the radial basis function kernel for continuous labels. Fröhlich et. al[17] start by defining $k_x(v,v')$ as the mean similarity between all neighbors of v_i and v'_j

$$k_x(v_i,v'_j) = k_v(v_i,v'_j) + \frac{1}{|v_i||v'_j|} \sum_{v \in n(v_i)} \sum_{v' \in n(v'_j)} k_v(v,v')k_e(\{v_i,v\}, \{v'_j,v'\})$$

where $n(v_i)$ denotes the set of all neighbors of vertex v_i. They then extend this definition to include all neighbors up to a given topological distance. In another publication, they replace the mean with an optimal assignment of the neighbors.[11]

2.2. Iterative Similarity Measures. We propose a different choice for k_x based on ideas from iterative similarity measures for graphs.[21] There, two vertices are considered similar if their neighbors are similar. This recursive definition naturally leads to iterative computation schemes for the matrix X of pairwise vertex similarities: The computation starts with an initial matrix $X^{(0)}$, which is then iteratively updated until convergence occurs. In each iteration, the similarity value for a vertex pair (v,v') is updated using the similarity values of all pairs of their neighbors $\{(n_i(v),n_j (v'))|i = 1... |v|,j = 1... |v'|\}$, where $n_i(v)$ denotes the ith neighbor of v. Some iteration schemes normalize the matrix X after each iteration.

Different update equations have been proposed, depending on the application. As an example, consider

$$\begin{aligned} \hat{X}^{(n)} &= AX^{(n-1)}A' \\ \Leftrightarrow \hat{X}^{(n)}_{i,j} &= \sum_{\mu=1}^{|v_i|} \sum_{\nu=1}^{|v'_j|} X^{(n-1)}_{n_\mu(v_i),n_\nu(v'_j)} \end{aligned} \tag{2}$$

where A, A' are the adjacency matrices of G, G', i.e. $A_{i,j} = \delta_{\{v_i,v'_j\} \in E}$. Together with the subsequent normalization

$$X^{(n)} = \frac{\hat{X}^{(n)}}{||\hat{X}^{(n)}||_2} \tag{3}$$

this is the update equation given by Zager,[21] adapted for undirected graphs. It is based exclusively on graph topology and does not consider vertex or edge labels. Although it has the advantage of a succinct matrix notation, there are drawbacks.[21]

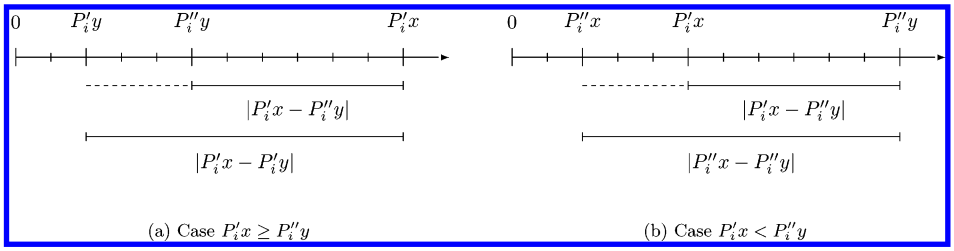
Convergence to a Unique Limit Is Not Guaranteed. Let $A' \otimes A$ denote the Kronecker matrix product of A' and A, let x denote the concatenated columns of X, and let ρ denote the Perron root (the real eigenvalue of largest absolute value) of $A' \otimes A$. Equations 2 and 3 can then equivalently be written as

$$x^{(n)} = \frac{(A' \otimes A)x^{(n-1)}}{||(A' \otimes A)x^{(n-1)}||_2}$$

If $-\rho$ is an eigenvalue of $A' \otimes A$, then the even and odd subsequences of the series $(x^{(n)})_{n \geq 0}$ converge to separate limits.

Initialization Influences the Convergence Limit. If ρ has multiplicity one and $-\rho$ is not an eigenvalue of $A' \otimes A$, the iteration converges to the unique dominant eigenvector of $A' \otimes A$. Otherwise, the initial value $x^{(0)}$ determines the convergence limit.

Specifics of Molecular Graphs Are Not Considered. The equations do not take any labeling of vertices or bonds into account, which is indispensable for chemical similarity measures as the graph topology alone does not provide enough information. Furthermore, molecular graph properties



(a) Case $P'_i x \geq P''_i y$ (b) Case $P'_i x < P''_i y$

Figure 1. The two cases of eq 6. In case (a), replacing $P''_i y$ by $P'_i y$ can only result in a lower value since P''_i maximizes $P''_i y$. Consequently, $|P'_i x - P''_i y| \leq |P'_i x - P'_i y|$. In case (b), the roles of $P'_i x$ and $P''_i y$ are exchanged. In both cases, $|P'_i x - P''_i y| \leq |P_i x - P_i y|$.

(e.g. boundedness of the vertex degree by a small constant) are not exploited. [Theoretically, the maximum vertex degree in a molecular graph could be as high as 6 (12 for metals). However, when we analyzed the data sets used in this study as well as several large vendor catalogues, we found the maximum vertex degree to be 5. What is more, the average vertex degree was consistently (over the different data sets) slightly above 2, which we attribute to the dominance of carbon atoms in ring systems (hydrogens were removed).]

2.3. Iterative Similarity for Molecular Graphs. For our purposes, we propose the following update equation which remedies the above limitations

$$X^{(n)}_{i,j} = (1-\alpha)k_v(v_i, v'_j) + \alpha \max_{\pi} \frac{1}{|v'_j|} \sum_{v \in n(v_i)} X^{(n-1)}_{v,\pi(v)} k_e(\{v_i,v\}, \{v'_j,\pi(v)\}) \quad (4)$$

for $|v_i| < |v'_j|$ and respectively

$$X^{(n)}_{i,j} = (1-\alpha)k_v(v_i, v'_j) + \alpha \max_{\pi} \frac{1}{|v_i|} \sum_{v' \in n(v'_j)} X^{(n-1)}_{\pi(v'),v'} k_e(\{v_i,\pi(v')\}, \{v'_j,v'\})$$

for $|v_i| \geq |v'_j|$. Here, $0 < \alpha < 1$. For each pair of vertices (v_i,v'_j), eq 4 optimally assigns the neighbors of the vertex with smaller degree to neighbors of the vertex with larger degree, based on the similarity values of the previous iteration. The weighting parameter α determines the influence of the constant and the recursive part of the equation. The remarks for eq 1 apply here as well. From now on, we assume that $k_v,k_e \in [0,1]$. Equation 4 obviously takes vertex and edge labels into account. In the following, we introduce a succinct matrix notation, prove that the corresponding iteration converges to a unique limit independent of $X^{(0)}$, and show how eq 4 exploits the bounded degree of molecular graphs.

To obtain the matrix notation, note that the graph neighborhood structure is fixed during the computation, which renders $k_v(v_i,v'_j)$, $|v_i|^{-1}$, $|v'_j|^{-1}$, $n(v_i)$, $n(v'_j)$, and $k_e(\{v_i,v_j\}, \{v'_{i'},v'_{j'}\})$ constants depending only on i, j, i', and j' and predetermines the case in eq 4 for each combination of i and j. Let $x^{(n)}$ denote as before the concatenated columns of $X^{(n)}$ and let

$$k_v = (k_v(v_1,v'_1),k_v(v_2,v'_1), ..., k_v(v_{|V|},v'_1), k_v(v_1,v'_2), ..., k_v(v_{|V|},v'_{|V'|}))$$

denote the corresponding vectorization of k_v (we use the same symbol for the function and the vector). We encode all the neighbor assignments within a single iteration into an $|V||V'| \times |V||V'|$ square matrix P as follows: Each row corresponds to a specific neighbor assignment in eq 4, e.g., the row $(j-1)|V| + i$, which corresponds to entry $X^{(n)}_{i,j}$, contains one possible assignment of neighbors of v_i to neighbors of v'_j or vice versa. The nonzero entries of P are the corresponding products of k_e and $|v'_j|^{-1}$ or $|v_i|^{-1}$, respectively. Equation 4 can then be written as

$$x^{(n)} = (1-\alpha)k_v + \alpha \max_P P x^{(n-1)} \quad (5)$$

where the maximum is over all possible matrices P, i.e. all matrices compliant with the graph neighborhood structure. For the formal determination of the maximum, we compare two vectors a and b using

$$a < b \Leftrightarrow \forall\, i{:}a_i \leq b_i \wedge \exists i{:}a_i < b_i$$

This corresponds to the componentwise determination of the maximum using eq 4.

Theorem 1. *For any $x^{(0)} \geq 0$, the iteration given by eq 5 converges to the unique solution of $x = (1-\alpha)k_v + \alpha \max_P Px$.*

Proof. Let $M = \{x \in \mathbb{R}^{|V||V'|} | x_i \geq 0\}$ denote the non-negative orthant and let $f: M \rightarrow M$, $x \mapsto (1-\alpha)k_v + \alpha \max_P Px$. We show that f is a contraction mapping on M, that is $||f(x) - f(y)|| \leq \lambda ||x - y||$ for some positive $\lambda < 1$ and some norm $||\cdot||$.

Let $P' = \arg\max_P Px$ and $P'' = \arg\max_P Py$, i.e. P' and P'' are the matrices that maximize Px and Py componentwise. Define P by setting

$$P_i = \begin{cases} P'_i & \text{if } P'_i x \geq P''_i y \\ P''_i & \text{if } P'_i x < P''_i y \end{cases} \quad (6)$$

where P_i denotes the ith row of P. Note that $|P'_i x - P''_i y| \leq |P_i x - P_i y|$, as illustrated by Figure 1, giving

$$\begin{aligned} ||f(x) - f(y)||_\infty &= \alpha ||\max_P Px - \max_P Py||_\infty \\ &= \alpha ||P'x - P''y||_\infty \\ &\leq \alpha ||P(x-y)||_\infty \\ &\leq \alpha ||x - y||_\infty. \end{aligned}$$

The last line follows from a property of P: At most $\min\{|v|,|v'|\}$ (the number of assigned neighbors) entries in the ith row of P are not zero, and every such entry contains the factor $1/\max\{|v|,|v'|\}$, so the ith component of Px can be at most $\max_i |x_i|$.

Since f is a contraction mapping defined on the complete metric space M, the proposition follows from Banach's fixed point theorem.[22]

The following lemma states a number of iterations k sufficient for the computation of $x^{(k)}$ to a desired precision ϵ. Note that due to the several inequalities used, the number of necessary iterations will, in general, be lower than k.

Lemma 1. *For given* $\epsilon > 0$, $||x^{(k)} - \lim_{n\to\infty} x^{(n)}||_\infty \leq \epsilon$ *after at most* $k = \lceil \log_\alpha((1-\alpha)\epsilon/||x^{(0)} - x^{(1)}||_\infty)\rceil$ *iterations.*

Proof. We want to find $k \geq 0$ such that $||x^{(k)} - x^{(k+m)}|| < \epsilon$ for all $m \geq 0$. By repeated application of the triangle inequality we get

$$||x^{(k)} - x^{(k+m)}||_\infty \leq \sum_{l=0}^{m-1} ||x^{(k+l)} - x^{(k+l+1)}||_\infty$$

Using Theorem 1 gives

$$||x^{(p)} - x^{(p+1)}||_\infty \leq \alpha||x^{(p-1)} - x^{(p)}||_\infty \leq \alpha^2||x^{(p-2)} - x^{(p-1)}||_\infty \leq ... \leq \alpha^p||x^{(0)} - x^{(1)}||_\infty$$

Combining these equations and applying the geometric series yields

$$||x^{(k)} - x^{(k+m)}||_\infty \leq \sum_{l=0}^{m-1} \alpha^{k+l}||x^{(0)} - x^{(1)}||_\infty \leq \alpha^k||x^{(0)} - x^{(1)}||_\infty \sum_{l\geq 0} \alpha^l = \frac{\alpha^k}{1-\alpha}||x^{(0)} - x^{(1)}||_\infty$$

If $x^{(0)} = x^{(1)}$, we are done. Otherwise, solving for k gives us

$$\frac{\alpha^k}{1-\alpha}||x^{(0)} - x^{(1)}||_\infty \leq \epsilon \Leftrightarrow k \geq \log_\alpha \frac{(1-\alpha)\epsilon}{||x^{(0)} - x^{(1)}||_\infty}$$

Since k is an integer, the proposition follows.

2.4. A Similarity Measure for Molecular Graphs. Combining the ideas of optimal assignments (eq 1) and iterative similarity for molecular graphs (eq 4), we obtain our new molecular similarity measure as

$$k_a(G,G') = \max_{\pi} \sum_{i=1}^{|V|} k_x(v_i, v'_{\pi(i)})$$

$$\text{with } k_x(v_i, v'_j) = (\lim_{n\to\infty} x^{(n)})_{(j-1)|V|+i}$$

k_x can be computed iteratively to a given precision ϵ using eq 5 and Lemma 1. To do so, in each iteration and for each pair of vertices v_i, v'_j, an optimal assignment π of the neighbors of v_i to the neighbors of v'_j (or vice versa; see eq 4) has to be found, based on the similarity values of the previous iteration. For each π, there are $|v'_j|!/(|v'_j| - |v_i|)!$ possibilities, leading to a factorial worst-case runtime complexity for general graphs. However, in molecular graphs the vertex degree is bounded by a small constant, up to which the corresponding assignments can be precomputed, allowing the determination of each π in constant time. As k_v and k_e can be computed in constant time as well, each iteration takes time in $O(|V||V'|)$. By Lemma 1, there are at most $k = \lceil \log_\alpha((1-\alpha)\epsilon/||x^{(0)} - x^{(1)}||_\infty)\rceil$ iterations. Since $||x^{(0)} - x^{(1)}||_\infty \leq 1$, the computation of the similarity matrix X takes time in $O(|V||V'|\log_\alpha (1 - \alpha)\epsilon)$. The final optimal assignment over X can be done using the Kuhn-Munkres assignment algorithm,[18−20] which has cubic runtime. Empirical runtimes are given in Table 1.

Table 1. Empirical Runtimes in Units of 10^{-2} s for a Java 1.5 (java.sun.com) Implementation of Our Algorithm Running on an Intel (intel.com) Xeon Processor (2.2 GHz, 3 GB RAM)[a]

	α								
	1/100	1/8	2/8	3/8	4/8	5/8	6/8	7/8	99/100
runtime	0.09	0.13	0.16	0.18	0.22	0.30	0.44	0.92	14.46

[a] Values for each α are average runtimes over all data sets and choices of k_v, k_e. The average runtime over all computed kernel values was 1.88 ± 0.17.

Table 2. Support Vector Machine and Kernel Parameter Values Used in the Parameter Optimization Grid Search

param.	kernel	values
α	sim meas	(1/100), (1/8), (2/8), (3/8), (4/8), (5/8), (6/8), (7/8), (99/100)
d	k_{poly}	1,2,3,4,5
γ	k_{rbf}	$\{2^k \mid k \in \mathbb{N}, -10 \leq k \leq 3\}$

Table 3. Tested Parametrizations of Our Molecular Similarity Measure[a]

abbrev	description
	Vertex Comparisons
none	$k_v(v,v') = 1$. No vertex kernel.
delement	$k_v(v,v') = 1_{l(v)=l(v')}$. Dirac kernel using the element types as vertex labels.
dppp	$k_v(v,v') = 1_{l(v)=l(v')}$. Dirac kernel using potential pharmacophore points (Table 5) as vertex labels.
echarge	$k_v(v,v') = \exp(-\lvert l(v) - l(v')\rvert^2/2\sigma^2)$. Gaussian kernel using Gasteiger−Marsili partial charges[27] as vertex label; σ was set to the standard deviation of the partial charges in a data set.
	Edge Comparisons
none	$k_e(e,e') = 1$. No edge kernel.
dbond	$k_e(e,e') = 1_{l(e)=l(e')}$. Dirac kernel using the bond type (single, double, triple) as edge label.

[a] All possible 8 combinations of vertex and edge comparisons were used.

3. EXPERIMENTS

We carried out retrospective virtual screening experiments using support vector machines (SVMs)[4,23] for binary classification and regression on the data sets shown in Table 4. We analyze the results of our method and compare them to the results of a related approach from the literature.

3.1. Virtual Screening Method. In the standard machine learning approach to virtual screening, a computational learning method is trained on a set of known biologically active and inactive (with respect to the target under investigation) molecules; the trained model is then applied to classify a large number of new molecules, e.g., from a vendor database. The aim is to identify active molecules not in the training set. In a prospective study, the actual activity of the found molecules is tested in a laboratory using an assay. In a retrospective study, performance is assessed using cross-validation on the training set instead.

We employed a soft margin, C-parameter variant of SVMs for binary classification and a C-parameter variant of SVMs

Table 4. Data Sets Used[a]

	samples		
abbrev	neg	pos	description
Drug	734	809	known and desirable bioactivity
AChE	58	92	acetylcholinesterase inhibitors
COX-2	136	126	cyclooxygenase-2 inhibitors
DHFR	60	60	dihydrofolate reductase inhibitors
FXa	228	221	factor Xa inhibitors
PPAR	94	92	peroxisome proliferator activated receptor ligands
Thrombin	185	186	thrombin inhibitors
PTCfm	202	135	PTC, female mice subset
PTCmm	204	118	PTC, male mice subset
PTCfr	224	118	PTC, female rats subset
PTCmr	186	146	PTC, male rats subset
BBB	115		blood-brain barrier

[a] Refer to section 3.2 for details. PTC = predictive toxicology challenge 2000−2001.

with ϵ-insensitive loss function for regression. In both cases, a modified version of the SVM^{light} package[24] was used as the implementation. As kernels we used the parametrizations of our similarity measure given in Table 3 and two standard kernels, the polynomial kernel $k_{\text{poly}}(x,y) = \langle x,y\rangle^d$ and the radial basis function kernel $k_{\text{rbf}}(x,y) = \exp(-\gamma\langle x - y, x - y\rangle)$. For the standard kernels, molecules were represented as vectors using the established Ghose-Crippen descriptors[25] and CATS2D descriptors.[26]

To evaluate performance, we used 10 runs of stratified 10-fold cross-validation[28] for each of the kernel parametrizations (α, d, γ) given in Table 2. As a performance measure, we computed the average (over all runs and cross-validation folds) correlation coefficient[29,30]

$$\text{cc} = \frac{\text{tp} \cdot \text{tn} - \text{fp} \cdot \text{fn}}{\sqrt{(\text{tp} + \text{fn})(\text{tp} + \text{fp})(\text{tn} + \text{fp})(\text{tn} + \text{fn})}}$$

where tp, tn, fp, and fn are the numbers of true positives, true negatives, false positives, and false negatives, respectively. The SVM parameter C was optimized on the training folds of each cross-validation run using a uniform grid search in log parameter space, $C \in \{2^k \,|k \in \mathbb{N}, -10 \leq k \leq 11\}$. For SVM regression, we set $\epsilon = 3\sigma\sqrt{(\ln n)/n}$, where n is the number of training samples, as proposed by Cherkassky and Ma.[31]

Note that in a prospective setting, the same cross-validation based methodology for parameter optimization can be used. The final classifier would then be trained using the parameters determined via cross-validation and all of the training data. We do not provide a single set of default parameter values as a good choice of parameters depends on the problem at hand (compare Table 6).

3.2. Data Sets. We used 12 data sets (summarized in Table 4) from 4 different sources. From these, 11 data sets are binary classification problems and one (*BBB*) is a regression problem.

The *Drug* data set was compiled using the DrugBank repository (redpoll.pharmacy.ualberta.ca/drugbank) and randomly sampled molecules from the Sigma-Aldrich catalog (sigmaaldrich.com). The data sets *AChE*, *COX-2*, *DHFR*, *FXa*, *PPAR*, and *Thrombin* are subsets of the COBRA database,[32] version 6.1. For each data set, all molecules belonging to the respective class were taken as positive samples, and an identical number of molecules was randomly selected from the database as negative samples. The data sets *PTCfm*, *PTCmm*, *PTCfr*, and *PTCmr* are binary classification subproblems of the predictive toxicology challenge 2000−2001.[33] The *BBB* data set was published by Hou and Xu.[34]

Table 5. Molecular Query Language (MQL) Definitions of the Used Potential Pharmacophore Points (PPP)

PPP	MQL definition
lipophilic	C[!bound(∼ Hetero)], Cl, Br, I
positive	*[charge > 0], N[allHydrogens > 1]
negative	*[charge < 0], O═P′ ∼ O′, O═S′ ∼ O′, O═C′ ∼ O′[allHydrogens=1 | charge < 0], O[allHydrogens=1 | charge < 0] ∼ C′═O
acceptor	O, N[allHydrogens=0]
donor	O[allHydrogens=1&!bound(−C═O)], N[allHydrogens > 0]

In all data sets, duplicate molecules and molecules that could not be processed by the used software were removed; molecular graphs did not include hydrogen atoms. Pharmacophore types were computed using the molecular query language;[35] Table 5 shows the corresponding definitions. Gasteiger−Marsili partial charges were computed using the PETRA software (version 3.11, mol-net.de).

3.3. Results. We used the data sets *Drug*, *AChE*, *COX-2*, *DHFR*, *FXa*, *PPAR*, and *Thrombin* to assess the performance of our similarity measure on a publicly available data set (*Drug*) and on a high-quality pharmacological data set (COBRA subsets). We used the other data sets (predictive toxicology challenge subsets, *BBB*) to compare our similarity measure to another graph-based approach from the literature, the optimal assignment kernel (cf. remarks on page 2) by Fröhlich et al.[11] Table 6 shows for each data set the best performing parametrization of each method with regard to 10-fold cross-validation averaged over 10 runs. On 11 out of 12 data sets, our similarity measure outperforms standard kernel/descriptor combinations as well as the optimal assignment kernel. On the remaining data set, our method performs about as good as standard kernel/descriptor combinations. The results for the latter were consistent with those of other studies.[36,37]

3.4. Expressiveness. As shown before, the time needed to compute our similarity measure increases with α and $1/\epsilon$. Such an increase in computation time should lead to an improvement in discriminative power, which is the case for our method: The number of molecule pairs that our similarity measure can separate, i.e., that have similarity < 1, increases monotonically with α (see the Supporting Information).

3.5. Positive Semidefiniteness. We have checked all the Gram matrices of our similarity measure computed in the previous experiments for positive semidefiniteness by the eigenvalue criterion using the MATLAB software (R2006b, mathworks.com). Negative eigenvalues were encountered but might be partly or entirely numerical artifacts. Figure 2 shows the distribution of the smallest eigenvalues by α-values. Most of the negative smallest eigenvalues are close to zero, which means that the corresponding matrices can be made positive semidefinite using only a small correction. [Adding the absolute value $|\lambda|$ of the smallest eigenvalue λ of a Gram matrix to its diagonal renders the matrix positive semidefinite. However, for larger $|\lambda|$ this can worsen performance due to diagonal dominance.[39]] Further, for $\alpha \rightarrow 1$ all eigenvalues

Table 6. Performance of Our Similarity Measure, Standard Kernel/Descriptor Combinations, and the Optimal Assignment Kernel[11,38] [a]

(a) Performance of Our Similarity Measure and Standard Kernel/Descriptor Combinations

	standard kernels and descriptors			own similarity measure		
data set	parameters	cc	pc	parameters	cc	pc
Drug	rbf/gc, $C = 97$, $\gamma = 2^{-6}$	0.745 ± 0.04	87.2 ± 2.2	dppp/dbond, $C = 2$, $\alpha = 0.875$	**0.777** ± **0.04**	**88.9** ± **2.0**
AChE	rbf/gc, $C = 6$, $\gamma = 2^{-6}$	0.874 ± 0.13	93.2 ± 7.1	delem/none, $C = 1$, $\alpha = 0.875$	**0.926** ± **0.09**	**96.0** ± **5.1**
COX-2	poly/gc, $C = 9$, $d = 3$	**0.861** ± **0.09**	**92.9** ± **4.7**	dppp/dbond, $C = 2$, $\alpha = 0.875$	0.858 ± 0.09	92.4 ± 4.7
DHFR	rbf/cats2d, $C = 1$, $\gamma = 2^{-2}$	0.983 ± 0.05	99.1 ± 2.6	none/none, $C = 1$, $\alpha = 0.875$	**0.994** ± **0.03**	**99.7** ± **1.6**
FXa	poly/cats2d, $C = 2$, $d = 5$	0.945 ± 0.05	97.2 ± 2.5	echarge/none, $C = 3$, $\alpha = 0.875$	**0.973** ± **0.03**	**98.6** ± **1.6**
PPAR	rbf/cats2d, $C = 3$, $\gamma = 2^{-2}$	0.822 ± 0.12	90.7 ± 6.5	dppp/none, $C = 3$, $\alpha = 0.625$	**0.989** ± **0.09**	**95.2** ± **4.6**
Thrombin	poly/cats2d, $C = 3$, $d = 4$	0.891 ± 0.07	94.4 ± 3.7	dppp/dbond, $C = 2$, $\alpha = 0.875$	**0.930** ± **0.06**	**96.3** ± **2.9**

(b) Performance of Standard Kernel/Descriptor Combinations, the Optimal Assignment Kernel,[11,38] and Our Own Method

		standard method		optimal assignment		own similarity measure	
data set	perf	parameters	performance	parameters	performance	parameters	performance
PTCfm	pc	rbf/gc, $C = 4$, $\gamma = 2^{-4}$	64.1 ± 4.5	OARG	64.0 ± 3.3	delem/dbond, $\alpha = 0.875$	**71.1** ± **5.9**
PTCmm	pc	rbf/gc, $C = 1$, $\gamma = 2^{-3}$	66.4 ± 3.5	OARG	67.8 ± 2.3	echarge/dbond, $\alpha = 0.125$	**72.8** ± **5.2**
PTCfr	pc	poly/gc, $C = 110$, $d = 1$	68.4 ± 3.5	OA	66.9 ± 1.6	delem/none, $\alpha = 0.875$	**72.4** ± **5.5**
PTCmr	pc	poly/cats2d, $C = 67$, $d = 1$	64.9 ± 6.2	OA	63.3 ± 2.4	delem/dbond, $\alpha = 0.75$	**69.4** ± **6.4**
BBB	r^2	rbf/cats2d, $C = 0.1$, $\gamma = 0$	0.598 ± 0.199	OA	0.603 ± 0.063	delem/dbond, $\alpha = 0.625$	**0.58** ± **0.178**

[a] For each data set, the parametrization (of each method) with best averaged cross-validated performance is shown. Performance measures are correlation coefficient (cc) and the percentage of correctly classified samples (pc) for binary classification as well as the squared error (r^2) for regression. Numbers are mean ± standard deviation; the best performance for each data set is printed in bold face: poly = polynomial kernel, rbf = radial basis function kernel, gc = Ghose-Crippen descriptor, cats2d = CATS2D descriptor; OA = optimal assignment kernel, OARG = optimal assignment kernel on reduced graphs. The parametrizations of our similarity measure are listed in Table 3. For the echarge vertex comparison on the *FXa* data set, $\gamma = 0.1622$; on the *PTCmm* data set, $\gamma = 0.1313$.

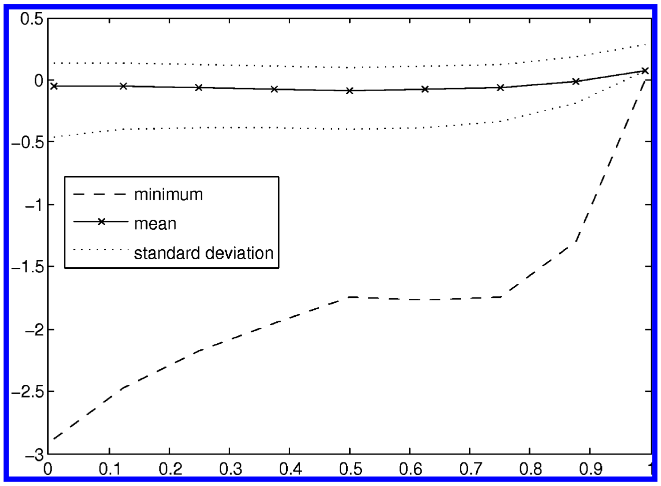


Figure 2. Plot of the smallest Gram matrix eigenvalue for different α-values of our molecular similarity measure. Minimum, average, and standard deviation are calculated over all parametrizations and all data sets. The dotted lines show the mean ± the standard deviation of the negative and positive eigenvalues only, respectively.

are non-negative, indicating that the recursive similarity part of our molecular similarity measure might be a kernel.

4. CONCLUSIONS

We have introduced a molecular similarity measure defined directly on the annotated molecular graph, based on a recursive concept of graph similarity and optimal assignments. An iterative algorithm was given, together with an upper bound on the necessary number of iterations. The method converges to a unique solution independent of initialization, incorporates chemical domain knowledge, and exploits structural properties of the molecular graph.

We successfully demonstrated the feasibility of our approach by retrospective virtual screening of several data sets. On those, our method showed on par or superior performance compared to both optimized combinations of standard kernels/descriptors and the similar method of Fröhlich et al.[11,38] A comprehensive comparison is, however, beyond the scope of this article. Altogether, the performance of our molecular similarity measure is encouraging. We currently test it in a prospective drug discovery project.

Although it seems that in general our similarity measure is not positive semidefinite, for practical purposes this shortcoming can be easily remedied. Further, empirical evidence suggests that our similarity measure is positive semidefinite for $\alpha \to 1$. However, this reduces eq 4 to the recursive graph similarity part, thereby losing information about vertex comparisons.

Possible extensions of our work include the reformulation of eq 4 to include vertex comparisons in the case of $\alpha \to 1$ and directly (instead of iteratively) solving the nonlinear system of equations underlying our similarity measure. More chemically motivated extensions include graph reduction, e.g., by a predefined dictionary of subgraphs or by extracting such a dictionary from the data.

A Java implementation of the algorithm is available via the Internet at modlab.de.

ACKNOWLEDGMENT

The authors thank Dr. Petra Schneider for compiling the COBRA data sets, Norbert Dichter for technical assistance, and Dr. Holger Fröhlich for helpful discussions. This research was supported by the Beilstein-Institut zur Förderung der Chemischen Wissenschaften, Frankfurt am Main, and the Deutsche Forschungsgemeinschaft (SFB 579 “RNA-Ligand Interaction”, project A11).

Supporting Information Available: Table of the expressiveness (number of nonseparable molecule pairs) of our similarity measure for each data set and parametrization. This material is available free of charge via the Internet at pubs.acs.org.

REFERENCES AND NOTES

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

(2) Willett, P. Chemical similarity searching. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 983–996.

(3) Todeschini, R.; Consonni, V. *Handbook of Molecular Descriptors*; Wiley-VCH: Weinheim, Germany, 2000.

(4) Shawe-Taylor, J.; Cristianini, N. *Kernel methods for pattern analysis*; Cambridge University Press: New York, NY, 2004.

(5) Haussler, D. *Convolution kernels on discrete structures*; Technical Report UCSC-CRL-99-10; Department of Computer Science, University of California: Santa Cruz, CA, 1999.

(6) Gärtner, T. A survey of kernels for structured data. *ACM SIG Knowledge Discovery Data Min. Explor. Newsl.* **2003**, *5*, 49–58.

(7) Kashima, H.; Tsuda, K.; Inokuchi, A. Marginalized kernels between labeled graphs. Proceedings of the 20th International Conference on Machine Learning (ICML 2003), Washington, DC, U.S.A., 2003; Fawcett, T., Mishra, N., Eds.; AAAI Press: Menlo Park, CA, pp 321–328.

(8) Gärtner, T.; Lloyd, J.; Flach, P. Kernels and distances for structured data. *Machine Learning* **2004**, *57*, 205–232.

(9) Mahé, P.; Ueda, N.; Akutsu, T.; Perret, J.-L.; Vert, J.-P. Extensions of marginalized graph kernels. Proceedings of the 21st International Conference on Machine Learning (ICML 2004), Banff, Canada, 2004; Brodley, C., Ed.; Omnipress: Madison, WI, pp 552–559.

(10) Borgwardt, K.; Ong, C. S.; Schönauer, S.; Vishwanathan, S.; Smola, A.; Kriegel, H.-P. Protein function prediction via graph kernels. *Bioinformatics* **2005**, *21S1*, i47–i56.

(11) Fröhlich, H.; Wegner, J.; Sieker, F.; Zell, A. Optimal assignment kernels for attributed molecular graphs. Proceedings of the 22nd International Conference on Machine Learning (ICML 2005), Bonn, Germany, 2005; de Raedt, L., Wrobel, S., Eds.; Omnipress: Madison, WI, pp 225–232.

(12) Jain, B.; Geibel, P.; Wysotzki, F. SVM learning with the Schur-Hadamard inner product for graphs. *Neurocomputing* **2005**, *64*, 93–105.

(13) Menchetti, S.; Costa, F.; Frasconi, P. Weighted decomposition kernels. Proceedings of the 22nd International Conference on Machine Learning (ICML 2005), Bonn, Germany, 2005; de Raedt, L., Wrobel, S., Eds.; Omnipress: Madison, WI.

(14) Ralaivola, L.; Swamidass, S.; Saigo, H.; Baldi, P. Graph kernels for chemical informatics. *Neural Networks* **2005**, *18*, 1093–1110.

(15) Ramon, J.; Gärtner, T. Expressivity versus efficiency of graph kernels. Presented at Proceedings of the 1st International Workshop on Mining Graphs, Trees and Sequences (MGTS 2003) [Online], Cavtat-Dubrovnik, Croatia, 2003. University of Osaka, Institute for Scientific and Industrial Research Web site. www.ar.sanken.osaka-u.ac.jp/MGTS-2003CFP.html (accessed Sep 17, 2007).

(16) Gärtner, T.; Flach, P.; Wrobel, S. On graph kernels: Hardness results and efficient alternatives. In *Learning Theory and Kernel Machines*, Proceedings of the 16th Annual Conference on Learning Theory and 7th Kernel Workshop (COLT/Kernel 2003), Washington, DC, U.S.A., 2003; Schölkopf, B., Warmuth, M., Eds.; Springer: Berlin, Germany, pp 129–143.

(17) Fröhlich, H.; Wegner, J.; Zell, A. Assignment kernels for chemical compounds. Proceedings of the 2005 International Joint Conference on Neural Networks (IJCNN 2005), Montréal, Canada, 2005; Elsevier: Amsterdam, The Netherlands, pp 913–918.

(18) Kuhn, H. The Hungarian method for the assignment problem. *Bull. Am. Math. Soc.* **1955**, *61*, 557–558.

(19) Munkres, J. Algorithms for the assignment and transportation problems. *J. Soc. Ind. Appl. Math.* **1957**, *5*, 32–38.

(20) Bourgeois, F.; Lassalle, J.-C. An extension of the munkres algorithm for the assignment problem to rectangular matrices. *Comm. ACM* **1971**, *14*, 802–804.

(21) Zager, L. Graph similarity and matching, Master's thesis, Massachusetts Institute of Technology: Cambridge, MA, 2005.

(22) Granas, A.; Dugundji, J. *Fixed point theory*; Springer: New York, NY, 2003.

(23) Boser, B.; Guyon, I.; Vapnik, V. A training algorithm for optimal margin classifiers. Proceedings of the 5th Annual ACM Conference on Computational Learning Theory (COLT 1992), Pittsburgh, PA, 1992; ACM: pp 144–152.

(24) Joachims, T. Making large-scale SVM learning practical. In *Advances in Kernel Methods: Support Vector Learning*; Schölkopf, B., Burges, C., Smola, A., Eds.; MIT: Cambridge, MA, 1999; pp 169–184.

(25) Viswanadhan, V.; Ghose, A.; Revankar, G.; Robins, R. Atomic physicochemical parameters for three dimensional structure directed quantitative structure-activity relationships. 4. Additional parameters for hydrophobic and dispersive interactions and their applications for an automated superposition of certain naturally occurring nucleoside antibiotics. *J. Chem. Inf. Comput. Sci.* **1989**, *29*, 163–172.

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

(27) Gasteiger, J.; Marsili, M. Iterative partial equalization of orbital electronegativity – a rapid access to atomic charges. *Tetrahedron* **1980**, *36*, 3219–3228.

(28) Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI 1995), Montréal, Canada, 1995; Morgan Kaufmann: San Francisco, CA, pp 1137–1145.

(29) Matthews, B. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim. Biophys. Acta* **1975**, *405*, 442–451.

(30) Baldi, P.; Brunak, S.; Chauvin, Y.; Andersen, C.; Nielsen, H. Assessing the accuracy of prediction algorithms for classification: An overview. *Bioinformatics* **2000**, *16*, 412–424.

(31) Cherkassky, V.; Ma, Y. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks* **2004**, *17*, 113–126.

(32) Schneider, P.; Schneider, G. Collection of bioactive reference compounds for focused library design. *QSAR Comb. Sci.* **2003**, *22*, 713–718.

(33) Helma, C.; Kramer, S. A survey of the predictive toxicology challenge 2000–2001. *Bioinformatics* **2003**, *19*, 1179–1182.

(34) Hou, T.; Xu, X. ADME evaluation in drug discovery. 3. Modeling blood-brain barrier partitioning using simple molecular descriptors. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 2137–2152.

(35) Proschak, E.; Wegner, J.; Schüller, A.; Schneider, G.; Fechner, U. Molecular query language (MQL) – A context-free grammar for substructure matching. *J. Chem. Inf. Model.* **2007**, *47*, 295–301.

(36) Byvatov, E.; Fechner, U.; Sadowski, J.; Schneider, G. Comparison of support vector machine and artificial neural network systems for drug/nondrug classification. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1882–1889.

(37) Müller, K.-R.; Rätsch, G.; Sonnenburg, S.; Mika, S.; Grimm, M.; Heinrich, N. Classifying 'drug-likeness' with kernel-based learning methods. *J. Chem. Inf. Model.* **2005**, *45*, 249–253.

(38) Fröhlich, H.; Wegner, J.; Sieker, F.; Zell, A. Kernel functions for attributed molecular graphs – A new similarity-based approach to ADME prediction in classification and regression. *QSAR Comb. Sci.* **2006**, *25*, 317–326.

(39) Greene, D.; Cunningham, P. Practical solutions to the problem of diagonal dominance in kernel document clustering. Proceedings of the 23rd International Conference on Machine Learning (ICML 2006), Pittsburgh, PA, 2006; Cohen, W., Moore, A., Eds.; ACM: pp 377–384.

CI700274R