International Journal of Chemical, Materials and Biomolecular Sciences
ISSN: 2415-6620
Vol:1, No:12, 2007

Weighted *k*-Nearest-Neighbor Techniques for High Throughput Screening Data

Kozak K, M. Kozak, K. Stapor

***Abstract*—**The k-nearest neighbors (knn) is a simple but effective method of classification. In this paper we present an extended version of this technique for chemical compounds used in High Throughput Screening, where the distances of the nearest neighbors can be taken into account. Our algorithm uses kernel weight functions as guidance for the process of defining activity in screening data. Proposed kernel weight function aims to combine properties of graphical structure and molecule descriptors of screening compounds. We apply the modified knn method on several experimental data from biological screens. The experimental results confirm the effectiveness of the proposed method.

***Keywords*—**biological screening, kernel methods, KNN, QSAR

I. Introduction

COMPLETE HIGH THROUGHPUT SCREENING (HTS) process can be divided in two parts: primary screening and secondary screening. Primary screening involves the testing of large compound libraries against targets to generate "hits": small proportion of the tests that show an effect. Secondary screening is the further investigation of hits. The k-nearest neighbors (knn) is a non-parametric classification method, which is simple but effective in many cases [2]. Many researchers have found that the knn algorithm achieves very good performance in their experiments on different data sets [21][13][17]. Also in categorization of biological screen data is one of the most popular algorithms [7]. In knn algorithm data record t to be classified, its k nearest neighbors are retrieved, and this forms a neighborhood of t. Majority voting among the data records in the neighborhood is usually used to decide the classification for t with or without consideration of distance-based weighting. The k nearest neighbors are selected to make an equal contribution to the prediction of a test compound, no matter where they are located relative to the test point. In a sparse region, the k-thy point may be far away from the test point and be unrelated to the test point and have little or no prediction power. However, this point must contribute to the prediction the same as the other k-1 compounds according to the knn rule. This does not sound reasonable.

A number of locally adaptive methods have recently been proposed to address the distance selection issue in knn. Friedman [5] describes a flexible metric that recursive on a query along the most (locally) relevant dimension, where local relevance is determined from a reduction in error between successive predictions. In [8], Hastie and Tibshirani propose a discriminant adaptive NN (DANN) method that computes a distance metric as a product of properly weighted local within and between sum-of-squares matrices. Domeniconi et al. [3] describe a locally adaptive NN method by approximating the Chi-squared distance (ADAMENN) [8], [14], [20]. The technique employs a "patient" averaging process to reduce variance. MORF [15] is another adaptive NN method where, for a query, a local linear support vector machine (SVM) is built and the normal of the SVM is used to determine discriminant feature dimensions for classifying the query.

This extension of knn proposed in this paper is based on the idea that such observations (chemical compound) within the learning set, which are particularly close to the new observation, should get a higher weight in the decision than such neighbors that are far away from tested element. This is not the case with standard knn: Indeed only the k nearest neighbors influence the prediction; however, this influence is the same for each of these neighbors, although the individual similarity to tested element might be widely different. We develop a much more flexible algorithm that extends the basic method. We introduce a weighting scheme for the nearest neighbors according to their similarity to a new observation using kernel function for screening compounds.

The rest of the paper is organized as follows. In section 2 we define the probability for decisions made by the majority rule based on a finite number of observations. The details of the proposed method are presented in section 3. In section 4 we test our method on biological screen datasets. The conclusion will be given in section 5.

II. Classification

The final decision in a recognition task is affected by two types of "a priori" knowledge: the number of samples previously seen of each of the objects to be recognized, and the discriminant power provided by the features extracted. The prior knowledge is reflected in the *a priori probabilities* that measure how likely we are to find each type in the data set. The proportions in which each type (class) is present in the sample area may provide such a measure. If we let ω_i (i = 1,...,M) denote the *state of nature*, i.e. the variable indicating the M possible classes, $P(\omega_i)$ denote the *a priori probabilities*. Generally speaking ω_i are called *classes* and the prior knowledge available is used to estimate the a *priori probabilities*.

Kozak K. Max Planck Institute of Molecular Cell Biology, Pfothenhauerstr 108, 01307 Dresden, Germany (corresponding author to provide phone: +49 351 210 2913; e-mail: kozak@mpi-cbg.de).

If we let x denote the vector containing a set of measurements (parameters), the *state-conditional probability density* $p(x|\omega_i)$ express the probability density function for x given that the state of nature is ω_i. The state conditional probability densities can also be estimated from the samples for each class. The two probabilistic measures derived by the samples, the *a priori* probability for each class $p(\omega_i)$ and the *state-conditional probability density (scpd)* $p(x|\omega_i)$, can be used to estimate the *a posteriori* probability $P(\omega_i|x)$ by means of Bayes rule:

$$P(\omega_i \mid x) = \frac{p(x \mid \omega_i)p(\omega_i)}{\sum_{j=1}^{M} p(x \mid \omega_j)P(\omega_j)}, \quad (1)$$

Given the feature vector x of an unclassified pattern, classification is carried out estimating the posterior probabilities for each class and, deciding for the class with the higher posterior probability value. The effect of such a decision rule is to divide the feature space into M decision regions.

According to Bayes rule, the class with the largest posterior probability is selected as the label of x. Ties are broken randome. Bayes rule guarantees the minimum misclassification rate. Sometimes the misclassifications rate differently for different classes. Then we can use a *loss matrix* $\Lambda=[\lambda_{ij}]$, where λ_{ij} is a measure of the loss incurred if we assign class label ω_i when the true label is ω_j. The *minimum risk classifier* assigns x to the class with the minimum expected risk:

$$R_x(\omega_i) = \sum_{j=1}^{M} \lambda_{ij} P(\omega_j \mid x). \quad (2)$$

In general, the classifier output can be interpreted as a set of M degrees of support, one for each class (discriminant scores obtained through discriminant functions). We label x in the class with the largest support. In practice, a priori probabilities and the scpd are not known. The scpd can be estimated from the data using either a parametric or nonparametric approach.

When we look at the classification problem in a supervised classifier, we observe a labeled (training) set of n observations $O_n = \{(x_1, \omega_1), \ldots, (x_i, \omega_i)\}$, where x_i are the feature vectors, ω_i are scalar labels of the d-dimensional real vectors X_i and (x_i, ω_i) are assumed to be from some unknown distribution Q of (x, ω) on n dimensional space R with M number of classes ω, $R^d \times \{\omega_1, \ldots, \omega_M\}$. Here we assume simply that the training vectors are random vectors in d-dimensional Euclidean space with well-behaved distributions and a well-defined density function f.

The goal in supervised learning is to design a function $\Phi_n : R^d \rightarrow \{\omega_1, \ldots, \omega_M\}$ that maps a new feature vector x drawn from Q to its desired class from $\{\omega_1, \ldots, \omega_M\}$. K-nearest neighbor (knn) belongs to one nonparametric approach in supervised learning strategies.

A. Probability in the k-nearest neighbors method

Given n samples we can easily estimate the joint probability $p(x, \omega_i)$, by placing a cell of volume V around observation x. Let k_i be the number of samples labeled ω_i captured by the cell, then the joint probability can be estimated as:

The above measure can be use to provide a reasonable

$$P_n(x, \omega_i) = \frac{k_i / n}{V}, \quad (3)$$

estimate of the posterior probability:

$$P_n(\omega_i \mid x) = \frac{p_n(x, \omega_i)}{\sum_{i=1}^{M} p_n(x, \omega_i)} = \frac{k_i}{k}, \quad (4)$$

where k is the total number of samples captured by the cell. Calculation of probability in this case is based on density estimation via Bayes theorem. Generally the knn rule suggests classifying x by assigning it to the class that appears most frequently among its k nearest neighbors. Let k_i denote the number of observations from the group of the nearest neighbors that belong to class i:

$$\sum_{i=1}^{M} k_i = k,, \quad (5)$$

Then a new observation is predicted into the class ω with:

$$k_\omega = \max_i (k_i), \quad (6)$$

As we know, selecting ideal k neighbors in practical data is quite difficult, because only a finite amount of training data is available. The way that knn takes an average of the k nearest points is a discrete process. The k nearest compounds is selected to make an equal contribution to the prediction of a test compound, no matter where they are located relative to the test point. In a sparse region, the k-thy compound may be far away from the test compound and be unrelated to the test compound and have little or no prediction power. However, this compound must contribute to the prediction the same as the other k-1 compounds according to the knn rule. This does not sound reasonable. Here, we propose method to avoid the problem by combining knn with kernel weights. We can automatically give the points close to the test compound more weight and the points far away less weight. A kernel-weighted knn was already proposed and implemented by Hchenbichler and Schliep (2004) [23]. The difference in our algorithm is a kernel function: we combine in kernel properties of graphical structure and molecule descriptors of screening compound together. The use of such kernels allows comparison of compounds, not only on graphs but also on molecular descriptors which play important role in classification of molecules according to their biological activity.

III. Weighted k-Nearest-Neighbors

The classification version of weighted knn is concieved to predict nominal classes and works with a weighted majority vote of the nearest neighbors:

$$\hat{P}_n(\omega_i \mid x) = \frac{p_n(x,\omega_i)\cdot w_i}{\sum_{i=1}^{M} p_n(x,\omega_i)\cdot w_i} = \frac{k_i \cdot w_i}{k \cdot w}, \qquad (7)$$

where w is a function of the distance between the ith nearest neighbor and the test point. The distances on which the search for the nearest neighbors is based in the first step, have to be transformed into similarity measures which can be used as weights.

A. Kernel function as weight.

The transition from distances to weights then follows in the second step according to any arbitrary kernel function. There are several kernel weight functions, and we introduced new kernel for chemical compounds used in screening process. Because chemical compounds are often represented by the graph of their covalent bonds, machine learning methods in this domain must be capable of processing graphical structures with variable size. The topology of chemical compounds can be represented as labeled graphs, where edge labels correspond to bond properties like bond order, length of a bond, and node labels to atom properties, like partial charge, membership to a ring, and so on. This representation opens up the opportunity to use graph mining methods to deal with molecular structures. Selecting optimal similarity features of a molecule based on molecular graph or descriptors is a critical and important step, especially if one is interested in quantifying structure-activity relationship (QSAR) studies. It has been shown [24, 11, 16] that the quality of the inferred model strongly depends on the selected molecular properties (graph or descriptors).

Optimal Assignment (OA) kernel [9] is one graph kernel used for chemical molecules. Let us assume now we have two molecules m and m', which have atoms $a_1,\ldots,a_{|m|}$ and $a'_1,\ldots,a'_{|m'|}$. Let us further assume we have a kernel k_{nei}, which compares a pair of atoms (a_h, $a'_{h'}$) from both molecules, including information on their neighborhoods, membership to certain structural elements and other characteristics. Then a valid kernel between m, m' is the optimal assignment kernel:

$$k_{OA}(m,m') := \begin{cases} \max_\pi \sum_{h=1}^{|m|} k_{nei}(a_h, a'_{\pi(h)}) & if\ |m'| \geq |m| \\ \max_\pi \sum_{j=1}^{|m'|} k_{nei}(a_{\pi(h')}, a'_{h'}) & otherwise \end{cases} \qquad (8)$$

where k_{nei} is calculated based on two kernels k_{atom} and k_{bond} which compare the atom and bond features, respectively. These feature vectors should include various information, for instance, whether an atom belongs to a ring, if it is in a donor or acceptor, what partial charge it has and so on. A natural choice for k_{atom} and k_{bond} would be the RBF-kernel, which computes the similarity of the feature vectors associated to a pair of atoms or bonds. Let us denote by $a \rightarrow n_i(a)$ the bond connecting atom a with its *i*th neighbor $n_i(a)$. Let us further denote by |N(a)| the number of neighbors of atom a. We now define a kernel R_o, which compares all direct neighbors of atoms (a, a') as the optimal assignment kernel between all neighbors of a and a_0 and the bonds leading to them, i.e.

$$R_o(a,a') = \frac{1}{|N(a')|} \max_\pi \sum_{i=1}^{|N(a)|} (k_{atom}(n_i(a), n_{\pi(i)}(a')) \cdot k_{bond}(a \rightarrow n_i(a), a' \rightarrow n_{\pi(i)}(a'))) \qquad (9)$$

were we assumed |N(a')| ≥ |N(a)| for the sake of simplicity of notation.

Each atom of the smaller of both molecules is assigned to exactly one atom of the larger molecule such that the overall similarity score is maximized. To prevent larger molecules automatically achieving a higher kernel value than smaller ones, kernel is normalized (e.g. Schoelkopf & Smola, 2002[21]), i.e.

$$k_{OA}(m,m') \leftarrow \frac{k_{OA}(m,m')}{\sqrt{k_{OA}(m,m)k_{OA}(m',m')}}, \qquad (10)$$

Nevertheless, after our examination of different molecules used in screening experiments, where the structure of the molecule is very complex, grouping of atoms, bonds proposed, based on OA kernel is not always relevant. In such cases we needed additional criteria to extend OA kernel also including molecule descriptors. As far as we know there is no extension of OA kernel where we should also consider molecule descriptors. Descriptors play an important role in the prediction of activity in screening data. One of the newest additions to this class of whole-molecule descriptors is the set of Burden metrics of Pearlman and Smith 1998 [18].

Let us assume now we have two molecules m and m', which have atoms $a_1,\ldots,a_{|m|}$ and $a'_1,\ldots,a'_{|m'|}$. Let us further assume we have a kernel K_{nei}, which compares a pair of atoms (a_h, $a'_{h'}$) from both molecules, including information on their neighborhoods, membership to certain structural elements and other characteristics. Then a valid kernel between m, m' is the optimal assignment kernel [9]:

$$K_{OA}(m,m') := \begin{cases} \max_\pi \sum_{h=1}^{|m|} K_{nei}(a_h, a'_{\pi(h)}) & if\ |m'| \geq |m| \\ \max_\pi \sum_{j=1}^{|m'|} K_{nei}(a_{\pi(h')}, a'_{h'}) & otherwise \end{cases}, \qquad (11)$$

where K_{nei} is calculated based on two kernels K_{atom} and K_{bond} which compare the atom and bond features, respectively. OA graph kernel, is of general use for QSAR problems, but is more focused on screening data. In developing a good quality QSAR model for these data we combined OA graph kernel and kernel where we considered Burden descriptors [18]. Now we will describe how to compute the kernel efficiently using Burden descriptors. Again, given two molecules m and m', the basic idea of our method is to construct a kernel k(**m, m')** which measures the similarity between **m** and **m'**. Gausian kernel was chosen because it readily produces a closed decision boundary, which is consistent with the method used to select the molecular descriptors.

$$k_{Gaus}(m,m') = \exp(-\gamma \mid m - m' \mid), \qquad (12)$$

Calculating the OA graph kernel and at same time the Gausian kernel with eight Burden features give us two ways of comparing molecules: looking at the chemical structure and looking for molecule activity. Having two values from these two kernels and making, at the same time an average, gives us

more precise molecule similarity information.

$$K_{scr}(m,m') = \frac{K_{Gaus}(m,m') + K_{OA}(m,m')}{2}, \quad (13)$$

If weights are considered, the estimated probability of tested point can be easily calculated by: Transform the kernel function $K_{scr}(m,m')$ into weights w.

$$w=K_{scr}(m,m'), \quad (14)$$

If weights are considered, the estimated probability of tested point can be easily calculated by:

$$\hat{P}_n(\omega_i \mid m;k) = \frac{p_n(m,\omega_i)\cdot K_{scr}(m,m_i)}{\sum_{i=1}^{M} p_n(m,\omega_i)\cdot \sum_{i=1}^{k} K_{scr}(m,m_i)}, \quad (15)$$

where K_{scr} is the kernel function between the compound x and its ith neighbor.

Choosing a single value of *k* for all data points may not be optimal as we claimed in previously. For example, if the density of points varies across the predictor space, a constant value of *k* implies neighborhoods of differing sizes. In the case of distances, which are defined as strictly positive values, of course only the positive domain of kernel *K* has to be used. In this sense the choice of the kernel is other parameter in our proposed technique.

In [12], the probability estimate $\hat{P}(m_i \mid \omega_M; k)$ at a test compound with descriptor values x_i takes a weighted average among the selected k nearest points. In this respect, we can automatically give the points close to the test compound more weight and the points far away less weight. This, in some sense, alleviates the bias drawn by including the points far away.

IV. Experimental evaluation

We experimentally evaluated the performance of weighted knn in a classification algorithm and compared it against that achieved by earlier approaches on a variety of chemical compound datasets.

A. Datasets

We used two different public available datasets to derive a total of eight different classification problems. The first dataset was obtained from the National Cancer Institute's DTP AIDS Anti-viral Screen program [4] [19]. Each compound in the dataset is evaluated for evidence of anti-HIV activity. The screen utilizes a soluble formazan assay to measure protection of human CEM cells from HIV-1 infection. Compounds able to provide at least 50% protection to the CEM cells were re-tested. Compounds that provided at least 50% protection on retest were listed as *moderately active* (CM, confirmed moderately active). Compounds that reproducibly provided 100% protection were listed as *confirmed active* (CA). Compounds neither active nor moderately active were listed as *confirmed inactive* (CI). We have formulated binary classification problems on this dataset, we consider only *confirmed active* (CA) and *confirmed inactive* (I) compounds and then build a classifier to separate these two compounds.

The second dataset was obtained from the Center of Computational Drug Discovery's anthrax project at the University of Oxford [10]. The goal of this project was to discover small molecules that would bind with the heptameric protective antigen component of the anthrax toxin, and prevent it from spreading its toxic effects. A library of small sized chemical compounds was screened to identify a set of chemical compounds that could bind to the anthrax toxin. The screening was done by computing the binding free energy for each compound using numerical simulations. The screen identified a set of 2,593 compounds that could potentially bind

TABLE I
RECOGNITION PERFORMANCE OF WEIGHTED KNN WITH GRAPH KERNEL BY VARYING THE NUMBER OF TRAINING SAMPLES (100, 200, 400) AND NEAREST NEIGHBORS K. NUMBERS REPRESENT CORRECT CLASSIFICATION RATE [%].

k	100		200		400	
	DTP AIDS	Toxic	DTP AIDS	Toxic	DTP AIDS	toxic
1	83.6%±1.05	81.5%±4.82	84.2%±2.05	83.1%±2.01	82.6%±3.02	82.5%±3.23
3	78.8%±5.68	73.8%±10.01	79.8%±2.12	74.9%±3.82	79.6%±3.12	73.9%±4.32
5	82.0%±7.01	81.3%±2.56	83.6%±0.98	81.5%±2.45	82.8%±1.52	81.0%±3.67
7	79.3%±4.81	73.9%±8.12	80.1%±2.43	75.6%±1.54	79.9%±2.78	74.1%±3.45

TABLE II
RECOGNITION PERFORMANCE COMPARISON OF WEIGHTED KNN WITH TRADITIONAL KNN, DANN AND ADAMENN FOR DIFFERENT VALUES OF K. NUMBERS REPRESENT CORRECT CLASSIFICATION RATE [%].

	weighted knn		DANN		ADAMENN	
	DTP AIDS	Toxic	DTP AIDS	Toxic	DTP AIDS	toxic
1	79.8%±2.54	74.9%±2.22	79.6%±4.04	73.9%±3.87	78.8%±5.34	73.8%±0.65
3	82.5%±3.67	80.3%±4.12	81.3%±2.54	79.8%±0.56	82.1%±3.87	79.2%±3.23
5	83.2%±3.54	81.5%±3.42	82.5%±2.75	81.3%±1.23	82.9%±3.98	80.3%±3.88
7	79.3%±4.81	73.9%±8.12	80.1%±2.43	75.6%±1.54	79.9%±2.78	74.1%±3.45

to the anthrax toxin and a set of 14,837 compounds that were unlikely to bind to the chemical compound. The average number of vertices in this dataset is 25 and the average number of edges is also 25. For these datasets we generated 8 features, called Burden descriptors [1]. In the case of 8 data properties, we can construct, and check our classifier in a maximum eight dimensional spaces.

B. Results

We tested a proposed extension of knn with molecule graph kernel in two benchmark experiments of chemical compound classification. In order to see the effect of generalization performance on the size of training data set and model complexity, experiments were carried out by varying the number of training samples (100, 200, 400) according to a 5-fold cross validation evaluation of the generalization error (Table 1).

The experimental results show that weighted knn has a better classification performance when the number of training samples is 200, while there is comparable performance when the number of samples is 400. Next we compared the classification accuracy of the proposed algorithm corresponding to the 1st- and 2nd datasets with other modifications of knn for different values of k neighbors. determine if a new weighted knn has an effect in classification in comparison to DANN and ADAMENN. A permutation test was selected as an alternative way to test for differences in compared algorithms in a nonparametric fashion (so we do not assume that the population has a normal distribution, or any particular distribution and, therefore, do not make distributional assumptions about the sampling distribution of the test statistics). The R package "exactRankTests" [11] was used for permutation test calculation. Table 3 lists the 2 calculation of accuracy with different k and results from the test. This Table shows four columns for each pair of compared different nearest neighbors methods (both data sets), the first and second giving the classification accuracy, while the last two columns have the raw (i.e., unadjusted) t-statistic result and p-values computed by the resampling algorithm already described [22]. The permutation test based on 2000 sample replacements estimated a p-value to decide whether or not to reject the null hypothesis. The null hypotheses for this test were $H0_1$: weighted knn =DANN, $H0_2$: weighted knn = ADAMENN and alternative hypothesis HA_1: weighted knn > DANN, HA_2: weighted knn > ADAMENN, additionally let's assume at a significance level $\alpha = 0.05$. The permutation test will reject the null hypothesis if the estimated P-value is less than α. More specifically, for any value of $\alpha <$ p-value, fail to

TABLE III
STATISTICAL TEST BETWEEN WEIGHTED KNN, DANN AND ADAMENN FOR DIFFERENT VALUES OF K NEAREST NEIGHBORS. NUMBERS REPRESENT CORRECT CLASSIFICATION RATE [%], T-STATISTIC (WITHOUT PERMUTATION) AND CALCULATED P-VALUE FROM PERMUTATION TEST. CALCULATED T*-STATISTIC TO THE NEW DATA SET WITH REPLACEMENT 2000 TIMES GAVE RESULT IN AVERAGE $T^*_{MIN} = 0.852$, $T^*_{MAX} = 1.254$

K	DTP AIDS				DTP AIDS			
	Weighted knn	DANN	t-stat	p-value	Weighted knn	ADAMENN	t-stat	P-value
1	79.8%±2.54	79.6%±4.04	3.78	0.0231	79.8%±2.54	78.8%±5.34	4.11	0.0012
3	82.5%±3.67	81.3%±2.54	3.22	0.0541	82.5%±3.67	82.1%±3.87	3.82	0.0012
5	84.2%±1.34	82.6%±1.06	-3.12	0.0773	84.2%±1.34	83.6%±2.04	2.77	0.0032
7	84.8%±0.92	83.9%±1.77	-2.89	0.0561	84.8%±0.92	82.1%±0.65	-.4.01	0.0044
			Average:	0.0527			Average	0.0025

K	DTP AIDS				DTP AIDS			
	weighted knn	DANN	t-stat	p-value	Weighted knn	ADAMENN	t-stat	P-value
1	74.9%±2.22	73.9%±3.87	4.02	0.0231	74.9%±2.22	73.8%±0.65	2.55	0.0002
3	80.3%±4.12	79.8%±0.56	4.56	0.0541	80.3%±4.12	79.2%±3.23	4.33	0.0023
5	83.1%±2.65	82.5%±1.88	-4.21	0.0773	83.1%±2.65	81.5%±3.98	3.53	0.0022
7	82.4%±3.54	81.9%±3.05	3.54	0.0561	82.4%±3.54	82.1%±.4.44	-2.22	0.0034
			Average	0.0426			Average	0.0020

Weighted knn shows performance improvements over the DANN and ADAMENN, for both of the (noisy) real screening data sets. Moreover it is worth mentioning that weighted knn does slightly better than DANN in general. Finally, weighted knn is significantly better than ADAMENN and k = 5 at a classification rate of 83% (Table 2). This is a very satisfying result as the definition of activity plays a very important role in modern biostatistics. We would like now to reject H0, and for any value of $\alpha \geq$ P-value, reject H0. The P-values (Table 3) on average for DTS AIDS and toxic data sets of 0.0527, 0.0025, 0.0426, 0.0020 indicates that the classification with weighted knn is probably not equal to DANN and ADAMENN. The P-value 0.0527 between weighted knn and DANN for DTS AIDS data sets, indicates weak evidence against the null hypothesis. There is strong evidence that all other tests null hypothesis can be rejected.

Permutation tests suggest, on average, that weighted knn for screening data is statistically significantly larger than DANN and ADAMENN.

V. Conclusion

We have proposed a new modification of knn algorithm that is suitable for discriminative classification with unordered sets of local features. Our weighted knn with graph kernel approximates the optimal partial matching of molecules by computing vertex labels, edge labels and burden values. The kernel is robust since it does not penalize the presence of extra features, respects the co-occurrence statistics inherent in the input sets, and is provably positive-definite. We have applied weighted knn to different datasets, and demonstrated recognition performance with accuracy comparable to current methods on screening data. Our experimental evaluation showed that our algorithm leads to substantially better results than those obtained by existing QSAR- and sub-structure-based methods.

References

[1] Burden, F.R. 1989. "Molecular Identification Number For Substructure Searches", Journal of Chemical Information and Computer Sciences, 29, 225-227.
[2] D. Hand, H. Mannila, P. Smyth.: Principles of Data Mining. The MIT Press. (2001)
[3] Domeniconi, C., Peng, J., Gunopulos, D.: Locally adaptive metric nearest-neighbor classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 1281–1285
[4] D.P. Mahe, N. Ueda, T. Akutsu, J.-L. Perret and J.-P. Vert, "Extensions of Marginalized Graph Kernels," Proc. 21st Int'l Conf. Machine Learning, 2004
[5] Friedman, J.: Flexible metric nearest neighbor classification. Technical Report 113, Stanford University Statistics Department (1994)
[6] Graham W. Richards. Virtual screening using grid computing: the screensaver project. Nature Reviews: Drug Discovery, 1:551–554, July 2002.
[7] Gregory A Landrum, Julie E Penzotti and Santosh Putta, Machine-learning models for combinatorial catalyst discovery. Rational Discovery LLC, 555 Bryant St 467, Palo Alto, CA 94301, USA
[8] Hastie, T., Tibshirani, R.: Discriminant adaptive nearest neighbor classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (1996) 607– 615
[9] H. Froehlich, J. K. Wegner, A. Zell, QSAR Comb. Sci. 2004, 23, 311 – 318.
[10] Hawkins, D.M., Young, S.S., and Rusinko, A. 1997. "Analysis of a Large Structure- Activity Data Set Using Recursive Partitioning", Quantitative Structure Activity Relationships 16, 296-302.
[11] http://cran.r-project.org/src/contrib/Descriptions/exactRankTests.html. "exactRankTests": Exact Distributions for Rank and Permutation Tests
[12] J. Kandola, J. Shawe-Taylor, and N. Cristianini. On the application of diffusion kernel to text data. Technical report, Neurocolt, 2002. NeuroCOLT Technical Report NC-TR-02- 122.
[13] Joachims T., 1998. Text Categorization with Support Vector Machines: Learning with Many Relevant Features [A]. In: Proceedings of the European Conference on Machine Learning [C].
[14] J.P. Myles and D.J. Hand, "The Multi-Class Metric Problem in Nearestneighbor Discrimination Rules," Pattern Recognition, vol. 723, pp. 1291-1297, 1990.
[15] J. Peng, D. Heisterkamp, and H.K. Dai, "LDA/SVM Driven Nearest Neighbor Classification," Proc. IEEE Computer Soc. Conf. Computer Vision and Pattern Recognition, pp. 58-63, 2001.
[16] Klopman, G. 1984. "Artificial Intelligence Approach to Structure-Activity Studies. Computer Automated Structure Evaluation of Biological Activity of Organic Molecules", American Chemical Society, Vol. 106, No. 24, 7315-7321.
[17] Li Baoli, Chen Yuzhong, and Yu Shiwen, 2002. A Comparative Study on Automatic Categorization Methods for Chinese Search Engine [A]. In: Proceedings of the Eighth Joint International Computer Conference [C]. Hangzhou: Zhejiang University Press, 117-120.
[18] Pearlman, R. S. and Smith, K. M. 1998. "Novel software tools for chemical diversity", Perspectives in Drug Discovery and Design, 9/10/11, 339-353.
[19] S. Kramer, L. De Raedt, and C. Helma. Molecular feature mining in hiv data. In 7th International Conference on Knowledge Discovery and Data Mining, 2001.
[20] R.D. Short and K. Fukunaga, "Optimal Distance Measure for Nearest Neighbor Classification," IEEE Trans. Information Theory, vol. 27, pp. 622-627, 1981.
[21] Yang Y. and Liu X., 1999. A Re-examination of Text Categorization Methods [A]. In: Proceedings of 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval [C]. 42-49.
[22] Westfall, P. H. & Young, S. S. (1993). Resampling-based multiple testing: Examples and methods for p-value adjustment, John Wiley & Sons.
[23] W. Hechenbichler, K., Schliep, K.: Weighted k-Nearest-Neighbor Techniques and Ordinal Classification. SFB Discussion paper 399. (2004)