Machine learning in molecular classification

Hongyu Su

Department of Computer Science
PO Box 68, FI-00014 University of Helsinki, Finland
firstname.lastname@cs.helsinki.fi

Abstract. Chemical and biological research is facilitated by big data repositories of chemical compounds from ultra-high-throughput screening techniques, where large numbers of molecules are tested and classified based on their activities against given target. With the increasing availability of data, it is quite necessary to develop accurate and robust models to predict chemical and biological properties of novel molecules based on their structural representations of different dimensions. In this report, we will briefly review literatures of several machine learning approaches on molecular classification problem. These approaches include mining statistically significant molecular substructures, graph walk kernel, and molecular fingerprints. We will focus on recently developed methodology of mining statistically significant molecular substructures. Then, we will propose our novel method which we call atom properties enrichment kernel. The new method can incorporate huge amount of atom level properties. Finally, we will give experimental results of these methods by applying them on standard datasets.

1 Introduction

Chemical and biological research is facilitated by big data repositories of chemical compounds from ultra-high-throughput screen techniques, where large number of molecules are tested and classified based on their activities against given targets. The increasing availabilities of data offer both challenges and opportunities to machine learning, where the goal is to infer chemical or biological properties of molecules based on structural representations in different dimensions. The prediction is based on the assumption that molecules with similar geometric or physiologic properties will also have similar chemical and biological characters, which is also the basis for early studies [1] of quantitative structure activity relationship (QSAR).

Molecules are represented by strings, or more commonly, graphs in 2D or 3D space where atoms are represented by nodes and bonds are represented by edges. Therefore, a prediction model should solve the problem of generating features from a molecule and construct a learning model effectively. There are two mainstream methods. One direction is to generate feature representations of molecules through "molecular fingerprints", which maps the molecule into a fixed width bit vector. Each position in the vector denotes the presence or

absence of certain substructure or fragment in the molecule. Another way is to represent a molecule by a set of statistically significant substructures which was developed in literature [2].

Support vector machine and kernel methods [3] emerged during the last decade and became important prediction methods in machine learning. They are suitable for structured data of various kinds. The advantage of these methods is that the learning machine does not have to access high dimensional feature space when it interacts with samples in the dataset. The challenges for kernel methods is to construct or learn a good kernel for a given task. In molecule classification problem, these kernels are functions capable of calculating similarity between molecules based on different molecule representations. One of the first successful applications of kernel methods in molecule classification is often known as walk kernel [4]. Later, a fast method for calculating walk kernel with dynamic programming algorithm was introduced [5]. Besides walk kernel, marginalized graph kernel, which used random walks on graphs was developed in the literature [6]. It was further extended by adding morgan indices on node labels and preventing nodes being revisited during generating walks [7]. After that, a number of kernel methods was developed and applied in this area, including kernels based on fingerprints [8] and kernels considering different string representations [9], local substructures [10] or three-dimensional substructures [11]. It is worth mentioning that subgraph kernels were proved hard to calculate [12], as they are NP-hard graph isomorphism problems in nature. However, subgraph kernels, which are restricted to several salient subgraphs and applied to moderate-sized datasets, can still product satisfactory results. These methods are usually known as reduced graph approaches [13].

Even though a number of methods have already been developed for molecule classification, new methods in this area are bing developed. The reason is that no single method can adequately capture all properties of molecules and can be universally applied to each specific area. Therefore, in the following sections, we will first review a substructure mining algorithm developed [2], and a graph walk kernel method [4]. We will also define molecular fingerprints approaches in a detailed manner. Then, we will describe our new method, which is a kernel method that can combine chemical properties in atom level into molecular representations. We will also show experimental results for the quality measurements of these methods.

2 Related work

In this section, we will briefly review several popular feature mining algorithms that are widely used in molecular classification. These algorithm includes GraphSig (an statistical method that aims to find statistically significant subgraph patterns), graph walk kernel (kernel methods using similar path to measure pairwise similarities), molecular fingerprints methods(feature representations of molecules).

2.1 Mining statistically significant subgraphs

GraphSig [2] is an algorithm for mining statistically significant subgraphs for efficient molecular classification. The algorithm aims to mine molecular substructures, which have p-value below a predefined threshold, from the entire dataset. After that, the algorithm represents each molecule by a feature vector that takes the statistically significant substructures as features. Then it employs support vector machine (SVM) for classification.

Incorporating domain knowledge on molecular graphs. Two different approaches are applied in the model to combine domain knowledge in molecular graphs. The first one is to enhance atom labels in graphs. Atoms in molecule graph are originally labeled by a simple atom type. For example a carbon atom is labeled by C and an oxygen atom is labeled by O. When atom labels are enhanced, they will also incorporate part of structural information, as shown in Figure 1. For example a carbon atom in an aromatic ring structure will be labeled by $C.ar$ and an oxygen atom connected by a double chemical bond will be labeled by $O.2$.

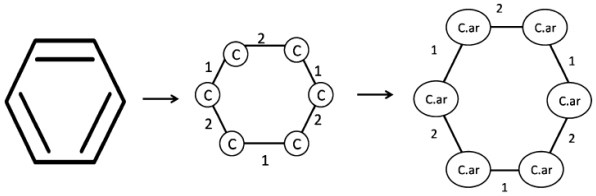


Fig. 1. [2] Enhanced atom labels.

Second, molecular graph will no longer be represented only in atom level. GraphSig detects functional groups in each molecule and replaces atoms that are part of functional groups by names of functional groups. for example, six carbon atoms in an aromatic ring structure will be replaced by a special node called "benzene", as shown in Figure 2.

Representing molecules as sets of histograms. Once domain knowledge is combined into molecular graph, it is necessary to represent molecule into a set of histograms. This is because GraphSig builds probability framework base on the histogram representations. GraphSig performed random walk with restart (RWR) from each nodes of a molecular graph to convert molecule to a set of histograms. The idea of RWR is to capture the distribution of node-node pairs in each molecule.

RWR simulates a random walker that starts from a target node and keeps jumping from one node in graph to its neighbors. Each neighbor of the current

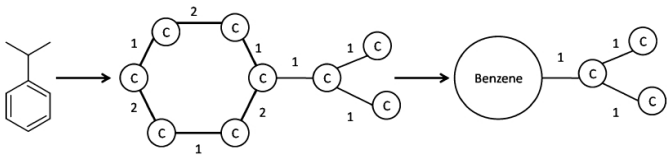
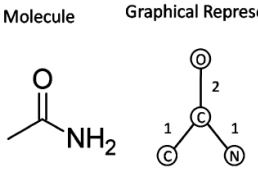


Fig. 2. [2] Extract benzene function group.

node has equal probability to be reached in the next step. The walker has a restart probability of α, because we do not want the walker go too far from the starting point. In GraphSig, restart probability α is set to 0.25, which means the walker restarts after four jumps on average. Perform RWR on each node of a molecular graph repeatedly until the distribution of node-node pairs converges. Then the distribution of node-node pairs is normalized and discretized. Figure 3 shows the results from RWR on carboxamide.



ID	Starting Atom	O-2-C	C-1-C	C-1-N
h_1	O	4	2	2
h2	C	2	3	3
h3	C	2	4	2
h4	N	2	2	4

Fig. 3. [2] RWR results on carboxamide. In its graph representation, hydrogen atoms are omitted.

Probabilistic framework. Suppose we have four sample graphs G_1, G_2, G_3, and G_4 as shown in Figure 4. The histograms of node a from RWR are shown in Table 5. The results of the histogram reveal there could be one common subgraph among G_1, G_2 and G_3 which is formed by node-node pairs $a-b$, $b-c$ and $b-d$. Therefore, we can calculate the significant of the substructures if we are able to calculate the p-value of the histogram.

Suppose the database has only one molecule, which is carboxamide as shown in Figure 3. Histograms of different starting nodes can be calculated from a RWR procedure, as shown in the table of Figure 3. The histograms denote the distributions of all node-node pairs from different starting nodes in the database. A prior probability matrix, as shown in Finger **??**, can be obtained by analyzing the histogram matrix. The histogram matrix denotes the probability of finding a node-node pair with a frequency over n. for example, third column of the second

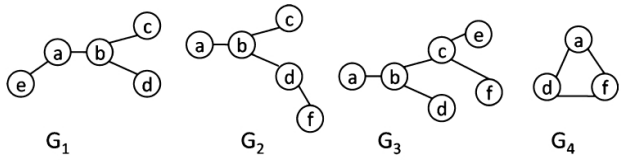


Fig. 4. [2] Sample graph database.

histogram	a−b	a−d	a−e	a−f	b−c	b−d	c−e	c−f	d−f
G_1	2	0	3	0	1	1	0	0	0
G_2	4	0	0	0	2	1	0	0	1
G_3	3	0	0	0	1	2	1	1	0
G_4	0	3	0	3	0	0	0	0	2

Fig. 5. [2] Histograms of node a in sample graphs of Figure 4.

row represents the probability that finding the C-1-C with frequency over 3 in table of Figure 3 is 2/4.

NNPs	1	2	3	4	5
O-2-C	4/4	4/4	1/4	1/4	0
C-1-C	4/4	4/4	2/4	1/4	0
C-1-N	4/4	4/4	1/4	1/4	0

Fig. 6. [2] Prior probability matrix.

Therefore, the probability of finding a histogram $X = [x_1 = f_1, x_2 = f_2, ..., x_n = f_n]$ in a database with a fixed prior probability matrix M_{prior} is modeled by a joint probability of finding each component node-node pair x_i with a frequency over f_i in the prior probability matrix, defined as:

$$p(X) = M_{prior}(x_1, f_1) * M_{prior}(x_2, f_2) * ... * M_{prior}(x_n, f_n)$$

for example, finding histogram h_1 in table of Figure 3 is calculated as:

$$p(h_1) = M_{prior}(O-2-O, 4) * M_{prior}(C-1-C, 2) * M_{prior}(C-1-N, 2) = 1/4$$

***p*-value of the histogram.** Given histogram of a database, the probability of finding X in a database can be calculated by $p(X)$, as described in the previous section. Suppose the number of histogram in the database is m, finding a specific

histogram X in the database with frequency μ can be modeled by a binomial distribution:

$$P(X;\mu) = \binom{m}{\mu} p(X)^{\mu}(1-p(X))^{m-\mu}$$

Therefore, the p-value of finding a histogram X with a given frequency μ_0 can be calculated by measuring the area under probability distribution function (PDF) of $p(X)$, defined as:

$$p\text{-}value(X;\mu_0) = \sum_{i\in[\mu_0,m]} P(X;i)$$

statistically significant histograms. Given such a probability framework, p-value threshold and μ frequency threshold (the expectation of the number of certain histogram in dataset), GraphSig uses GraphRank [14] to generate all histograms that satisfies the thresholds in the database. Each histogram from GraphRank will represent a significant substructure that can be used as a feature to represent the molecule.

2.2 Walk kernel

Another approach for molecular classification problem is by kernel methods. Kernel methods are different from using feature representations of molecules which tries to map a molecule into a feature vector. It actually measures the similarity between each pair of molecular graphs and organizes these similarity values into a matrix form. The kernel matrix can be used by support vector machine directly.

Walk kernel is one of the kernel methods that is based on measuring walks in two graphs that have same labels. The calculation is based on direct product graph and calculating the multiplication of adjacent matrix.

Graph terminology. An undirected graph $G = (V, E)$ is defined by a finite set of nodes $V = \{v_1, v_2, ..., v_n\}$ and a finite set of edges $E = \{e_1, e_2, ..., e_n\}$. The $n \times n$ adjacency matrix E of graph G is defined such that the (i, j) entry of E equal to 1 if and only if there is an edge between v_i and v_j. A walk length of m in the graph G is defined as $w = \{v_1, v_2, ..., v_m\}$ so that for $i = 1, 2, ..., m-1$ there exists an edge $(v_i, v_{i+1}) \in E$.

Walk kernel definition. A product graph between two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is denoted by $G_\times(G_1, G_2) = G_1 \times G_2$. The nodes and edges of the product graph $G_\times(G_1, G_2)$ is defined as:

$$V_\times(G_1, G_2) = \{(v_1, v_2) \in V_1 \times V_2, label(v_1) = label(v_2)\}$$
$$E_\times(G_1, G_2) = \{((v_1, v_2), (u_1, u_2)) \in V_\times \times V_\times, (v_1, u_1) \in E_1 \wedge (v_2, u_2) \in E_2\}$$

Walk kernel between two graphs denoted by $K_{wk}(G_1, G_2)$ is defined as the number of same walks up to infinite length shared by two graphs [4]. The contribution of single walk is also downscaled according to its length. Walk kernel can be easily calculated from the adjacency matrix of product graph obtained from two original graphs:

$$K_{wk}(G_1, G_2) = \sum^{|v\times|} [\sum_{n=0}^{\infty} \lambda^n E_{\times}^n],$$

where $v\times$ is the node in product graph and λ is the positive downscaling factor which is strictly less than 1.
Walk kernel using walks up to moderate length can mimic the one with infinite length of walks [4]. Therefore, dynamic programming algorithm was employed for fast calculation of walk kernel with finite walks. A walk kernel using walk up to length of p is defined as

$$K_{wk}(G_1, G_2) = \sum_{v_{\times i} \in V_{\times}} D_p(v_{\times i}),$$

where $D_p(v_i)$ is calculated by

$$D_0(v_i) = 1$$

$$D_n(v_i) = \sum_{(v_{\times i}, v_{\times j}) \in E_{\times}} D_{n-1}(v_{\times j})$$

2.3 Molecular fingerprints

Besides kernel methods and methods which aims to mine statistically significant part subgraphs, there is another method exists and widely used. The method is known as molecular fingerprints. Molecular fingerprints are designed to encode a molecular structure in a fixed width binary bit vector, which represents the present or absence of particular substructures or fragments in the molecule. As a result, molecular fingerprints are capable of encoding large number of features in a compact manner, and are extensively used for various tasks in chemical informatics, with the hope that two similar molecules create similar fingerprints and similar fingerprints mean the molecules are similar. There are two variations.

Hash fingerprints. One kind of fingerprints is commonly known as "hash fingerprints", which enumerates all linear fragments of length n in the molecule. Parameter n is usually bounded from three to seven. A hash function assigns each of the fragments a hash value, which determines its position in descriptor space.

Substructure keys. Another major fingerprint type is called "substructure keys", which is based on pattern matching of a molecular structure to a set of pre-defined substructures. Each substructure becomes a key and have a fixed position in descriptor space. These substructures are considered to be independent functional units and are identified by domain experts as prior knowledge.

Tanimoto coefficient. Once the molecules have been represented as fingerprints, Tanimoto coefficient is usually employed to measure the similarity between pair of molecules. Given two molecular fingerprints fp_1 and fp_2, Tanimoto coefficient is defined as

$$T(fp_1, fp_2) = \frac{N_{fp_1,fp_2}}{N_{fp_1} + N_{fp_2} - N_{fp_1,fp_2}},$$

where N_{fp_1} is the number of bits being "ON" in fingerprint fp_1, N_{fp_2} is the number of bits being "ON" in fingerprint fp_2, and N_{fp_1,fp_2} is the number of bits being "ON" in both of the fingerprints.

3 Feature enrichment method

In this section we describe a novel method which we call atom properties enrichment kernel. It uses detailed atom level properties for substructure comparisons. Follow the idea of molecule fingerprints, it is possible to assign a bit vector to each atom in the molecule. The positions of the vector represent properties of this atom. Possible properties include chemical properties, electric ones and geometric ones as shown in Table 1. We call this bit vector atom fingerprints, which represents pharmacophore properties of atoms. Therefore, atoms can be further specified in graph kernel methods. The comparison between each pair of atoms will return a similarity value based on their fingerprints, other than binary 0 or 1. For example, given two atom v_1 and v_2 together with their atom level fingerprints fp_1, fp_2, the similarity between this pair of atoms can be defined as

$$\sigma(v_i, v_j) = \begin{cases} 0 & \text{if } v_i \neq v_j \\ \frac{\gamma(fp_1,fp_2)}{\gamma(fp_1,fp_1)+\gamma(fp_2,fp_2)-\gamma(fp_1,fp2)} & \text{if } v_i = v_j \end{cases},$$

where γ function measures the number of same positions in two vectors.

Bit positions	Set when
1	Atom is acceptor
2	Atom is donor
3	Atom is negative
4	Atom is positive
5	Atom in ring
6	Atom is terminal carbon
...	...

Table 1. Atom fingerprint

Decomposition kernel [10] uses similarities of substructure pairs to calculate the similarity between two molecules. When comparing two substructures, decomposition kernel actually compares the spectrum of the selector atoms, which contain atom type, bond type and atom charge information of surrounding atoms. However, surrounding atoms in different radius are mixed together in the spectrum. It is necessary also to consider neighboring atoms on different radius. Therefore, an atom properties enrichment kernel, which derives from decomposition kernel by adding atom properties and considering atom in different radius, is defined as:

$$K(G_1, G_2) = \sum_{v_i, s_i \in R(v_i, s_i, G_1) v_j, s_j \in R(v_i, s_i, G_2)} \sigma(v_i, v_j) K_s(s_i, s_j)$$

$$K_s(s_i, s_j) = \sum_{r \in R} K_r(s_i, s_j)$$

$$K_r(s_i, s_j) = \sum_{v_m \in v_{ir} v_n \in v_{jr}} \sigma(v_m, v_n)$$

$R(v_i, s_i, G)$ is one substructure decomposition of molecule graph G, the selector node of substructure s_i is v_i. Function $K_s(s_i, s_j)$ returns similarity value between two substructures s_i and s_j. Function $K_r(s_i, s_j)$ returns similarity value between two substructures s_i and s_j according to radius r neighboring atoms. v_{ir} and v_{jr} are atoms in radius r of two subgraphs s_i and s_j.

4 Data

We apply methodologies described in the previous sections to the problem of predicting mutagenicities and anti-cancer activities based on two public available standard dataset respectively (Mutag and NCI-cancer).

4.1 Mutag dataset

Mutag dataset [15] was constructed based on data from review of literatures about mutagenicities in *Salmonella Typhimurium* based on 200 aromatic and heteroaromatic nitro compounds. The determinants for mutagenecity in the study are hydrophobicity and energies of the lowest unoccupied molecular orbitals. As a result, 188 congeners were extracted together with their structure-activity relationship (SAR) data. Therefore, Mutag dataset is very suitable for machine learning and is widely used as one of the standard dataset. The measures of success of various learning methods on Mutag dataset are usually reported as accuracies from a leave-one-out(LOO) cross validation procedure and are compared with each other.

More specifically, 125 molecules in Mutag dataset are labeled as positive and 63 are labeled as negative. Distribution of number of atoms in molecules is shown in Figure 7. Distribution of number of chemical bonds in molecules is shown in Figure 8.

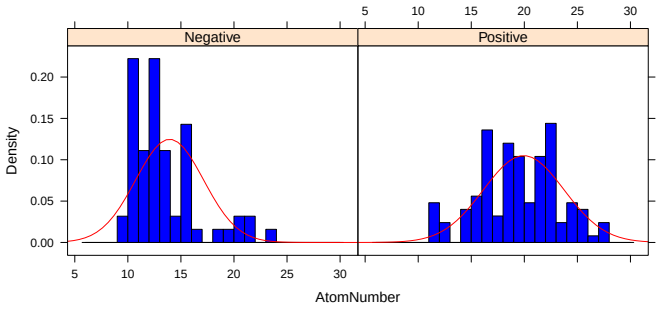


Fig. 7. Distributions of number of atoms in molecules in Mutag dataset

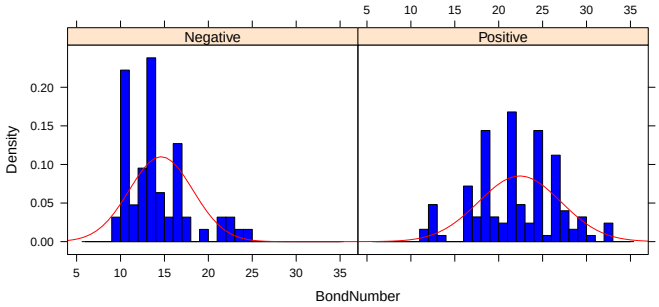


Fig. 8. Distributions of number of chemical bonds in molecules in Mutag dataset

4.2 NCI cancer dataset

Developmental Therapeutics Program[1] (DTP) from National Cancer Institute and National Institutes of Health (NCI/NIH) was designed to screen up to 3,000 compounds every year searching for potential anti-cancer drugs. This program utilizes compound data in several human cancer cell lines including leukemia, melanoma and cancers of the lung, colon, brain, ovary, breast, prostate, and kidney. The data for each cell line can be obtained through PubChem Bioassay[2] [16] data repository. For each tested molecule in certain cell line, PubChem [17] provides its bioactivity score and bioactivity outcome. Bioactivity score provides relative ranking score of the molecular activity. Bioactivity outcome categorizes bioactivity scores and includes five classes, namely chemical prob, active, inactive, inconclusive, and unspecified.

[1] http://dtp.nci.nih.gov/index.html
[2] http://pubchem.ncbi.nlm.nih.gov

Currently, there are 43,884 compounds in the PubChem Bioassay database together with anti-cancer activities in 73 cell lines. 59 cell lines have screen experimental results for most compounds, therefore are suitable for machine learning algorithms. The number of active compounds and total number of compounds in each cell line is reported in Table 2 and Table 3. Since NCI-cancer datasets are relatively big, performances of algorithms are usually analyzed by n-fold cross validation methods.

NCI-cancer datasets were first employed in literature [9] of early study in predicting mutagenecity, toxicity and anti-cancer activity by sampling equivalent number of active molecules and inactive ones from original datasets. The datasets resulted from sampling were mostly used in researches that followed. However, we found the sampled datasets were erroneous and lots of molecules were mislabeled compared to PubChem Bioassay database. Therefore, we use data from PubChem Bioassay database directly in our experiments.

AIDs	***Cell line names***	***Actives***	***Total***	***AIDs***	***Cell line names***	***Actives***	***Total***
1	NCI-H23	2052	40560	3	NCI-H226	1811	37365
5	NCI-H322M	1533	39213	7	NCI-H460	2351	39224
9	HOP-62	1862	39283	11	HOP-18	594	10999
13	HOP-92	1933	35764	15	NCI-H522	2707	36809
17	LXFL	720	13300	19	A549/ATCC	1961	40785
21	EKVX	1444	39570	23	LOX	2481	37850
25	M14	1940	39821	27	M19-MEL	789	14450
29	MALME-3M	1959	37933	31	UACC-62	2147	39636
33	UACC-257	1646	40244	35	SK-MEL-2	1600	37926
37	SK-MEL-5	2200	39585	39	SK-MEL-28	1448	39937
41	PC-3	1571	27596	43	DU-145	1488	27511
45	SF-268	2005	40113	47	SF-295	2023	40482
49	SF-539	2067	37933	51	XF	771	11853
53	SNB-19	1686	39999	55	SNB-75	2047	37810
57	SNB-78	580	13360	59	U251	2166	40482
61	DMS	888	13100	63	DMS	931	14174
65	HT29	2108	40408	67	COLO	2120	40124
69	DLD-1	766	13925	71	HCT-15	2099	40117
73	KM12	2000	40261	75	KM20L2	683	13608

Table 2. Information of NCI-cancer datasets

AIDs	*Cell line names*	*Actives*	*Total*	*AIDs*	*Cell line names*	*Actives*	*Total*
77	HCC-2998	1802	36296	79	HCT-116	2473	40194
81	SW-620	2405	40733	83	MCF7	2292	28003
85	MDA-MB-435	1810	27669	87	MDA-N	1757	27083
89	BT-549	1243	24766	91	T-47D	1508	26113
93	NCI/ADR-RES	1428	27950	95	MDA-MB-231/ATCC	1411	27180
97	HS	1417	26059	99	OVCAR-3	2100	39315
101	IGROV1	2034	40153	103	SK-OV-3	1524	38427
105	OVCAR-4	1523	38587	107	OVCAR-5	1301	39512
109	OVCAR-8	2077	40724	111	P388	331	980
113	RPMI-8226	2697	37663	115	SR	3267	33647
117	P388/ADR	249	959	119	CCRF-CEM	3526	38895
121	K-562	2899	39790	123	MOLT-4	3130	40189
125	HL-60(TB)	3331	37125	127	SN12K1	251	995
129	A498	1620	34968	131	CAKI-1	2053	37869
133	RXF	2012	36135	135	RXF-631	496	10746
137	786-0	2093	39667	139	ACHN	2016	39918
141	TK-10	1278	39346	143	UO-31	1875	39892
145	SN12C	1953	40201				

Table 3. Information of NCI-cancer datasets

5 Experimental results

5.1 Measurements of success

One way to measure the quality of classification is by accuracy (ACC), which is the proportion of true results, including *true positive* (TP) and *true negative* (TN), in the entire population. As we sometimes come up with biased data and accuracy is affected by the number of positive labeled examples in the dataset, it does not always tell the true stories of the classification quality. Therefore, we also measure the quality by calculating the area under the receiver operator characteristics (ROC) curve, known as AUC. ROC curve can be represented as the ratio between *true positive rate* (TPR) and *false positive rate* (FPR). The AUC of an ideal classifier is 1, and an random classifier should have AUC around 0.5.

For NCI-cancer dataset, both accuracy and AUC score are reported from a 5-fold cross-validation method. Given a training set $Z = \{x_i, y_i\}_{i=1}^n$, the k-fold cross-validation is calculated as follow: first of all, the training set Z is randomly divided into k subset $Z_1, Z_2, \ldots, Z_k$ of approximatly equal size. Then SVM is trained based on $k-1$ subsets, and error rate is estimated on the remaining subset. The process is repeated k times, such that each subset is tested once. The overall error rate from cross-validation is the average error rate from each testing. For Mutag dataset, a leave-one-out procedure is employed. Leave-one-out method is equivalent to n-fold cross-validation, which divides training set Z into n subsets and uses each subset for testing.

5.2 Results

We use following notations for different methods: SK for molecular fingerprints of substructure keys, HF for molecular fingerprints of hash fragments, WK for graph walk kernel method, FE for atom properties enrichment kernel, and GS for GraphSig. These notations are listed in Table 4. Substructure keys and hash fragments features are generated by OpenBabel [3]. Support vector machine used in the experiments is LibSVM [4]. To obtain a fare comparisons of these methods, SVM C parameter is fixed to 1 for all the experiments.

Notations	***Methods***
SK	Substructure keys
HF	Hash fingerprints
WK	Graph walk kernel
FE	Feature Enrichment kernel
GS	GraphSig

Table 4. Notations used for different methods.

Mutag dataset. For Mutag dataset, the prediction accuracies of all methods are above 80%, which means it is a well-defined dataset and the prediction task is relatively easy. The best accuracy and AUC score are achieved by the atom properties enrichment kernel, as shown in Table 5. GraphSig did not use Mutag dataset. Besides, it uses different file format to represent graph, which makes it difficult to use it on new datasets. Therefore we do not have results available from GraphSig here.

Methods	***Accuracies***	***AUCs***
SK	84.57%	86.56%
HF	86.17%	89.92%
WK	81.38%	88.33%
FE	**90.02%**	**95.45%**

Table 5. Prediction accuracies and AUC scores in Mutag dataset

[3] http://openbabel.org
[4] http://www.csie.ntu.edu.tw/ cjlin/libsvm/

NCI-cancer dataset. For NCI-cancer datasets, we again carried out the experiments by using substructure keys, hash fragments, graph walk kernel and our atom properties enrichment kernel. Totally, 59 datasets were used based on the criterion that there was no missing screen data for molecules. The behavior of different methods are shown in Figure 9. The results show that these three methods have similar behaviors on NCI-cancer dataset. Besides, one can infer that hash fragments features worked better than the others. Graph walk kernel and substructure keys features are almost equivalent methods on NCI-cancer datasets. Atom properties enrichment kernel did not work as well as other methods.

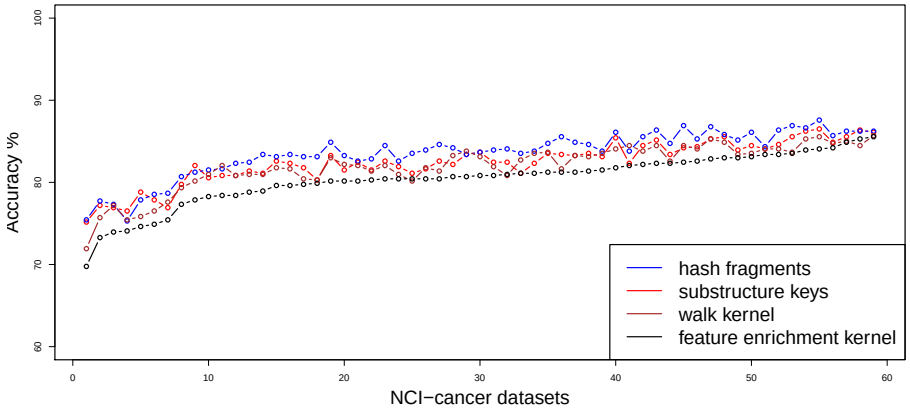


Fig. 9. Prediction accuracies on NCI-cancer dataset.

GraphSig sampled molecules from original database and there is no information about the real data employed [2]. Besides, in GraphSig accuracy and AUC score were not calculated directly from SVM but from other evaluation software. Therefore in our experiments, the results from GraphSig and other methods are not directly comparable.

To make GraphSig comparable to other methods, we carried out experiments on the sample dataset in GraphSig package. By using GraphSig and the evaluation software, we achieved an AUC score of 91.66% which coincided with the AUC scores from the experiments on NCI-cancer in the literature [2]. We then used GraphSig and LibSVM package for evaluation, giving an accuracy of 84.48%. This means GraphSig should have similar behavior with other methods.

Performance differences. As illustrated in the results, feature enrichment kernel has good performances in Mutag dataset. However, it does not work well

on NCI-cancer datasets. The differences in performances may be caused by the differences in properties of molecules shows in two kinds of datasets. Mutag is about the activity data of molecules causing gene mutations, and NCI-cancer is about the activity data of molecules that kill cancer cells. Needless to say, different aspects of activities are represented by different molecular features. The feature enrichment kernel may well capture the molecular features which can explain gene mutations. However, the kernel can not represent those features that kill cancer cells. That's why the performance differences are so different. Besides, the mechanism of gene mutations seem to be simpler than one that kills cancer cells. This may be the reason that algorithm always work well on Mutag rather than NCI-cancer.

6 Conclusions

In the report, we first reviewed several current popular approaches for molecular classification problem. These approaches include GraphSig which is a newly developed algorithm for mining statistically significant subgraphs in a graph dataset, graph kernel methods and molecular fingerprints methods. After analyzing these methods, we then proposed our novel methods which is atom properties enrichment kernel that is able to combine detailed chemical, electric and geometric properties of atom level into graph kernels. Experimental results were also available in the report in order to measure the performances of these methods.

GraphSig is a newly developed method that aims to mine statistically significant substructures from molecular graphs. It also combines domain knowledge into molecular graph representations, such as detailed atom labels and functional group extractions. Besides, it offers good classification accuracies. In a sense, GraphSig is one of the the-state-of-the-art methods in molecular classification problem. However, the results of the experiments show that GraphSig is not reasoning as good as described in [2]. Other methods can offer competing results as well, which can be seen from last section.

Molecular fingerprints methods including substructure keys and hash fragments also make good predictions. Besides, they are computational more efficient than GraphSig. They also combine prior knowledge in chemical domain about functional groups in molecules. However, there is no feature selection in these methods, which means some features are irrelevant and some are redundant.

Kernel methods derived from support vector machine are flexible for classification tasks. Most of them are also efficient to calculate. Graph walk kernel was proved in this article to have a good performance in molecular classification. The atom properties enrichment kernel, which combines chemical properties in atom level, achieves the best result in Mutag dataset. The performances of atom properties enrichment kernel on NCI-cancer datasets are also acceptable.

Though lots of methods exist and have good performances on molecular classification tasks, new methods are still actively pursued. This is because no single method can collect all information of molecules. In the future, we will

mainly focus on combine other domain knowledge in representing molecules, for example from chemical and biological areas.

References

1. Hansch, C., Maloney, P.P., Fujita, T., Muir, R.M.: Correlation of biological activity of phenoxyacetic acids with hammett substituent constants and partition coefficients. Nature **194** (1964) 178–180
2. Ranu, S., Singh, A.: Mining statistically significant molecular substructures for efficient molecular classification. Chemical Information and Modeling **49**(11) (November 2009) 2537–2550
3. Taylor, J.S., Cristianin, N., eds.: Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press (2000)
4. Gärtner, T.: A survey of kernels for structured data. SIGKDD Explor. Newsl. **5**(1) (2003) 49–58
5. Saunders, C., Demco, A., Pitkänen, E., Rousu, J.: Molecule graph and reaction kernels for drug discovery and system biology
6. Kashima, H., Tsuda, K., Inokuchi, A.: Marginalized kernels between labeled graphs. In: Proceedings of the Twentieth International Conference on Machine Learning, AAAI Press (2003) 321–328
7. Mahé, P., Ueda, N., Akutsu, T., Perret, J.L., Vert, J.P.: Extensions of marginalized graph kernels. In: ICML '04: Proceedings of the twenty-first international conference on Machine learning, New York, NY, USA, ACM (2004) 70
8. Ralaivola, L., Swamidass, S.J., Saigo, H., Baldi, P.: 2005 speical issue: Graph kernels for chemical informatics. Neural Netw. **18**(8) (2005) 1093–1110
9. Swamidass, S.J., Chen, J., Bruand, J., Phung, P., Ralaivola, L., Baldi, P.: Kernels for small molecules and the prediction of mutagenicity, toxicity and anti-cancer activity. Bioinformatics **21** (June 2005) i359–i368
10. Menchetti, S., Costa, F., Frasconi, P.: Weighted decomposition kernels. In: ICML '05: Proceedings of the 22nd international conference on Machine learning, New York, NY, USA, ACM (2005) 585–592
11. Ceroni, A., Costa, F., Frasconi, P.: Classification of small molecule by two- and three-dimensional decomposition kernels. Bioinformatics **23** (May 2007) 2038–2045
12. Gärtner, T., Flach, P., Wrobel, S.: On graph kernels: Hardness results and efficient alternatives. In: Proceedings of the 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop, Springer-Verlag (August 2003) 129–143
13. Harper, G., Bravi, G.S., Pickett, S.D., Green, D.V.S.: The reduced graph descriptor in virtual screening and data-driven clustering of high-throughput screening data. Chem. inf. Compute Sci. **44** (2004) 2145–2156
14. He, H., Singh, A.K.: Graphrank: Statistical modeling and mining of significant subgraphs in the feature space. In: ICDM '06: Proceedings of the Sixth International Conference on Data Mining, Washington, DC, USA, IEEE Computer Society (2006) 885–890
15. Debnath, A.K., de Compardre, R.L., Debnath, G., Shusterman, A.J., Hansch, C.: Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. Medicinal Chemistry **34**(2) (July 1991) 786–797

16. Wang, Y., Bolton, E., Dracheva, S., Karapetyan, K., Shoemaker, B.A., Suzek, T.O., Wang, J., Xiao, J., Zhang, J., Bryant, S.H.: An overview of the pubchem bioassay resource. Nucleic Acids Research **38** (November 2009) D255–D266
17. Wang, Y., Xiao, J., Suzek, T.O., Zhang, J., Wang, J., Bryant, S.H.: Pubchem: a public information system for analyzing bioactivities of small molecules. Nucleic Acids Research **37** (June 2009) W623–W633