A Comparative Study on Feature Selection Methods for Drug Discovery

Ying Liu*

Georgia Institute of Technology, College of Computing, Atlanta, Georgia 30322

Received April 13, 2004

Feature selection is frequently used as a preprocessing step to machine learning. The removal of irrelevant and redundant information often improves the performance of learning algorithms. This paper is a comparative study of feature selection in drug discovery. The focus is on aggressive dimensionality reduction. Five methods were evaluated, including information gain, mutual information, a χ^2-test, odds ratio, and GSS coefficient. Two well-known classification algorithms, Naïve Bayesian and Support Vector Machine (SVM), were used to classify the chemical compounds. The results showed that Naïve Bayesian benefited significantly from the feature selection, while SVM performed better when all features were used. In this experiment, information gain and χ^2-test were most effective feature selection methods. Using information gain with a Naïve Bayesian classifier, removal of up to 96% of the features yielded an improved classification accuracy measured by sensitivity. When information gain was used to select the features, SVM was much less sensitive to the reduction of feature space. The feature set size was reduced by 99%, while losing only a few percent in terms of sensitivity (from 58.7% to 52.5%) and specificity (from 98.4% to 97.2%). In contrast to information gain and χ^2-test, mutual information had relatively poor performance due to its bias toward favoring rare features and its sensitivity to probability estimation errors.

1. INTRODUCTION

Drug discovery encompasses understanding cellular processes, predicting protein structures, and estimating interactions between a molecule and the normal biological molecular targets.[1] Chemists and biologists would ideally like to fully understand the pathways involved in a disease, and from this knowledge develop a molecule (or several molecules) that can interact with the disease agents to neutralize them. However, many complex interactions are occurring at the cellular level that makes the full rational drug design process extremely difficult. Methods have been developed to circumvent some of these problems through the use of high-throughput screening.[1]

The advent of combinational chemistry in the mid-1980s has allowed the synthesis of hundreds, thousands, and even millions of new molecular compounds. The need for a more refined search than simply producing and testing every single molecular combination possible has meant that statistical approaches and, more recently, intelligent computation have become an integral part of the drug production process.[2] Structure−activity relationship (SAR) analysis is one technique used to reduce the search for new drugs. SAR bases its prediction on the assumption that there exists a relationship between the structural or molecular features of a compound and its biological activity (such as chemical activity, aqueous solubility, blood-brain barrier penetration, oral absorption, or toxicity). SAR analysis aims at discovery these rules in order to predict the activity of new molecules based on their physiochemical descriptors.[3,4]

Machine learning techniques have been successfully applied to SAR analysis to predict if a compound is likely to demonstrate drug-like activity in the presence of a given disease (or simply a given chemical target).[5] Artificial neural networks have been used to discriminate potential drug-like molecules from large compound databases.[3] Wagener et al.[4] reached 70−80% accuracy in a similar type problem using decision tree algorithms. Burbidge et al.[2] showed that support vector machine outperformed other machine learning approaches for the prediction of inhibition of dihydrofolate reductase by pyrimidines.

Complex molecular compounds can be described by a large number of attributes or features, such as topological indices, characterizing the three-dimensional molecular structures, quantum mechanical descriptors, and molecular field parameters, which could be tens or hundreds of thousands of features. This is prohibitively high for many learning algorithms.[6] Therefore, a first step in the machine learning process consists of identifying the most relevant features for the problem at hand. The larger the number of irrelevant features in the input space is, the more difficult for the algorithms to identify a correct decision function: the system may not converge to an optimal solution in an acceptable amount of time, or much more training data may be needed to reach a correct solution. To reduce the dimensionality of the input space one will need to (1) identify all the variables relevant to the concept and determine how relevant they are and how related to one another and (2) choose a minimum subset of variables or alternative subsets that maximize the inducer's efficiency (i.e. provide good generalization). Aggressive reduction of the feature space has been repeatedly shown to lead to little accuracy loss and to a performance gain in text categorization.[7]

Automatic feature selection methods have been well studied in text categorization. Information-theoretic functions, such as information gain,[6,8−12] mutual information,[6,8,10] χ^2-

* Corresponding author phone: (404)385-6380; fax: (404)894-9442; e-mail: yingliu@cc.gatech.edu.

10.1021/ci049875d CCC: $27.50
Published on Web 07/08/2004

test,[6,8,9,14] odd ratios,[9,11,13] GSS coefficient,[15] association factor,[16] NGL-coefficient,[17,18] and relevancy score[19] have been used in the feature selection. Liu et al.[20] compared an entropy-based, χ^2-statistics, a correlation-based, a t-statistics, and an MIT correlation-based feature selection method using gene expression profiles and proteomic patterns.

The focus of this paper is to evaluate and compare different feature selection methods in the reduction of a high dimensional feature space in drug discovery. Two classifiers, Naïve Bayesian and support vector machine (SVM), were used to classify the chemical compounds.

2. METHODS

2.1. The Data Set. The data used in this paper was from the data set in the 2001 KDD cup.[21] In this classification competition, DuPont Pharmaceutical Research Laboratories made available the results of lab experiments that tested 1909 (training set) organic compounds for whether they bind to thrombin (a protease involved in blood clotting). Only 42 of the compounds showed a positive result. Each compound was described by a single feature vector comprised of a class value ("A" for active, "I" for inactive) and 139 351 binary features, which describe three-dimensional properties of the compound. The test set, in the 2001 KDD cup, included 634 compounds, of which 150 were active.[21]

2.2. Feature Selection Methods. In this paper, five methods were evaluated. The methods are all based on assigning a score to each feature that suggests how important or valuable the feature is likely to be for the training and categorization. Let $\{c_i\}_{i=1}^m$ denote the set of categories in the target space, c_i be the ith category, and $\overline{c_i}$ be the (m-1) categories other than c_i. In this paper, there were two categories ($m = 2$), active and inactive. Therefore, if c_i is the active category, then $\overline{c_i}$ is the inactive category.

2.2.1. Information Gain (IG). Information gain is frequently employed as a feature-goodness criterion in the field of machine learning. It measures the number of bits of information obtained for category prediction by knowing the presence or absence of a feature. It is measured as

$$IG(f_k) = \sum_{c\in(c_i,\overline{c_i})}\sum_{f\in(f_k,\overline{f_k})} Pr(f, c)\log\frac{Pr(f, c)}{Pr(f)\times Pr(c)} \quad (1)$$

where f_k means the presence of the feature k and $\overline{f_k}$ means the absence of feature k.

2.2.2. Mutual Information (MI). Mutual information (MI) is a basic concept in information theory. It is a measure of general interdependence between random variables. MI is commonly used in statistical language modeling of word associations and related applications. It is measured as

$$MI(f_k, c_i) = \log\frac{Pr(f_k, c_i)}{Pr(f_k)\times Pr(c_i)} \quad (2)$$

2.2.3. χ^2-Test (CHI). CHI measures the lack of independence between a feature f and a category c and can be compared to the χ^2 distribution with one degree of freedom to judge extremeness.[6] It is defined as

$$CHI(f_k, c_i) = \frac{N\times(Pr(f_k, c_i)\times Pr(\overline{f_k},\overline{c_i}) - Pr(f_k, \overline{c_i})\times Pr(\overline{f_k},c_i))^2}{Pr(f_k)\times Pr(\overline{f_k})\times Pr(c_i)\times Pr(\overline{c_i})} \quad (3)$$

2.2.4. Odds Ratio (OR). Odds ratio was proposed originally for selecting terms for relevance feedback in text classification. The basic idea is that the distribution of features on the relevant documents is different from the distribution of features on the nonrelevant documents.[11] It is defined as follows:

$$OR(f_k, c_i) = \frac{Pr(f_k|c_i)\times(1 - Pr(f_k|c_i))}{(1 - Pr(f_k|c_i))\times Pr(f_k|c_i)} \quad (4)$$

2.2.5. GSS Coefficient (GSS). GSS coefficient is a simplified variant of the χ^2 statistics proposed by Galavotti et al.,[15] which is defined as

$$GSS(f_k, c_i) = Pr(f_k, c_i)\times Pr(\overline{f_k},\overline{c_i}) - Pr(f_k, \overline{c_i})\times Pr(\overline{f_k},c_i) \quad (5)$$

For the methods with one value per category (MI, CHI, OR, GSS), the maximum value was used as the score, e.g.

$$MI(f_k) = \max_{i=1}^{m} MI(f_k, c_i) \quad (6)$$

$$CHI(f_k) = \max_{i=1}^{m} CHI(f_k, c_i) \quad (7)$$

$$OR(f_k) = \max_{i=1}^{m} OR(f_k, c_i) \quad (8)$$

$$GSS(f_k) = \max_{i=1}^{m} GSS(f_k, c_i) \quad (9)$$

2.3. Classifiers. After selecting the most discriminatory features, two classifiers were applied to assess the effectiveness of feature selection methods.

2.3.1. Naïve Baysian (NB). NB is a probabilistic learner based on the Bayes' rule. It is among the most practical approaches to certain types of learning problems.[20] The predicted category c for a compound d is the one that maximizes the posterior probability $P(c|d)$, which is proportional to $P(c)\Pi_i P(f_i|c)$, where $P(c)$ is the prior probability that a compound belongs to category c, and $P(f_i|c)$ is the probability that a feature i is chosen randomly in a compound from category c.

2.3.2. Support Vector Machine (SVM). SVM is a kind of blend of linear modeling and instance-based learning. An SVM selects a small number of critical boundary samples from each category and builds a linear discriminate function that separates them as widely as possible. In the case that no linear separation is possible, the technique of "kernel" will be used to automatically inject the training samples into a higher-dimensional space and to learn a separator in that space.[20,22] In linearly separable cases, SVM constructs a hyperplane which separates two different categories of feature vectors with a maximum *margin*, i.e., the distance between the separating hyperplane and the nearest training vector. The hyperplane was constructed by finding another vector

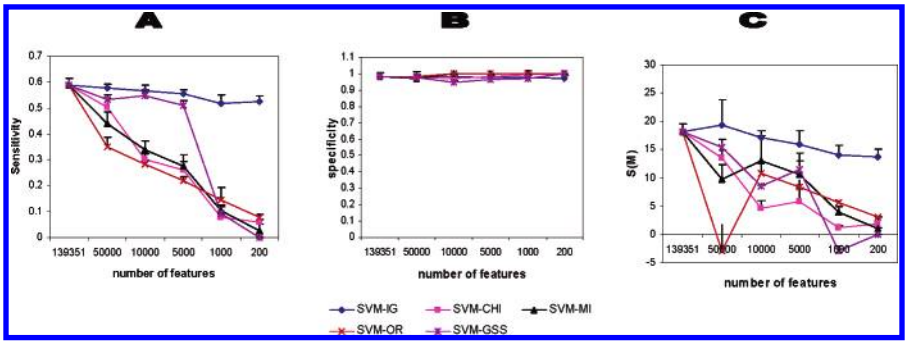

Figure 1. Effect of different feature selection methods in combination of SVM on sensitivity measure (A), specificity measure (B), and cost saving measure (C). Note the different scales on the vertical axes. The horizontal axes refer to the number of features used by SVM to classify the compounds. Error bars indicate the standard errors.

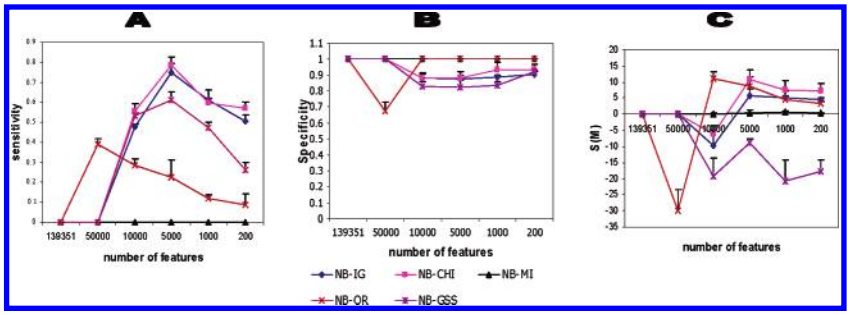

Figure 2. Effect of different feature selection methods in combination of Naïve Bayesian classifier on sensitivity measure (A), specificity measure (B), and cost saving measure (C). Note the different scales on the vertical axes. The horizontal axes refer to the number of features used by Naïve Bayesian to classify the compounds. Error bars indicate the standard errors.

w and a parameter b that minimizes $||w||^2$ and satisfies the following conditions

$$w \cdot x_i + b \geq +1, \text{ for } y_i = +1 \text{ Category 1 (active)}$$

$$w \cdot x_i + b \leq -1, \text{ for } y_i = -1 \text{ Category 2 (inactive)}$$

where y_i is the category index (i.e. active, inactive), w is a vector normal to the hyperplane, $|b|/||w||$ is the perpendicular distance from the hyperplane to the origin, and $||w||^2$ is the Euclidean norm of w. After the determination of w and b, a given vector x can be classified by sign$[(w \cdot x) + b]$.

In this paper, SVMLight v.3.5 was used.[22]

2.4. Cross-Validation of the Models. The normal method to evaluate the classification results is to perform cross-validation on the classification algorithms.[23] Tenfold cross-validation has been proved to be statistically good enough in evaluating the classification performance.[24] In this paper, the training and test sets from the KDD Cup Competition were merged into a single data set. Then the data set was partitioned into 10 subsets with both active and inactive compounds spread as equally as possible between the sets. Each of these sets in turn was set aside, while a model was built using the other nine sets. This model was then used to classify the compounds in the tenth set, and the accuracy was computed by comparing these predictions with the actual category. This process was repeated 10 times, and the results were averaged.[25]

2.5. Performance Measures. The effectiveness of the feature selection methods were evaluated using the performance of Naïve Bayesian and SVM classifiers on the drug data set mentioned in section 2.1. Several statistics were used as performance measures:

(1) Sensitivity: the percent of active compounds which were correctly classified.

(2) Specificity: the percent of inactive compounds which were correctly classified.

(3) Active predictivity: the percentage of the compounds predicted to be active that were correct.

(4) Inactivity predictivity: the percentage of the compounds predicted to be inactive that were correct.

(5) A Costing Function.[26] To judge overall performance, the cost of using the method M was defined as C(M) = fp(M) + 2*fn(M), where fp(M) was the number of false positives for method M, and fn(M) was the number of false negatives for method M. The false negatives were weighted more heavily than the false positives because, for these data, the number of active examples (192) was small compared with the number of inactive ones (2351). The cost for each method was compared with the cost C(N) for using the null learning procedure, which classifies all test examples as inactive. We defined the cost savings of using the learning procedure M as S(M) = C(N) − C(M).

3. RESULTS

The features were ranked based on the scores the feature selection methods assigned. And only the top-ranking features were used for classification purpose. The numbers of features tested were 200, 1000, 5000, 10 000, 50 000, and 139 351.

3.1. Sensitivity. The effect of the feature selection on the sensitivity of SVM and Naïve Bayesian results were shown in Figures 1A and 2A, respectively. SVM result had higher sensitivity when all the features were used. Naïve Bayesian performed well when the number of features were reduced to 500 000 by OR and 5000 by IG, CHI, and GSS. Under the tested conditions, there was no active compound predicted by a Naïve Bayesian classifier when MI was used to select

Table 1. Effect of Different Feature Selection Methods in Combination with Different Classifiers on the Active Predictivity and Inactive Predictivity Measures

	predictivity			predictivity	
SVM	active	inactive	NB	active	inactive
IG-ALL[a]	0.88	0.97	IG-ALL[a]	*b*	0.92
IG-50k	0.87	0.97	IG-50k	*b*	0.92
IG-10k	0.84	0.97	IG-10k	0.47	0.95
IG-5k	0.84	0.96	IG-5k	0.51	0.98
IG-1k	0.80	0.96	IG-1k	0.42	0.96
IG-200	0.80	0.96	IG-200	0.54	0.96
CHI-ALL[a]	0.88	0.97	CHI-ALL[a]	*b*	0.92
CHI-50k	0.83	0.96	CHI-50k	*b*	0.92
CHI-10k	0.71	0.94	CHI-10k	0.48	0.96
CHI-5k	0.62	0.94	CHI-5k	0.55	0.98
CHI-1k	0.25	0.93	CHI-1k	0.55	0.97
CHI-200	0.50	0.93	CHI-200	0.60	0.96
MI-ALL[a]	0.88	0.97	MI-ALL[a]	*b*	0.92
MI-50k	0.74	0.96	MI-50k	*b*	0.92
MI-10k	1.00	0.95	MI-10k	0.1	0.92
MI-5k	1.00	0.94	MI-5k	0.2	0.93
MI-1k	0.25	0.93	MI-1k	0.2	0.93
MI-200	0.50	0.93	MI-200	0.2	0.93
OR-ALL[a]	0.88	0.97	OR-ALL[a]	*b*	0.92
OR-50k	0.27	0.93	OR-50k	0.21	0.93
OR-10k	1.00	0.94	OR-10k	1.00	0.95
OR-5k	1.00	0.94	OR-5k	1.00	0.94
OR-1k	1.00	0.93	OR-1k	1.00	0.93
OR-200	0.70	0.93	OR-200	0.80	0.93
GSS-ALL[a]	0.88	0.97	GSS-ALL[a]	*b*	0.92
GSS-50k	0.85	0.96	GSS-50k	*b*	0.92
GSS-10k	0.79	0.96	GSS-10k	0.42	0.95
GSS-5k	0.78	0.96	GSS-5k	0.45	0.96
GSS-1k	0.24	0.93	GSS-1k	0.40	0.94
GSS-200	*b*	0.92	GSS-200	0.40	0.94

[a] ALL: all the features (139 351) were used. [b] No active compound was predicted.

the features which showed, in Figure 2A, that the sensitivity was 0.

3.2. Specificity. Figures 1B and 2B showed the effect of the feature selection on the specificity of SVM and Naïve Bayesian classifiers. SVM was less sensitive to the feature selection than Naïve Bayesian. When all the features were used to classify the compounds, no inactive compounds were misclassified (no false positive), which showed in the figures that the specificity was 1. Since all the compounds were predicted as inactive by a Naïve Bayesian classifier when MI was used to select the features under the tested conditions, the specificity was always 1 (Figure 2B). There were no false positives when OR was used to select the features, and a Naïve Bayesian classifier was used to classify the compounds except when the feature number was 50 000.

3.3. Active Predictivity and Inactive Predictivity. The activity predictivity and inactivity predicitity results were summarized in Table 1. When all the features (139 351) were used for classification purposes, no active compound was predicted by a Naïve Bayesian classifier.

3.4. Costing Saving. SVM did not benefit from the feature reduction, except that when the number of features was reduced to 50 000 by IG, the cost saving raised from 18.2 to 19.3 (Figure 1C), which was not statistically significant ($p = 0.738$). A Naïve Bayesian classifier performed well when the number of features was reduced to 5000 (about 96% removal) by IG and CHI. A Naïve Bayesian classifier was a null learning procedure, which classifies all compounds as inactive, when MI was used as the feature selection method. Therefore, the cost saving was 0 (Figure 2C). The cost saving increased significantly ($p < 0.001$) (Figures 1C and 2C), when features were selected by OR and the number of features were reduced from 50 000 to 10 000.

4. DISCUSSION

Feature selection, as a preprocessing step to machine learning, is effective in reducing dimensionality, removing irrelevant data, increasing learning accuracy, and improving result comprehensibility.[27] In this paper, five feature selection methods were evaluated and compared. The effect of the feature selection on the quality of different classifiers were measured by sensitivity, specificity, active predictivity, and inactive predictivity, as these evaluation measures are commonly used in machine learning. A cost saving function was also used for evaluation purpose.

4.1. Naïve Bayesian vs SVM. The experiments with a Naïve Bayesian classifier confirmed the well-known fact that a Naïve Bayesian classifier benefits greatly from appropriate feature selection.[13] When the number of features was reduced to 5000 by IG, CHI, or OR, the sensitivity and cost saving of a Naïve Bayesian classifier results increased significantly (Figure 2A,C).

The results of the experiments indicated that SVM did not benefit from feature selection, which had been reported in text classification.[6,8,13] Taira and Haruno[28] compared SVM and decision tree in text categorization, and the best average performance was achieved when all the features were given to SVM, which was a distinct characteristic of SVM compared with the decision tree learning algorithm. Joachims[29] argued that, in text classification, feature selection was often not needed for SVM, as SVM tends to be fairly robust to overfitting and can scale-up to considerable dimensionalities. Furthermore, our result showed that when IG was used to select the features, SVM was much less sensitive to the reduction of feature space. The number of features was reduced by 99% (from 139 351 to 200), while losing only a few percent in terms of sensitivity (from 58.7% to 52.5%) and specificity (from 98.4% to 97.2%). Rogati and Yang[8] and Brank et al.[13] had a similar observation when they compared different feature selection methods for text classification.

4.2. Feature Selection Methods. An observation emerged from the classification results of Naïve Bayesian, which was IG and CHI had similar effects on the performance of the classifier (Figure 2). Both of them can eliminate more than 90% of the features with an improvement in classification accuracy (as measured by sensitivity and cost saving). Using IG as the feature selection method, for example, the number of features was reduced from 139 351 to 50 000, and the sensitivity of a Naïve Bayesian classifier was improved from 0% to 75%. CHI had even better classification results. The similar performance of IG and CHI in feature selection had previously been reported in text categorization.[6] Yang and Pedersen[6] found that there were strong correlations between IG and CHI values of a feature, and the correlations were general in text categorization. Moreover, IG and CHI shared the same bias, i.e., scoring in favor of common features over rare features. The good performance of IG and CHI indicated that common features were informative for classification tasks.

In this experiment, the result indicated that OR, as a feature selection method, can improve the performance of a Naïve Bayesian classifier. In their study of Naïve Bayes, Mladenic and Grobelnik[30] found that the feature selection based on odds ratio scores had consistently resulted in statistically significant improvements in classification performance over the use of the full feature set. Compared with other feature selection methods, OR did not improve SVM performance well. This was in contrast to earlier research on text categorization, when OR was found to work well in combination with the SVM classifier.[13] The difference may be attributable to the fact that the most discriminating features in text analysis are relatively common by nature. The cost saving raised significantly when the number of features was reduced from 50 000 to 10 000 when OR was used to select the features (Figures 1C and 2C). This may be due to the fact that feature selection has the risk to select features that are not actually relevant, omit features that are, and overstate the value of the features that end up selected.[31,32] When the number of features decreased to 50 000, some potentially useful features on the meaning of the compounds may be removed. Nonuseful or redundant features were disregarded when the feature size was reduced to 10 000, which led to the improvement of cost saving measure.

The poor performance of MI was also informative. Its bias toward low-frequency feature is known. Furthermore, MI is very sensitive to probability estimation error.[6] Yang and Pedersen[6] pointed out that information gain is the weighted average of the mutual information $MI(f_k, c)$ and $MI(\bar{f}_k, c)$, where the weights are the joint probabilities $Pr(f_k, c)$ and $Pr(\bar{f}_k, c)$, respectively. So information gain is also called average mutual information. There are two fundamental differences between IG and MI: (1) IG makes use of information about feature absence in the form of $MI(\bar{f}_k, c)$, while MI ignores such information; and (2) IG normalizes the mutual information scores using the joint probabilities while MI uses the nonnormalized scores.[6] Using a cross-method comparison, the results of this paper quantitatively showed that the theoretical weakness of MI caused significant accuracy loss in classification tasks.

4.3. Feature Selection and Overfitting. Overfitting is the use of models or procedures that include more features than are necessary or use more complicated approaches than are necessary.[31] Therefore, there are two types of overfitting: (1) using a model that is more flexible and complicated than it needs to be and (2) using a model that includes irrelevant features or components. Feature selection methods aim to provide protection against the second type of overfitting.[33] Complex molecular compounds, such as potential drugs, can be described by a large number of attributes or features. The compounds in the data set tested in this study were described by 139 351 binary features. This is prohibitively high for many learning algorithms.[6] Proper feature selection methods can be applied to reduce the feature size, while they keep and even improve the classification performance.

CONCLUSION

This paper is an evaluation of feature selection methods in dimensionality reduction for drug discovery at all the reduction levels of aggressiveness, from using all the features to removing more than 99% of the features. SVM did not benefit from feature selection, while feature reduction improved classification accuracy of a Naïve Bayesian classifier. Information gain and χ^2-test were more effective than other methods in feature removal. Mutual information had inferior performance compared with other methods due to its bias favoring rare features and a strong sensitivity to probability estimation errors.

ACKNOWLEDGMENT

I thank Dr. Liping Wei and Theodore A. Chiang for helpful discussions. The author is grateful to the editor and the two anonymous reviewers for a number of suggestions for improvement.

REFERENCES AND NOTES

(1) Corwin, C.; Kuntz, I. D. Database searching: Past, present and future. In *Designing bioactive molecules*; Martin, Y. C., Willett, P., Eds.; American Chemical Society: Washington, DC, 1998; pp 1−16.
(2) Burbridge, R.; Trotter, M.; Buxton, B.; Holden, S. Drug design by machine learning: support vector machines for pharmaceutical data analysis. Proceedings of the AISB'00 Symposium on Artificial Intelligence in Bioinformatics, 2000, pp 1−4.
(3) Frimurer, T. M.; Bywater, R.; Narum, L.; Lauritsen, L. N.; Brunak, S. Improving the odds in discriminating drug-like from non drug-like compounds. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 1315−1324.
(4) Wagener, M.; van Geerestein, V. Potential drugs and nondrugs: prediction and identification of important structural features. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 280−292.
(5) Eschrich, S.; Chawla, N. V.; Hall, L. O. Generalization methods in Bioinformatics. BIOKDD02: Workshop on Data Mining in Bioinformatics, 2002, pp 25−32.
(6) Yang, Y.; Pederson, J. O. A comparative study on feature selection in text categorization. International Conference on machine Leaning (ICML'97), 1997; pp 412−420.
(7) Dabrowski, Y.; Deltorn, J.-M. Machine learning application to drug design. http://www.miningmachines.com
(8) Rogati, M.; Yang, Y. High-performing feature selection for text classification. CIKM'02, 2002, pp 659−661
(9) Caropreso, M. F.; Martwin, S.; Sebastiani, F. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In *Text Databases and Document Management: Theory and Practice*; Chin, A. G., Ed.; Idea Group Publishing: Hershey, PA, 2001; pp 78−102.
(10) Lewis, D. D.; Ringuette, M. A comparison of two learning algorithms for text categorization. Proceedings of SDAIR-94, 1994; pp 81−93.
(11) Mladenic, D. Feature subset selection in text learning. Proceedings of ECML-98, 1998; pp 95−100.
(12) Moulinier, I.; Ganascia, J. G. Applying an existing machine learning algorithm to text categorization. In *Connectionist, Statistical and Symbolic Approaches to Learning for Natural Language Processing*; Wermter, S., Riloff, E., Schaler, G., Eds.; Spring-Verlag: Heidelberg, Germany, 1998; pp 343−354
(13) Brank, J.; Grobelnik, M.; Milic-Frayling, N.; Mladenic, D. Interaction of feature selection methods and linear classification models. Workshop on Text Learning (TextML-2002), 2002.
(14) Sebastiani, F.; Sperduti, A.; Valdambrini, N. An improved boosting algorithm and its application to automated text categorization. CIKM'00, 2000; pp 78−85.
(15) Galavotti, L.; Sebastiani, F.; Simi, M. Experiments on the use of feature slection and negative evidence in automated text categorization. ECDL'00, 2000; pp 59−68.
(16) Fuhr, N.; Hartmann, S.; Knorz, G.; Lustig, G.; Schwantner, M.; Tzeras, K. AIR/X−a rule-based multistage indexing system for large subject fields. RIAO'91, 1991; pp 606−623.
(17) Ng, H. T.; Goh, N. B.; Low, K. L. Feature selection, perceptron learning, and a usability case study for text categorization. Proceedings of SIGIR-97, 1997; pp 67−73.
(18) Ruiz, M. E.; Srinivasan, P. Hierarchical neural networks for text categorization. Proceedings of SIGIR-99, 1999; pp 281−282.
(19) Wiener, E. D.; Pedersen, J. O.; Weigend, A. S. A neural network approach to topic supporting. Proceedings of SDAIR-95, 1995; pp 317−332.
(20) Liu, H.; Li, J.; Wong, L. A comparative study on feature selection and classification methods using gene expression profiles and proteomic pattern. *Genomic Informatics* **2002**, *13*, 51−60.

(21) Chang, J.; Hatzis, C.; Hayashi, H.; Krogel, M. A.; Morishita, S.; Page, D.; Sese, J. KDD cup report. *ACM SIGKDD Explorations* **2001**, *3*, 47−64.
(22) Joachims, T. Transductive Inference for Text Classification using Support Vector Machines. International Conference on Machine Learning (ICML'99), 1999; pp 200−209.
(23) Tan, A. C.; Gilnbert, D. Ensemble machine learning on gene expression data for cancer classification. *Appl. Bioinformatics* **2003**, *3*, S75−S83.
(24) Witten, I. H.; Frank, E. *Data mining: practical machine learning tools and techniques with java implementations*; Morgan Kaufmann: San Francisco.
(25) Bahler, D.; Stone, B.; Wellington, C.; Bristol, D. W. Symbolic, neural, and Bayesian machine learning models for predicting carcinogenicity of chemical compounds. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 906−914.
(26) Brown, M. P. S.; Grundy, W. N.; Lin, D.; Cristianini, C.; Sugnet, C. W.; Furey, T. S.; Ares, M.; Haussler, D. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl. Acad. Sci. U.S.A.* **2000**, *97*, 262−267.
(27) Yu, L.; Liu, H. Feature selection for high-dimensional data: a fast correlation-based filter solution. ICML-03, 2003; pp 856−863.
(28) Taira, H.; Haruno, M. Feature selection in SVM text categorization. AAAI-99, 1999; pp 480−486.
(29) Joachims, T. Text categorization with support vector machines: learning with many relevant features. Proceedings of ECML-98, 1998; pp 137−142
(30) Mladenic, D.; Grobelnik, M. Feature selection for unbalanced class distribution and Naïve Bayes. Proceedings of the 16th Int. Conf. on Machine Learning, 1999; pp 258−267
(31) Hawkins, D. M. The problem of overfitting. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1−12
(32) Sebastiani, F. Machine learning in automated text categorization. *ACM Comput. Surveys* **2002**, *34*, 1−47
(33) Xing, E. P.; Jordan, M. I.; Karp, R. M. Feature Selection for High-Dimensional Genomic Microarray Data. Proc. 18th International Conf. on Machine Learning (ICML-01), 2001; pp 601−608

CI049875D