

Using Decision Trees and Support Vector Machines to Classify Genes by Names

Abstract: In this paper we report an application of machine learning methods to classify gene names into two categories: known and unknown ones. We acquired a data set of 1,624 genes by letting a human expert classify them manually. To capture the knowledge of classification, we also asked the expert to derive a set of rules. In parallel, we trained two machine learners to capture the same knowledge. Both decision trees (CART) and Support Vector Machines (SVMs) outperform the expert rules; the cross-validated error rates are below 1%, and the area under the curve of Receiver Operating Characteristic (ROC) curves reach higher than 0.99. In summary, CART and SVMs reduced the overall error rate of prediction by 40% and 88%, respectively, compared with classification using expert rule sets. In addition, the machine classifiers are able to find some errors made by the human expert himself. Finally, we used the expert system to classify 7,447 genes on the Affymetrix U74A microarray chip. Results show 70% of the genes on this chip are known ones. In conclusion, we successfully demonstrate that the machine-derived classifiers are more capable of handling the job efficiently than the expert-derived classifier. It further supports the idea that in many application domains, experts can perform the task, but cannot tell how; whereas expert systems are able to capture the knowledge from the experts.

Keywords: Gene name, Known gene, unknown gene, Classification and Regression Tree (CART), Support Vector Machines (SVM), expert system, text data mining

Corresponding author:

Simon M. Lin, M.D.

Lin00025@mc.duke.edu

Tel: 919-681-9646

FAX: 919-681-8028

Box 3958, Duke University Medical Center

Durham, NC 27710

Using Decision Trees and Support Vector Machines to Classify Genes by Names

Simon Lin¹, Sandip Patel², Andrew Duncan², and Linda Goodwin³, {Simon.Lin, Sandip.Patel, Aduncan, Linda.Goodwin}@duke.edu, ¹Bioinformatics Shared Resource, ²Department of Pharmacology and Cancer Biology, and ³School of Nursing, Duke University Medical Center, Durham, NC 27710

ABSTRACT

In this paper we report an application of machine learning methods to classify gene names into two categories: known and unknown ones. We acquired a data set of 1,624 genes by letting a human expert classify them manually. To capture the knowledge of classification, we also asked the expert to derive a set of rules. In parallel, we trained two machine learners to capture the same knowledge. Both decision trees (CART) and Support Vector Machines (SVMs) outperform the expert rules; the cross-validated error rates are below 1%, and the area under the curve of Receiver Operating Characteristic (ROC) curves reach higher than 0.99. In summary, CART and SVMs reduced the overall error rate of prediction by 40% and 88%, respectively, compared with classification using expert rule sets. In addition, the machine classifiers are able to find some errors made by the human expert himself. Finally, we used the expert system to classify 7,447 genes on the Affymetrix U74A microarray chip. Results show 70% of the genes on this chip are known ones. In conclusion, we successfully demonstrate that the machine-derived classifiers are more capable of handling the job efficiently than the expert-derived classifier. It further supports the idea that in many application domains, experts can perform the task, but cannot tell how; whereas expert systems are able to capture the knowledge from the experts.

Keywords

Gene name, Known gene, unknown gene, Classification and Regression Tree (CART), Support

Vector Machines (SVM), expert system, text data mining

1. INTRODUCTION

The difference between a gene name versus its database identifier is that a name usually conveys some comprehensible meaning. Simply by its name, biologists can usually tell if a gene is a known one or an unknown one. By classifying the gene into the known category, the expert indicates that the gene was previously studied or characterized. For example, “superoxide dismutase” is a known gene, whereas “ESTs, Weakly similar to MAP-kinase activating death domain” is an unknown one.

Identifying if the gene is known or unknown has many practical applications:

- In computational annotation of genomic sequences, annotators are generally transferring the name and function of a known gene to the un-annotated one. By knowing which genes are already functionally known, we can prevent “null-chaining” by claiming a gene’s function and directing it to another unknown gene.
- In pharmaceutical research, unknown genes are usually given higher priority as a potential novel drug target. Quickly finding out what are the unknown genes in a large screening data set will let the investigator make an informed decision regarding which targets to pursue experimentally.
- In analyzing the statistics of genomic databases, we need to know what

percentage of the genes are still functionally unknown.

- In the tracking and cleaning of genomic data warehouses, it is crucial to monitor the status of the genes being changed from unknown to known. Updated nomenclature of the genes can expedite biological discoveries by providing information regarding an unknown gene that might not otherwise be considered by a human viewing database identifiers.

However, judging if a gene is known or unknown by using an expert to look at the name is not only time consuming but is also prone to human errors. Thus, we would like to devise a machine that can mimic human experts in performing the same task. That casts the problem into a typical machine learning task, which usually contains two phases. In phase one, a data set will be acquired from a human expert; i.e., the true answer to the problems are labeled by the expert. In this phase, a machine learner will be trained. Cross-validation will be used to assess how well the machine 'learns' and to prevent overfitting. In phase two, we will use the machine to perform the classification task automatically, i.e., the production phase.

In this paper, we report how we utilized two machine learning tools -- decision trees (CART) and support vector machines (SVM)-- to classify genes into known and unknown categories by looking at their names.

2. RELATED WORK

Since no previous research has been reported on categorizing genes into known and unknown groups by their names, we reviewed literature in text categorization and machine learning.

In text categorization, it is indicated (Yang & Liu, 1999) that SVM and k-nearest neighbor (kNN) techniques outperform neural networks (NNet) and naïve Bayes (NB) classifiers. Rule-based learners were also studied due to their expressive power (Cohen, 1996).

Artificial intelligence and expert systems research have a great impact not only on medical informatics but also on bioinformatics. Intelligent Systems in Molecular Biology (ISMB), as coined by Larry Hunter and colleagues more than ten years ago, has continuously been a major conference in bioinformatics. Here we report the design and application of an intelligent system that can classify genes by names.

3. METHODS

Knowledge acquisition from the expert

Training data set. We randomly selected 1,624 genes and presented their names to the human expert. The expert was asked to classify the genes into known or unknown categories.

Expert rule-sets. We also asked the expert how he conducted the classification. The expert generated a set of rules that included certain keywords (See Figure 1).

Data model

We chose a much simpler data model than the commonly used term frequency/inverse document frequency weighting (TF-IDF) model (Salton, 1991). Each gene name is represented by a vector of words, where 1 indicates the presence of the word and 0 indicates the absence of the word. A survey of the training data indicates the gene names have a different structure than regular text data. Thus, we do not remove the so called stop words, such as 'and', 'or', and 'is'. However, we do filter out infrequent words and numbers from the training set. A high-pass filter at a frequency of 2 is used. Our classification performance indicates that this simple data model is adequate for this task.

CART

We used the `rpart` implementation of CART (Breiman, 1993) in the R-statistical package

(<http://cran.r-project.org>). Five-fold cross validation was used to select the complexity (Cp) parameter. Cp parameter controls the size of the tree. Large tree can cause over fitting. Thus, we used the parsimonious “1-SE” rule (Venables & Ripley, 2002) to choose the Cp associated with the largest error rate within one standard deviation of the minimum error rate.

SVM

SVM is a new learning method introduced by Vapnik and coworkers (Vapnik, 1995). We used the SVMlib (Chang & Lin, 2001) implemented in the R-statistical package. Different choice of kernels-- linear, polynomial, radial basic function, and sigmoid -- flexibly maps the input space into a higher-dimensional space where the cases are separated with the maximum margin. Five-fold cross validation was used to select the appropriate kernels.

ROC curve comparison

To compare the two classification methods, we use the ROC analysis (Metz, 1978). Receiver Operator Characteristics (ROC) curves plot sensitivity (Y axis) against 1 minus the specificity (X axis) providing a clear visualization of area under the curve (AUC). The greater the accuracy, the greater the AUC (1.0 is perfect classification). Thus, the ‘better’ classification method is the one with the most area under the curve.

Outside data set to be classified

We tested the production expert system with a set of 7,447 genes to be classified. These genes were retrieved from the Affymetrix U74A DNA microarray chip.

4. RESULTS

Training data set acquisition

We acquired a training data set consisting of 1,624 mouse gene names, of which 698 genes are known, and 926 genes are unknown. After filtering out

infrequent words, this training data set consists of 522 unique words as attributes, and expressed as a 1,624 by 522 data matrix of zeros and ones. The training data set acquired from the expert is called T0 (see Table 1). During the machine learning process, we found errors in this data set (details are discussed later). The corrected data set is called T1.

Table 1. Training data sets.

	Data Set T0	Data Set T1
unknown	926	929
known	698	695
total	1,624	1,624

Expert rule-set acquisition

The expert summarized his knowledge into a set of rules to look at certain keywords (Figure 1). Performance of the rule-based classifier is summarized in Table 2.

```

if ((x$ESTS==1)) {
  y.predicted <- 0}
if ((x$HYPOTHETICAL==1)) {
  y.predicted <- 0}
if ((x$SIMILAR==1)& (x$TO==1)) {
  y.predicted <- 0}
if ((x$CDNA==1)& (x$SEQUENCE==1)) {
  y.predicted <- 0}
if ((x$RIKEN==1) & (x$CDNA==1)) {
  y.predicted <- 0}
if ((x$EXPRESSED==1) & (x$SEQUENCE==1)) {
  y.predicted <- 0}
if ((x$DNA==1) & (x$SEGMENT==1)) {
  y.predicted <- 0}
if ((x$MUS==1)& (x$MUSCULUS==1) &
(x$CLONE==1)){
  y.predicted <- 0}

```

Figure 1. Classification rules derived by the expert.

Table 2. Confusion table by expert rules using training data set T1.

	true.unknown	true.known
predicted.unknown	923 (56.8%)	10 (0.6%)
predicted.known	6 (0.4%)	685 (42.2%)

Classifier by CART

CART was used to induct the decision tree. To determine the best complexity of the tree, we ran a five-fold cross validation. According to the “1-SE” rule, we chose $C_p = 0.0018$ (Figure 2), which corresponds to a tree with 9 leaves (Figure 3).

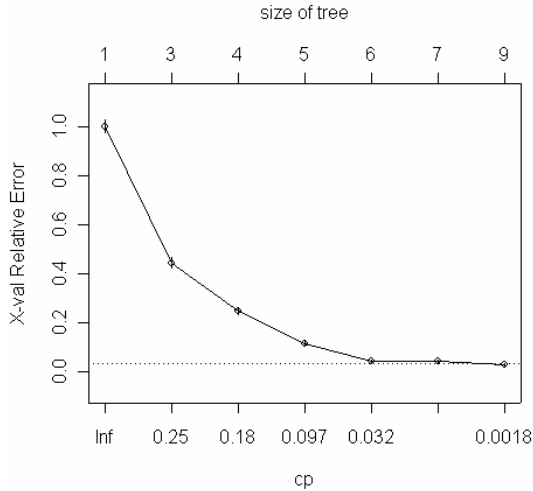


Figure 2. Using five fold cross-validation to determine the best complexity parameter C_p . The dotted line indicates the highest error rate within one SE of the lowest cross-validated training error. Results were obtained using training data set T0.

After cross-validation, we used all training data in set T1 to induct the tree. Results are shown in Figure 3 and Table 3.

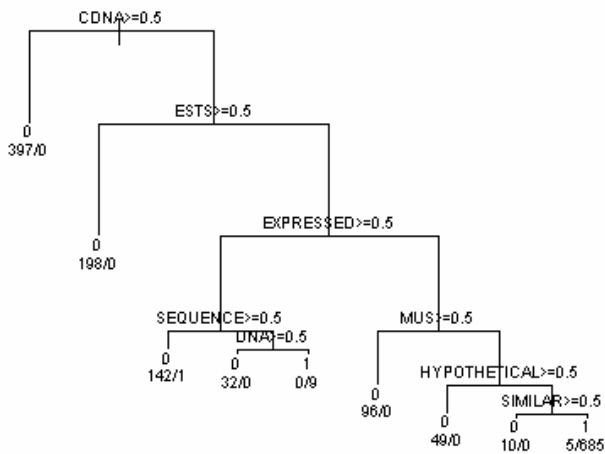


Figure 3. Decision tree by CART analysis. Unknown genes (class 0) are distinguished from know genes (class 1) by answering a series of questions in the flow chart. The purity of each leaf is shown below each leaf (class0/class1).

Table 3. Confusion table by CART using training data set T1.

	true.unknown	true.known
predicted.unknown	924 (56.9%)	1 (0.1%)
predicted.known	5 (0.3%)	694 (42.7%)

Classifier by SVM

To choose the best kernel for SVM, we ran a five-fold cross validation. A simple linear kernel is the best choice (Table 4). The performance of the SVM classifier is summarized in Table 5.

Table 4. Choice of kernel by SVM using five fold cross-validation (data set T1).

Kernel	Error Rate (%)
linear (gamma=0.0019)	0.3 ± 0.4
polynomial (degree=3, gamma=0.0019)	42.8 ± 2.0
RBF (gamma=0.0019)	8.8 ± 5.4
sigmoid (gamma=0.0019)	25.1 ± 18.9

Table 5. Confusion table by linear SVM using training data set T1.

	true.unknown	true.known
predicted.unknown	928 (57.1%)	1 (0.06%)
predicted.known	1 (0.06%)	694 (42.7%)

Not all training data points are equally important in determining the decision boundary of SVM. “Support vectors” are those critical data points that are close to the separating hyperplane. A list of the supporting vectors is found in Table 6.

Table 6. Support vectors indicates those cases close to the decision boundary.

	gene.name	support
1	acetylcholinesterase	1.0000000
2	aconitase 1	1.0000000
3	apelin	1.0000000
4	axin	1.0000000
5	carbonic anhydrase like sequence 1	1.0000000
6	expressed sequence 2 embryonic lethal	1.0000000
7	major urinary protein 2	1.0000000
8	placental protein 6	0.9363062
9	EST AI426782	-1.0000000
10	EST AI447490	-1.0000000
11	ESTs	-1.0000000
12	expressed sequence 2 embryonic lethal	-1.0000000
13	expressed sequence AA408140	-1.0000000
14	expressed sequence AA420392	-1.0000000
15	expressed sequence AA517758	-1.0000000
16	hypothetical protein MNCb 4414	-1.0000000
17	hypothetical protein DKFZp564K0822	-0.9991421
18	RIKEN cDNA 0610007P06 gene	-1.0000000

Machine classifiers detect errors made by experts in the training data set

Using CART and SVM, we found some mistakes made by the human expert in training data set T0, and revised the training data set into set T1 accordingly (Table 7). In the next section, we use the training data set T1 to re-induct the classifiers and compare them.

Table 7. Mistakes made by the human expert in training data set T0.

Gene Name	Label in T0	Label in T1
ESTs Weakly similar to A Chain A Crystal Structure Of The Human Acyl Protein Thioesterase 1 At 1 5 A Resolution H sapiens	known	unknown
expressed sequence Al173355	known	unknown
hypothetical gene LOC150274 clone MGC 41340 IMAGE 1244839 mRNA	known	unknown

Comparing the three classifiers

Comprehensibility. The expert rules and CART are easily comprehensible. They describe what attributes are important for classification; a flow chart is given for performing the classification. In contrast, SVM reveals the classification from another angle by looking at the critical cases. In that sense, Nishikawa and colleagues (El-Naqa *et al.*, 2002) called SVM a template-matching detector. Table 6 shows the cases on the decision boundary whose ‘support’ is important to make the decision. They consist of the difficult-to-classify examples on the “border line”. The SVM classifier memorizes these cases as critical knowledge to perform the task. In general, we can see the three classifiers acquired the same knowledge such as ‘Riken cDNA’ and ‘hypothetical protein’ etc for the classification.

Performance and efficiency. As shown in Table 8, expert rules are the most expensive to obtain. It took more than 10 hours for the expert to generate this set of rules. The production runtime for the three methods are comparable. The error rate in

Table 8 is a non cross-validated error rate, since it is impossible to cross-validate the expert rule process.

Table 8. Comparison of three classifiers.

	Expert Rule	CART	SVM
nominal error rate	0.99%	0.37%	0.12%
rule induction time	> 10 hr	< 5 min	< 5 min
production runtime	< 5 min	< 5 min	< 5 min

ROC. Since no threshold structure exists in the expert rules, only CART and SVM are compared here. Both CART and SVM achieve a good balance of sensitivity and specificity. The area under the curve for CART and SVM are 0.9971 and 0.9995, respectively (see Figure 4).

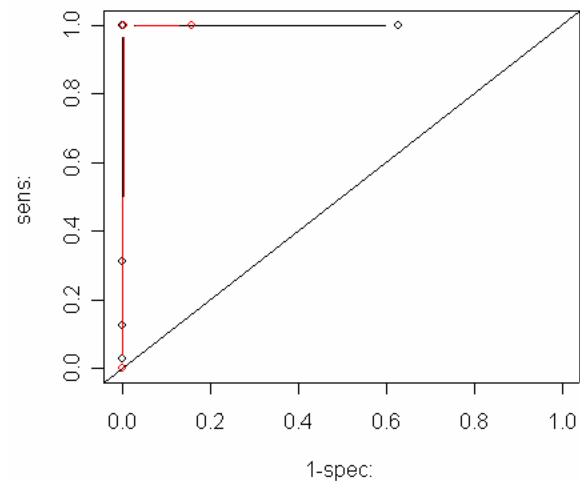


Figure 4. ROC analysis of CART and SVM. Red: CART; Black: SVM. The area under the curve (AUC) of CART and SVM are 0.9971 and 0.9995, respectively.

Application of the machine classifiers

In the production phase, we applied the two machine classifiers to categorize 7,447 genes on the Affymetrix U74A microarray DNA chip.

Table 9. Prediction results of CART and SVM for the 7,447 genes on the Affymetrix U74A microarray chip.

	CART	SVM
predicted.unknown	1747 (23.5%)	1797 (24.0%)
predicted.known	5700 (76.5%)	5650 (76.0%)
total	7447 (100%)	7447 (100%)

The major discrepancy of CART and SVM prediction results in Table 9 is that of 51 genes, where CART classified them as known but SVM classified them as unknown. A manual inspection of these genes indicated that the CART rule is unable to handle cases like ‘EST C78513’ (34 cases) and ‘DNA Segment Chr 6 human D12S2489E’ (12 cases) correctly. Thus, the production performance is in agreement with the evaluations on the training data set. The SVM classifier consistently outperforms the CART classifier in this application.

5. DISCUSSION

We have successfully built an expert system that is able to capture knowledge from the expert to perform a gene name classification task. Machine classifiers have also been used to find classification errors by experts in the training data set. Similar experience has also been reported in a leukemia microarray data set (Golub *et al.*, 1999), where a consensus of different classifiers strongly suggests the potential errors made by the human expert (Lin & Johnson, 2002).

In this application domain, we observed that SVM outperforms CART, although the prediction errors of both are acceptably low for production purposes. It can be explained by the capability of SVM to handle high-dimensional data based on Vapnik’s statistical learning theory. In contrast, CART utilizes tree nodes to select only a small number of attributes for classification. In text data mining, a variable selection by CART might not be appropriate, because the message in the text is not only conveyed by certain keywords but also by other words in the context. In other words, text data is presented by a dense concept vector where few irrelevant features exist (Joachims, 1998).

We demonstrated an application of this expert system where we classified the genes on a mouse U74A chip. It is one of the most commonly used chips in biomedical research. According to Affymetrix, on the U74A microarray chip, the majority of the genes are known ones. However, there is no quantitative description of what this ‘majority’ is. Here we estimate 70% of the genes on U74A are known ones. With our results, we can tell which genes are known or unknown on this chip. This system can help to accelerate the drug development process where priority is given to unknown genes after a microarray screening.

6. ACKNOWLEDGMENTS

The authors thank Kim Johnson for motivating discussions.

7. REFERENCES

- [1] Breiman L. (1993). "Classification and regression trees," Chapman & Hall, New York, N.Y.
- [2] Chang C. C., and Lin C. J. (2001). Training nu-support vector classifiers: Theory and algorithms. *Neural Computation* **13**: 2119-2147.
- [3] Cohen W. (1996). Learning rules that classify e-mail. In "Proc. Machine Learning in Information Access", AAAI.
- [4] El-Naqa I., Yang Y. Y., Wernick M. N., Galatsanos N. P., and Nishikawa R. M. (2002). A support vector machine approach for detection of microcalcifications. *IEEE Transactions on Medical Imaging* **21**: 1552-1563.
- [5] Golub T. R., Slonim D. K., Tamayo P., Huard C., Gaasenbeek M., Mesirov J. P., Coller H., Loh M. L., Downing J. R., Caligiuri M. A., Bloomfield C. D., and Lander E. S. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* **286**: 531-7.
- [6] Joachims T. (1998). Text categorization with support vector machines: learning with many relevant features. In "Proceedings of {ECML}-98, 10th European Conference on Machine Learning", Springer Verlag, Heidelberg.

- [7] Lin S. M., and Johnson K. F. (2002). "Methods of microarray data analysis : papers from CAMDA '00," Kluwer Academic Publishers, Boston.
- [8] Metz C. E. (1978). Basic principles of ROC analysis. *Semin Nucl Med* **8**: 283-98.
- [9] Salton G. (1991). Developments in Automatic Text Retrieval. *Science* **253**: 974-980.
- [10] Vapnik V. N. (1995). "The nature of statistical learning theory," Springer, New York.
- [11] Venables W. N., and Ripley B. D. (2002). "Modern applied statistics with S," Springer, New York.
- [12] Yang Y., and Liu X. (1999). A re-examination of text categorization methods. *In* "Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information", ACM Press, New York.