On-line Supplement to "Text Categorization Models of High Quality Articles in Internal Medicine"

Y. Aphinyanaphongs, M.S.

I. Tsamardinos, Ph.D.

A. Statnikov, M.S.

D. Hardin, Ph.D.*

C.F. Aliferis M.D., Ph.D.

Department of Biomedical Informatics, Vanderbilt University, Nashville, TN

*Department of Mathematics, Vanderbilt University, Nashville TN

Reprints:

Yin Aphinyanaphongs

Department of Biomedical Informatics

4th Floor Biomedical Eskind Library

2209 Garland Avenue

Vanderbilt University

Nashville, TN 37232

Other Communication:

Name: Same as above

Address: Same as above

Telephone Number: 615-936-1556

Fax Number: 615-936-1427

E-mail Address: ping.pong@vanderbilt.edu

Table of Contents

Section 1: Mathematical Descriptions of Learning Methods

1. Log frequency wih Redundancy [1]

The number of occurrences of term w_k in document t_i is denoted by $f(w_k, t_i)$ and $f(w_k) = \sum_i f(w_k, t_i)$ is the number of occurrences of term w_k in the whole document collection. The vector of logarithmic type frequencies of document t_i is defined by:

$$\mathbf{l}_i = (\log(1 + f(w_1, d_i)), ..., \log(1 + f(w_1, d_i))) .$$

The weights of each term w_k are defined by:

$$r_k = \log N + \sum_{i=1}^{N} \frac{f(w_k, d_i)}{f(w_k)} \log \frac{f(w_k, d_i)}{f(w_k)}$$

where $f(w_k, d_i)$ is the frequency of occurrence of term w_k in document t_i and N is the total number of documents in the collection. The weights are combined into a vector for the whole document collection:

$$\mathbf{r} = (r_1, ..., r_n)$$

The final scheme used is defined by:

$$\mathbf{x}_i = \frac{\mathbf{l}_i * \mathbf{r}}{\|\mathbf{l}_i * \mathbf{r}\|_{L_2}}$$

where the "$*$" corresponds to multiplication of vectors, and L_2 refers to the normalization of the vector multiplication.

2. Naïve Bayes [2]

The Naïve Bayes classifier for text classification [3] is a model that estimates the probabilities of the class c_j given the terms $\mathbf{w}$ by using the training data to determine parameters. The classification can be described as:

$$C_{learned} = \underset{c_j \in C}{\operatorname{argmax}}\ P(c_j \mid w_1, w_2, w_3, ...) \tag{0.1}$$

where C is the set of classes, c_j is one class in the set of classes, $\mathbf{w} = (w_1, w_2, ..., w_n)$ is the vector composed of individual words, and $C_{learned}$ is the max a posteriori class as predicted by the Naïve Bayes classifier. Bayes theorem can be used to rewrite the expression as

$$\begin{aligned} C_{learned} &= \underset{c_j \in C}{\operatorname{argmax}}\ \frac{P(w_1, w_2, w_3, ... \mid c_j) P(c_j)}{P(w_1, w_2, w_3, ...)} \\ &= \underset{c_j \in C}{\operatorname{argmax}}\ P(w_1, w_2, w_3, ... \mid c_j) P(c_j) \end{aligned} \tag{0.2}$$

where $P(c_j) = \frac{N_c}{N}$, N_c is the number of documents in the category, and N is the total number of documents are before.

Because the denominator scales each category equally, it is left out. In order to solve this final equation, the term $P(w_1, w_2, w_3, ... \mid c_j)$ would have to be estimated. This estimate would require very large datasets to calculate accurately. A simplifying assumption is made that each term is conditionally independent given the class value. The final equation becomes

$$C_{learned} = \underset{c_j \in C}{\operatorname{argmax}} P\left(c_j\right) \prod_i P(w_i \mid c_j) \tag{0.3}$$

where $P(w_i \mid c_j)$ is often estimated using

$$P(w_i \mid c_j) = \frac{1 + N_{ij}}{T_c + T}. \tag{0.4}$$

T_c is the total number of words in all training examples whose target values is c, N_{ij} is the number of times word *i* occurs within documents of class *c*, and T is the total number of words in the training data.

For the text categorization task, the simplifying assumption is that the probability of any word occurring in a document is independent of whether it occurs once one knows the document class. This assumption does not hold for all possible document sets. Nevertheless, this assumption is made to make the calculations of probabilities tractable. The results to date have proven to be good even when the independence assumption is violated [3-5]. Domingos and Pazzani give theoretical explanation for the good performance. Specifically, they show that the classifier can have optimal performance under zero-one loss for many target functions even though it may be sub-optimal under a squared error loss function [6].

3. Boostexter (AdaBoost.MR) [7]

Adaboost.MR is defined as follows:

1. Given $(\mathbf{d}_1, c_1), ..., (\mathbf{d}_m, c_m)$ where $d_i \in X, c_i \in \{-1, +1\}$.
2. Initialize distribution $D_1(i, l_0, l_1) = \begin{cases} 1/(m \cdot |Y_i| \cdot |Y - Y_i|) & \text{if } l_0 \notin Y_i \text{ and } l_1 \in Y \\ 0 & \text{else.} \end{cases}$.
3. For count $t = 1..T$:
    a. Train the weak learner using distribution D_t.
    b. Get weak hypothesis $h_t : X \times Y \to R$
    c. Choose $\alpha_t \in R$
    d. Update:

$$D_{t+1}(i, l_o, l_1) = \frac{D_t(i, l_0, l_1) \exp\left(\frac{1}{2}\alpha_t (h_t(x_i, l_0) - h_t(x_i, l_1))\right)}{Z_t}$$

    where Z_t is a normalization factor so that D_{t+1} is a distribution.

4. Output final hypothesis

$$f(x, l) = \sum_{t=1}^{T} \alpha_t h_t(x, l)$$

The goal of Adaboost.MR is to find a function that minimizes the number of misorderings so that the labels in Y are ranked above the labels not in Y. The function fails to rank l_1 above l_0 for a crucial pair l_0, l_1 if $f(x, l_1) \le f(x, l_0)$.

The algorithm denotes the weight for instance $\mathbf{d}_i$ and the pair l_0, l_1 by $D_t(i, l_0, l_1)$. The distribution is zero except for triples (i, l_0, l_1) for which l_0, l_1 is a crucial pair.

The weak learner has the form $h_t : X \times Y \to R$. In the Boostexter implementation, it is a one-level decision tree that outputs predictions for the article being in the class based on the word being present or absent.

4. Support Vector Machines [8]

For a binary classification problem with data points $\mathbf{x}$ with labels $y \in \{1, -1\}$, the equation for the separating hyperplane is:

$$\mathbf{w} \bullet \mathbf{x} - b = 0 \tag{0.5}$$

where $\mathbf{w}$ is a vector perpendicular to the separating hyperplane, and b is a constant. The quadratic classification rule for a linear, soft margin support vector machine is to solve the quadratic program:

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}\left\|w^2\right\| + C\sum_i \xi_i \\ s.t.: \quad & y_i(w \cdot x_i + b) \geq 1 - \xi_i, \ \forall x_i \\ & \xi_i \geq 0 \end{aligned} \tag{0.6}$$

where minimizing $\min \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$ maximizes the margin between the supporting planes with a cost C applied to the summation of the Euclidian distance of any misclassified training examples ξ_i

Quadratic programming techniques are used to solve this problem. Using Lagrange multipliers and duality [9], the quadratic problem becomes:

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j x_i \bullet x_j - \sum_{i=1}^{m} \alpha_i$$
$$\text{where } \sum_{i=1}^{m} y_i \alpha_i = 0, \alpha_i \geq 0, i = 1,..,m \tag{0.7}$$

As discussed the linear problem solution is extended to the linearly non-separable case by mapping the input space to (so called) "feature" space via a mapping ("kernel") function so that the classes are linearly separated in feature space. The modification of the Lagrangian equation is to introduce a kernel represented by *K*.

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j K(x_i, x_j) - \sum_{i=1}^{m} \alpha_i \tag{0.8}$$
$$\text{where } \sum_{i=1}^{l} y_i \alpha_i = 0, C \geq \alpha_i \geq 0, i = 1,..,m$$
$$\text{and } K(x_i, x_j) = \theta(x_i) \bullet \theta(x_j)$$

Some common kernels used include polynomial, RBF, and 2 layer neural network kernels. The polynomial kernel used in this work is

$$K(x_i, x_j) = (x_i \bullet x_j + 1)^d \tag{0.9}$$

As discussed previously, addition of kernel modifications allows non-linear solutions by mapping to a feature space where features are linearly separable.

Section 2: Additional Experiments

Conversion to Boolean Queries

The black box nature of our methods prevents a direct conversion of the generated models to Boolean queries. In [10], we have explored a feature selection/ decision tree conversion method and compared the performance of the resulting Boolean queries to the treatment and etiology models created in this study.

The Boolean queries were generated through a combination of feature selection and decision trees in the treatment and etiology categories. First, we applied SVM and Markov Blanket based feature selection algorithms [11]. Then, with the resulting feature sets from each algorithm, we applied decision trees using the gini index of diversity to rank the relevant features. For the treatment category, the important words for methodological classification were “publication type randomized controlled trial” at the top node with “publication type meta analysis” and “treatment” as second level nodes. Similarly for etiology, the top node is the stemmed title word “title_mortal,” and the second level nodes are “mh_Risk Factors” and “95.” The decision trees do not perform as well as the polynomial SVM models presented in this paper [12].

These decision trees are easily extendable for use in Boolean based search engines such as PubMed [10]. The example tree for the treatment category is illustrated in Figure 1. The triangles are decision nodes. The left branch corresponds to the word being absent, and the right branch to the word being present. The leaf values indicate the probability of a high quality article. Each leaf is a possible query to a Boolean based search engine. For example in Figure 1, the Boolean query of the rightmost leaf is "randomized controlled trial" [PTYP] OR "treatment" [WORD].

We used the same procedure for generating Boolean queries for the etiology category. In contrast to the relatively small treatment Boolean tree, the etiology tree is complex with depths up to 7 levels (Figure 2 and labels in Table 1). Nevertheless, the method to generate Boolean queries was the same. Sometimes the Boolean query process we outlined here returns simple trees and query sets and other times (depending on the complexity of the target function) more complex query sets.

Section 3: Ranked Retrieval Effectiveness Supplement

Ranked retrieval effectiveness curves are a novel method of comparing learning method performances and illustrating recall and precision metrics. The curves show recall and precision percentages limited to the first *N* returned articles. We chose an *N* of 100 articles because most information seekers are unwilling to look beyond the first 100 documents for relevant results [13].

The first graph shows the recall at X returned documents. The curve gives a user an indication of the number of high quality articles out of all high quality articles in the corpus that are returned in the set of documents. The recall curve is monotonically increasing, and the best retrieval method has, in general, the steepest slope. The curve will eventually converge to 100% recall when all high quality articles are in the returned set.

The second graph shows the precision at X returned documents. The graph shows the percentage of high quality articles in X returned documents. The precision curve is *not* monotonically decreasing, but does eventually converge to 0 as the proportion of ACP- to ACP+ returned articles increases at lower ranks. In general, the best retrieval method will retain the highest precision as more documents are returned (i.e. the flattest slope). Both recall and precision ranked retrieval graphs are specific to this corpus and are a method to compare the learning methods.

Figures 3 and 4 show the recall and precision ranked retrieval effectiveness graphs for each learning method in each category. The curves were averaged over all cross-validation sets in each category.

For the recall graphs in Figure 3, the polynomial SVM performed the best in most categories, except in treatment where the linear and polynomial SVMs perform comparably. In the treatment and etiology tasks, inspecting the top 100 gave 58% (44 out of 76 articles) and 56% (23 out of 41 articles) recall respectively. In the sample size limited prognosis and diagnosis categories, the top 100 gave 38% (6 out of 15 articles) and 67% (13 out of 20 articles) respectively. Considering the size of the returned article sets, the high recall percentages in the first 100 articles were promising. This ranked retrieval recall analysis was further shown at 50 articles returned in Table 2.

For the precision graphs (Figure 4), the linear and polynomial SVMs performed similarly across all categories. The precision graphs are sensitive to the priors for high quality articles in each category. In the prognosis and diagnosis categories where the priors are low, the precision rapidly fell as more articles were inspected. Inspecting 100 documents, for the prognosis category, 1 out of 20 documents, and similarly, for the diagnosis category, 1 out of 10 articles were high quality. In etiology and treatment where the priors are high, 1 out of 2 articles and 1 out of 4 articles were high quality respectively.

A natural next step in analyzing the top ranked articles is to inspect the false positives and see if they match the ACP inclusion criteria. In this paper, we do not address this analysis. A more thorough analysis of the actual ranked articles would be an interesting extension to this work.

Table 1: Map for node values of etiology tree in Figure 2.

X1	associ
X2	mh Risk Factors
X3	title mortal
X4	95
X5	title meta
X6	killip
X7	drinker
X8	phentermin
X9	mh Sick Role
X10	mh Autoimmunity
X11	homocyst
X12	mh Smoking Cessation
X13	mh Weather

Table 2 – Ranked Retrieval Performance at 50 and 100 returned articles

Category	Learning Method	Recall At 50 Articles Returned	Recall At 100 Articles Returned	Total Positive Articles In Each Set	Total Articles in Each Set
Treatment	Linear SVM	36% (27 articles)	58% (44 articles)	76	3157
	Polynomial SVM	35% (27 articles)	58% (44 articles)	76	3157
Etiology	Linear SVM	34% (14 articles)	50% (20 articles)	41	3157
	Polynomial SVM	36% (15 articles)	56% (23 articles)	41	3157
Prognosis	Linear SVM	33% (5 articles)	38% (6 articles)	15	6988
	Polynomial SVM	33% (5 articles)	38% (6 articles)	15	6988
Diagnosis	Linear SVM	55% (11 articles)	63% (13 articles)	20	6988
	Polynomial SVM	56% (11 articles)	67% (13 articles)	20	6988

Table 3 - ACP selection criteria used to define/identify high quality articles in the corpus [14].

Treatment	• random allocation of participants to comparison groups. • follow-up (endpoint assessment) of at least 80% of those entering the investigation. • outcome measure of known or probable clinical importance.
Diagnosis	• inclusion of a spectrum of participants, some but not all of whom have the disorder or derangement of interest. • objective diagnostic ("gold") standard (e.g., laboratory test not requiring interpretation) OR current clinical standard for diagnosis (e.g., a venogram for deep venous thrombosis), preferably with documentation of reproducible criteria for subjectively interpreted diagnostic standard (i.e., report of statistically significant measure of agreement beyond chance among observers). • each participant must receive both the new test and some form of the diagnostic standard. • interpretation of diagnostic standard without knowledge of test result. • interpretation of test without knowledge of diagnostic standard result.
Prognosis	• inception cohort of individuals, all initially free of the outcome of interest. • follow-up of at least 80% of patients until the occurrence of a major study endpoint or to the end of the study.
Etiology	• exploration of the relation between exposures and putative clinical outcomes. • prospective data collection with clearly identified comparison groups for those at risk for the outcome of interest (in descending order of preference from randomized controlled trial, quasi-randomized controlled trial, nonrandomized controlled trial, cohort studies with case-by-case matching or statistical adjustment to create comparable groups, to nested case-control studies. • masking of observers of outcomes to exposures (criterion assumed to be met if outcome is objective, i.e., all-cause mortality, objective test).

Table 4 – Journals Reviewed by the ACP journal club and used in this study [14].

Age and ageing	Hypertension
American Journal of Cardiology	Journal of the American Board of Family
American Journal of Epidemiology	Physicians
American Journal of Medicine	Journal of the American College of
American Journal of Public Health	Cardiology
American Journal of Respiratory and	Journal of the American Geriatrics Society
Critical Care Medicine	Journal of the American Medical
Annals of Emergency Medicine	Informatics Association
Annals of Internal Medicine	Journal of Clinical Epidemiology
Annals of Medicine	Journal of Family Practice
Archives of Family Medicine	Journal of General Internal Medicine
Archives of Internal Medicine	Journal of Infectious Diseases
Archives of Neurology	Journal of Internal Medicine
Arthritis and Rheumatism	Journal of Neurology, Neurosurgery, and
British Medical Journal	Psychiatry
British Journal of General Practice	Journal of Vascular Surgery
Canadian Medical Association Journal	Journal of the American Medical
Canadian Journal of Cardiology	Association
Canadian Journal of Gastroenterology	Lancet
Chest	Medical Care
Circulation	Medical Journal of Australia
Clinical and Investigative Medicine	New England Journal of Medicine
Critical Care Medicine	Neurology
Diabetes Care	Pain
Gastroenterology	Spine
Gut	Stroke
Heart	Thorax

Table 5: Feature Sets with and without MeSH terms/publication types AUC.

Category	Feature Set	Average AUC	Min AUC	Max AUC	Significant Difference at 0.001 level (Delong)
Treatment	Title + Abstract	0.971	0.965	0.978	
	Title + Abstract + MeSH + Publication Types	0.973	0.962	0.979	No
Etiology	Title + Abstract	0.934	0.891	0.954	
	Title + Abstract + MeSH + Publication Types	0.937	0.892	0.953	No
Prognosis	Title + Abstract	0.913	0.870	0.936	
	Title + Abstract + MeSH + Publication Types	0.911	0.871	0.946	No
Diagnosis	Title + Abstract	0.955	0.944	0.967	
	Title + Abstract + MeSH + Publication Types	0.959	0.947	0.980	No

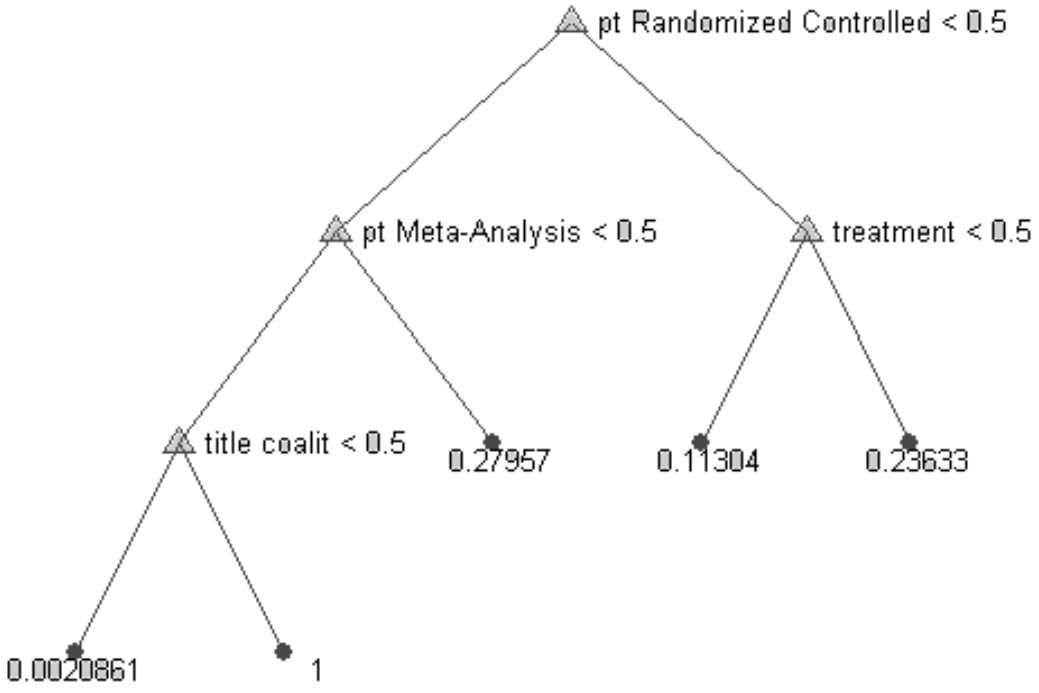

Figure 1: Treatment Category Decision Tree
Words are processed according to the algorithm in the Section 3 of the main paper. All words are stemmed. The terminal nodes are labeled with the predicted value for that node based on the training items.

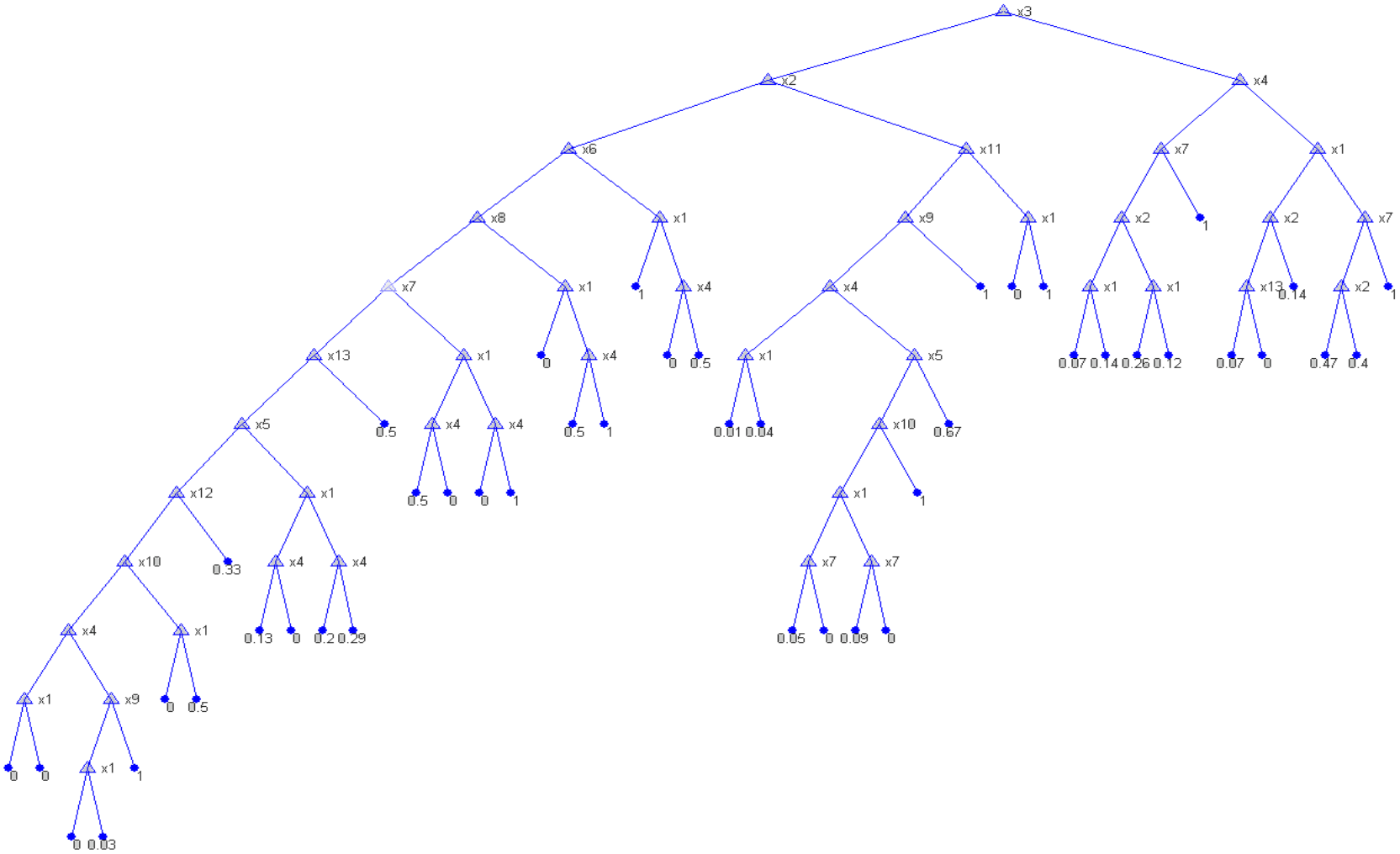

Figure 2: Etiology Category Decision Tree.
Words are processed according to the algorithm in the Section 3 of the main paper. All words are stemmed. The terminal nodes are labeled with the predicted value for that node based on the training items. The label for each node is located in Table 6.

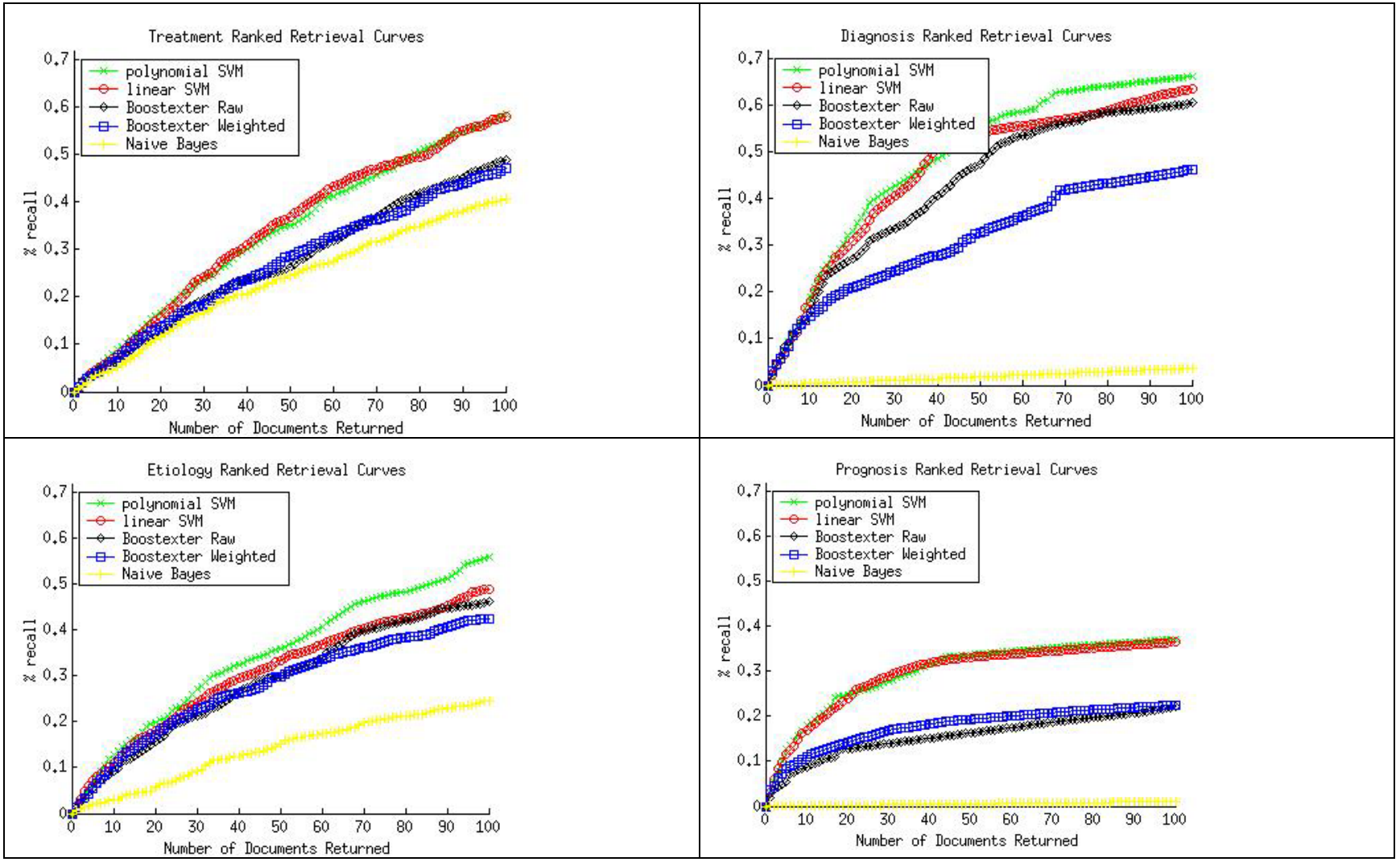

Figure 3 – Ranked Retrieval Effective Curves – The comparison to the query filters is not shown in these graphs due to the limited range of the x-axis. Refer to Table 2 for comparison to the number of returned documents at a given recall level.

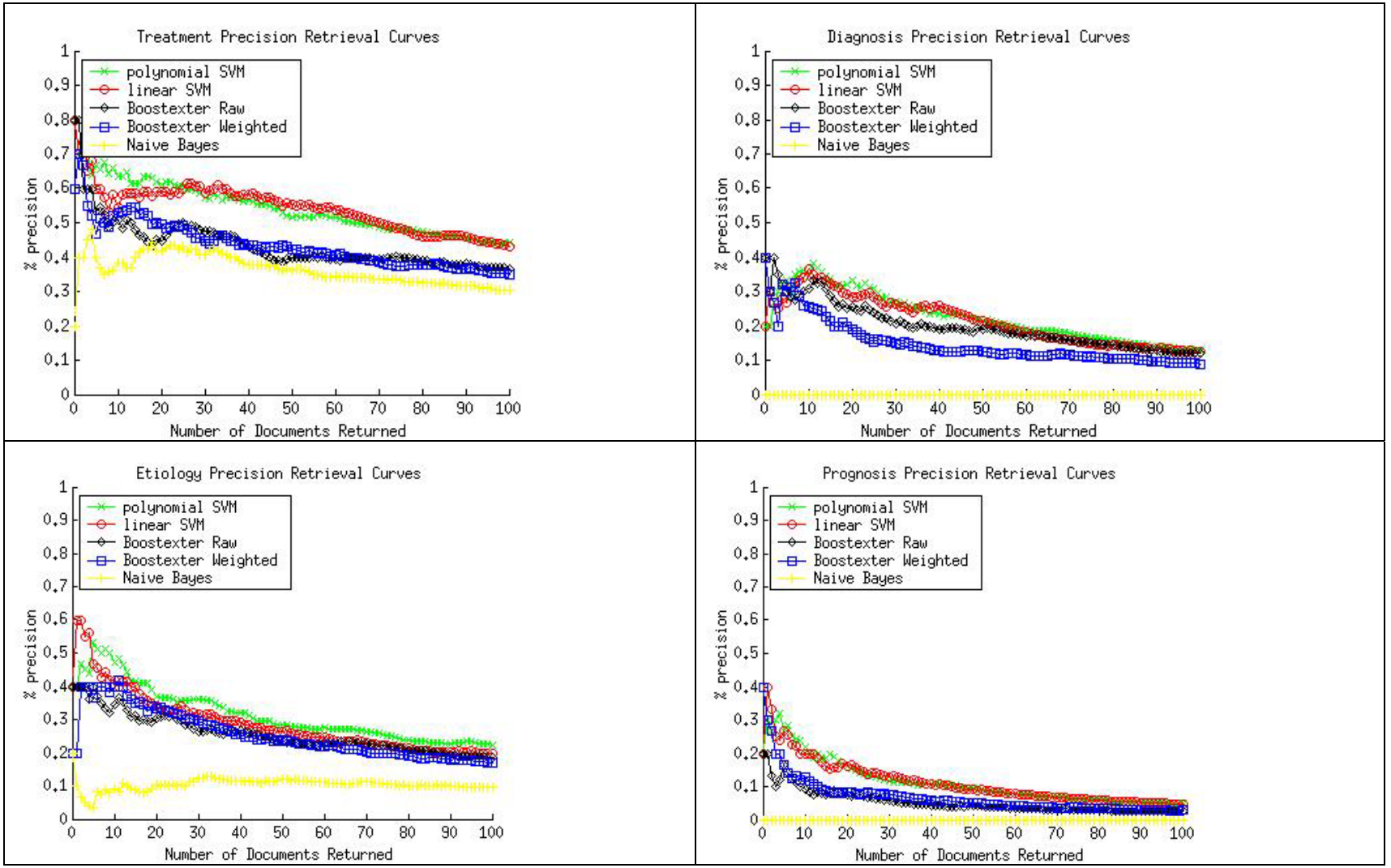

Figure 4 – Precision Retrieval Effectiveness Curves. Depiction of how precision changes as the number of documents returned changes.

References

1. Leopold, E. and J. Kindermann, *Text Categorization with Support Vector Machines. How to Represent Texts In Input Space?* Machine Learning, 2002. **46**: p. 423-444.

2. Mitchell, T.M., *Machine learning*. 1997, New York: McGraw-Hill. xvii, 414.

3. Joachims, T. *A probabilistic analysis of the Rocchio Algorithm With TFIDF for text categorization*. in *14th International Conference on Machine Learning*. 1997. Nashville, TN: Morgan Kauffman.

4. Yang, Y. and J. Pederson. *Feature selection in statistical learning of text categorization*. in *14th International Conference on Machine Learning*. 1997. Nashville, TN: Morgan Kauffman.

5. Lewis, D.D. and M. Ringuette. *A comparison of two learning algorithms for text categorization*. in *Proceedings of Third Annual Sympsium on Document Analysis and Information Retrieval*. 1994.

6. Domingos, P. and M. Pazzani, *On the Optimality of the Simply Bayesian Classifier under Zero-One Loss.* Machine Learning, 1997. **29**: p. 103-130.

7. Schapire, R.E. and Y. Singer, *Boostexter: A Boosting-based System for Text Categorization.* Machine Learning, 2000. **39**(2/3): p. 135-168.

8. Cristianini, N. and J. Shawe-Taylor, *An Introduction to Support Vector Machines and other Kernel-based Learning Methods*. 2000: Cambridge University Press.

9. Vapnik, V., *Statistical Learning Theory*. 1998: Wiley.

10. Aphinyanaphongs, Y. and C.F. Aliferis. *Learning Boolean Queries for Article Quality Filtering*. in *MEDINFO*. 2004. San Francisco, CA.

11. Aliferis, C., I. Tsamardinos, and A. Statnikov. *HITON: A Novel Markov Blanket Algorithm for Optimal Variable Selection*. in *Proceedings AMIA Symposium*. 2003. Washington DC.

12. Aphinyanaphongs, Y. and C.F. Aliferis. *Text Categorization Models for Retrieval of High Quality Articles in Internal Medicine*. in *Proceedings AMIA Symposium*. 2003. Washington DC.

13. http://www.iprospect.com/media/press2004_04_20.htm

14. *Purpose and Procedure.* ACP Journal, 1999. **131**(1): p. A-15 - A-16.