Knowledge Discovery for Document Classification

Chidanand Apté

IBM Research Division
T.J. Watson Research Center
Yorktown Heights, NY 10598

Fred Damerau

IBM Research Division
T.J. Watson Research Center
Yorktown Heights, NY 10598

Sholom Weiss

Rutgers University
Dept. of Computer Science
New Brunswick, NJ 08903

Abstract

We report on extensive experiments using rule-based induction methods for document classification. The goal is to automatically discover patterns in document classifications, potentially surpassing humans who currently read and classify these documents. By using a decision rule model, we induce results in a form compatible with expensive human engineered systems that have recently demonstrated excellent performance. Using computer-intensive rule induction methods, we have conducted experiments over a vast set of document families, including UPI, Reuters, NTIS, and the Library of Congress Catalog. We report on several approaches to classic problems for such applications, including choosing the right representation for text, and handling high dimensionality.

1 Introduction

Assigning classifications to documents is essential for management of knowledge and the subsequent retrieval of documents. Document classifications are currently assigned by humans who read the documents and have some prior experience with the expected topics. In any large organization, huge volumes of text information are examined, and these documents must be categorized in some fashion.

Assigning subject classification codes to documents manually is time consuming and expensive. Rule-based methods for assigning subject codes, while relatively effective, require manual construction of assignment rules, which is also expensive. This report presents preliminary results on experiments to derive the assignment rules automatically from samples of the text to be classified. In many carefully organized text storage and retrieval systems, texts are classified with one or more codes chosen from a complete classification system. Examples include the NTIS (National Technical Information Service) documents from the US government, news services like UPI (United Press International) and Reuters, publications like the ACM (Association for Computing Machinery) Computing Reviews and many others. Recent work has shown that in certain environments, rule based systems can do code assignment quickly and accurately [3, 4]. However, the rule sets must be constructed by hand for each application, a time consuming and expensive process. On the other hand, machine learning methods provide an interesting alternative for automating the rule construction process.

One of the earliest non-numerical problems to which computers were applied was the storage and retrieval of machine-readable documents, even though access to files was serial from magnetic tape. The major problem in document retrieval is determining from a representation of a query and a representation of a document whether the document is relevant to the query. This determination is inherently imprecise, since experienced people can differ on their judgments with respect to the same document and query pair, even with the whole document available and a considerable range of background information on which to draw. The document and query representations available to computer programs are much less rich, and the results are therefore less precise. Nevertheless, since the number of documents of potential interest to a human searcher far exceeds what one could hope to read, the search for better computer representations continues.

One way that has been used to limit a search to relevant topics is to assign one or more subject codes from a predetermined list to each document added to the storage system. An example is the Dewey decimal classification or Library of Congress classification used in libraries to store and retrieve books. There is a large number of such classification systems in common use for documents shorter than books. One of the main drawbacks of this procedure is the cost in time and money of human assignment of classification codes. An example which has attracted recent attention in the artificial intelligence and machine learning communities is the assignment of subject codes to the news stories produced by wire services such as Reuters, the Associated Press and United Press International. One reason these are so attractive to researchers is that there is a large set of examples which are already coded by subject, so that discovery procedures for mechanically assigning codes can use very large training sets.

A well known example of an expert system for this task is the CONSTRUE system [4] used by the Reuters news service. This is a rule based expert system using manually constructed rules to assign subject categories to news stories, with a reported accuracy of over 90% on 750 test cases [3]. An example of a machine learning system for the same task is a system built by Thinking Machines Corporation for Dow Jones. This system uses a paradigm called Memory Based Reasoning, which is in the category of nearest neighbor classifiers. It has a reported accuracy in the range of 70-80%, [9].

In considering the problem of categorizing documents, the rule based approach has considerable appeal. While weighted solutions such as the linear probabilistic methods used in [7] or nearest-neighbor methods may also prove reasonable, their weak interpretable capabilities makes them less desirable for this application. Human engineered systems have been successfully constructed using rule-based solutions. Thus it would be most useful to continue with a model that is compatible with human expressed knowledge. Because of the parsimonious and interpretable nature of decision rules, we can readily augment our knowledge or verify the rules by examining related categorized documents. In the remainder of this paper, we describe our approach to automating the discovery of text categorization knowledge.

We note that research on classification in the information retrieval community has had a different focus [13]. The emphasis there has been on the discovery of classification structures, rather than on the assignment of documents within an existing classification scheme.

2 Attribute Selection

The selection of attributes for document classification is quite complex. Two kinds of document content can be distinguished, the bibliographic part found in scientific articles, which describes the external context of a document, e.g., author, publisher, conference at which it was presented, etc., and the text part, which is intrinsic to all documents. While the external content can often provide useful identifying attributes, this is generally not true of newswire text such as Reuters or UPI. Consequently, we and others have chosen to focus only on the textual part. This still leaves a wide choice for potential attributes.

Document retrieval systems are supposed to chose those documents which are *about* some concept. However, documents do not have concepts, but rather *words*. Words clearly do not correspond directly to concepts. Some words are used for more than one concept, e.g., "bank" as a financial institution and "bank" as part of a river. Some concepts require more than one word for their designation, e.g, the football player "running back", and most concepts can be referenced by more than one word or phrase, e.g., "doctor" and "physician". Humans are relatively good at inferring concepts from the words of a document. To do this, they bring to bear vast knowledge of the grammar of the language and of the world at large. Very little of this knowledge is available to a computer system, in large part because we have only sketchy and incomplete methods for organizing or inferring such information automatically. Programs for parsing sentences and representing their semantic content in some formal language, e.g., first order logic, often fail and are sometimes wrong when they do not fail. On even simpler tasks, like deciding whether a particular use of the word "bear" is to be taken as a noun or verb, or which "bank" is being referred to, sophisticated parsing systems are far from error-free.

Current research on text categorization supports the efficacy of the simpler schemes for attribute selection [7]. We chose to use a simple attribute selection method. A program scanned the text portion of the documents to produce a list of single words, and of word pairs in which neither word belonged to a defined list of function words or was a number or part of a proper name, (a proper name was identified as a succession of capitalized words, a method also prone to error). The statistical attribute selection techniques should in principle remove function words and pairs containing them in any case, since they are poor predictors of content, but removing them ahead of time reduces an already very large computational task. Following accepted practice in statistical language processing, [2], we then immediately eliminated any word or pair which occurred less than five times. These two lists are similar to those identified by Lewis, [7], for words and phrases.

Preliminary experiments indicated that using only pairs as attributes gave poor results in general, mainly because relatively few attributes were assigned to each document. This is a result of pair frequencies being much lower than single word frequencies. In [6], a more sophisticated phrase selection method was used, and the same conclusion was reached. While single words alone were quite successful, there were cases where including pairs as attributes gave better results. Most of our experiments have been with attribute sets consisting of both single words and pairs. The single word and the pair lists were merged, sorted by frequency, and those terms in the most frequent 10,000 retained. The list was further reduced by eliminating the bottom ranking terms if not all of the terms of that frequency were in the set of the 10,000 most frequent and by eliminating all function words. As a result, our basic attribute list started with approximately 10,000 attributes. Our first experiments were run with the full attribute set. Later, in order to process more cases, we experimented with attribute pruning methods.

Choosing the right attribute set to represent a document is critical to successful induction of classification models. The attribute selection process we just described creates a dictionary of terms for a document family. Each individual document in the family can then be characterized by a set of features that are boolean indicators denoting whether a term from the dictionary is present or absent in the document. The key to getting good performance from an induction run is the fine tuning of the dictionary creation process for a document family, and the feature creation process

.. Tokheim Corp. has announced the formation of a wholly owned environmental subsidiary in response to new U.S. Environmental Protection Agency regulations on underground storage tanks ..

announce
formation
wholly
own
wholly own
environmental
subsidiary
response
regulation
underground
storage
tank
storage tank

Figure 1: Example of a text fragment and corresponding attribute list

for individual documents. Alternatives for the basic methodology that we have described include creating localized individual dictionaries for each classification to be induced (by using as input only those members of the document family that belong to the classification to be learned), and using frequency count features (that denote how many times a term from the dictionary is present in a document) instead of boolean features.

3 Inducing the Rule Sets for a Document Category

The text retrieval classification rule induction system consists of:

- A preprocessing step for determining the values of the attributes used to describe text and selecting the category of a document from a defined set of categories.
- An induction step for finding rule sets which distinguish categories from one another.
- An evaluation step for choosing the best rule set, based on minimizing the classification error.

The initial task is to produce a set of attributes from samples of text of labeled documents. The attributes are single words or word phrases, and the values are either binary, i.e., the attribute appears in the text or does not, or are numbers which are derived from the frequency of occurrence in the text being processed and the frequency in the sample as a whole. An example of some UPI text, dated 12/01/88, and the attributes of single words and pairs which might be generated from this text are illustrated in Figure 1.

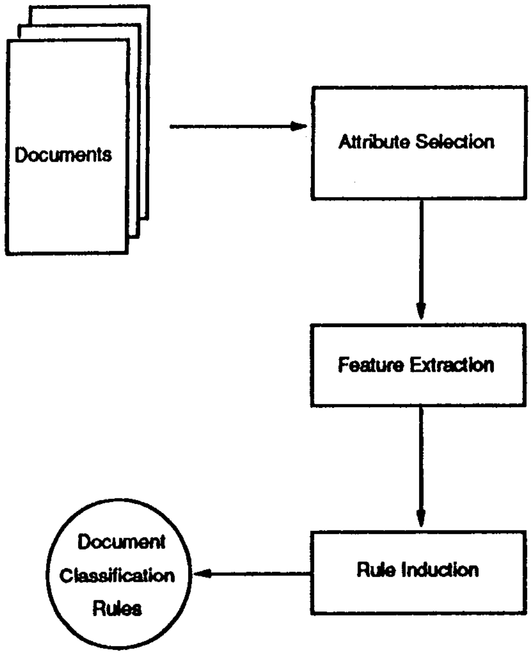

Figure 2: Organization of Document Classification Knowledge Discovery Apparatus

Given an attribute list, sample cases can be described in terms of the words or phrases found in the documents. Each case consists of the values of the attributes for a single article. In addition, each case is labeled to indicate the classification of the article it represents. The objective is to compute sets of rules which distinguish one category of text from the others. The best rule set is selected, where best is a set which is both accurate and not excessively long. The output is the preferred rule set for identifying a classification, along with an indication of the expected classification error. This organization of programs is illustrated in Figure 2.

Rule and tree induction methods have been extensively described in published works [1, 12, 10]. For our document indexing apparatus, we have used a rule induction technique called SWAP-1 [5, 11]. Rule induction methods attempt to find a compact covering rule set that completely separates the classes. The covering set is found by heuristically searching for a single best rule that covers cases for only one class. Having found a best conjunctive rule for a class C, the rule is added to the rule set, and the cases satisfying it are removed from further consideration. The process is repeated until no cases remain to be covered. Unlike decision tree induction programs and other rule induction methods, SWAP-1 has an advantage in that it uses optimization techniques to revise and improve its decisions. Once a covering set is found that separates the classes, the induced set of rules is further refined by either pruning or statistical techniques. Using train and test evaluation

running back ⟶ football article
...
kicker ⟶ football article
injure reserve ⟶ football article
...
award & player ⟶ football article
...

	TRAINING CASES	
	Football	Not Football
Football	151	10
Not Football	0	1081

	TEST CASES	
	Football	Not Football
Football	135	26
Not Football	12	1069

Figure 3: Example of induced rule set and estimated performance measures

methods, the initial covering rule set is then scaled to back to the most statistically accurate subset of rules.

For the document classification application, SWAP-1 induces rules that represent patterns, i.e. combinations of attributes, that determine the most likely class for an article. A result of applying SWAP-1 to a training set of cases results in a set of rules, and the associated error rates on the training as well as test samples. An illustration of this appears in Figure 3.

4 Handling High Dimensionality

The high dimensionality issue occurs along three major axes; the number of documents in a document database, the size of the dictionary, and the number of mutually exclusive classes for which classification models have to be learned. Usually, we have had access to hundreds of thousands of documents for training purposes. Clearly, these many samples pose a problem to any typical in-core rule induction process. We have chosen to use a random sampling process to extract a representative subset for the training cycle. Repeated tests of induced models with the full database have demonstrated that the estimated error rates remain consistent. For a typical document database, we have encountered hundreds of classes for which classification models need to be learned. While most rule induction systems conceptually permit the simultaneous learning of multiple classification models, we have obtained stronger results by treating the training problem as a series of dichotomous classification induction problems. That is, we iterate over the entire class set, learning at a time a classification model for a particular class with all the documents not in that class being grouped into a single negative instance set.

Finally, the more serious dimensionality problem lies with the dictionary size, which can be in the tens of thousands. Clearly, these many features pose a severe computational problem to any rule induction apparatus. We have experimented with two primary methods for controlling the size of the feature space. One has been to couple, as a pre-processor to SWAP-1, a conventional feature extraction algorithm based on the information entropy metric, as used in decision tree constructions [1, 12, 10]. Typically, using this pre-processor has resulted in the pruning down of the feature set to 100th its original size. For example, from a 10,000 sized attribute set, the pre-processor would choose anything from 30 to 150 features that provided complete discrimination.

Another more radical approach we have tested is to sort an attribute set by frequency (in descending order), and choose the top n features for a specified topic. For example, one could take a sorted list of attributes (in descending frequency) and choose the top 100, or 1000, or 2500 features for training the classification model.

We have consistently observed that results for these two approaches are within a couple of percentage points of each other. In addition, the induced rule sets for the two approaches do not vary much in complexity either (number of rules and terms per rule). Discovering the optimal rule set that can be used for classification is a heuristic optimization process, and the SWAP-1 approach to rule induction (influenced by heuristic optimization techniques [8]) seems to perform robustly across these radically different methods for cutting the feature space. An explanation to this phenomena could be the observation that the information entropy based feature extractor, after doing all its computations, would produce a feature set that would mostly (as high as 80-90%) consist of highly frequent attributes. These are features that would most certainly list in the top couple of deciles of a frequency sorted dictionary. Taking the second (occam's razor) approach therefore is not at all as penalizing as we had initially suspected.

Given that the first approach adds computational time to the overall induction process, while the second one is practically free, one can argue for the preference of the second approach over the first. However, for specific applications where we are trying to maximize performance, the extra computational time taken by feature extraction is worth a try.

5 Experimental Results

In order to come to some conclusions regarding the general applicability of this technique, we have run experiments on a number of very large data bases, including scientific abstracts originating from the National Technical Information Service, library catalogue records representing the holdings of the IBM libraries, a 1988 sample of the UPI newswire, and a 1987 sample of the Reuters newswire, properly identified as Reuters-22173, but hereafter referred to as "Reuters"[1]. We have not attempted to exhaustively cover any of these databases, but rather to run what appeared to us to be reasonable sample problems.

To provide an initial basis for comparison of our results with others, particularly [6, 7], we made a number of runs using the Reuters data. The first step was to generate a list of words and word pairs as described above. A number of the stories did not have subject codes assigned, and the vocabulary from these stories was not included. Given the attribute list, we then generated a feature vector for each acceptable story. Besides the stories with assigned subject, we also eliminated stories

[1] The latter was obtained by anonymous ftp from /pub/reuters on canberra.cs.umass.edu. Free distribution for research purposes has been granted by Reuters and Carnegie Group. Arrangements for access were made by David Lewis.

with less than 10 features and stories which were essentially duplicates of one of the preceding 50 stories (determined by keeping a history table of 50 feature vectors). As a result, our story count is substantially less than that of Lewis, and therefore exact comparisons have not been made.

Our next step was to generate rule sets for the most common 10 topics, (*earnings, acquisitions, foreign exchange markets, grain, crude, corporate news, trade, loan, interest, wheat*). We generated rules using 1) attribute selection from the full feature set, 2) all of the most frequent 1000 attributes, and 3) all of the most frequent 2500 attributes.

Results in this area are measured by *recall* and *precision*. Recall is the percentage of total documents for the given topic that are correctly classified. Precision is the percentage of predicted documents for the given topic that are correctly classified. For an accurate evaluation, performance should be measured on independent test cases. In Figure 3, the recall is 135/(135+26), and precision is 135/(135+12). Because the document topics are not mutually exclusive, document classification problems are usually analyzed as a series of dichotomous classification problems, i.e the given topic vs. not that topic. To evaluate overall performance on many topics, the results are microaveraged, i.e. the tables, such as in Figure 3, are added and the overall recall and precision are computed.

The recall and precision figures for these experiments, using microaveraging are shown in Table 1. In these experiments, over 10,000 cases were used with one third of these cases reserved solely for testing. The recall figures include cases for the category "corporate news", for which our procedure was unable to generate a rule set. This category had a very wide range of subject matter and relatively few examples. For those categories for which rules were generated, recall was about 4% higher.

From UPI, we sampled categories which we thought would exhibit representative results, some good and some bad. For these, averaging does not seem realistic. Results are shown in Tables 2 and 3. We did not try the first 1000 features for these, since we had noted that some good attributes were less frequent than that. These results confirm what one might have expected - more specific categories perform better. We believe that a significant part of the problem of recall for topics like mergers and acquisitions (M/A) results from deficiencies in the pre-assigned codes. Inspection of the false negatives shows a substantial number of cases where it would seem that the M/A code should have been assigned. By contrast, football and hockey stories are easy to identify. We had run a few tests with domain-specific attributes, including one for M/A. We used this M/A attribute set for football, resulting in 68% recall with 91% precision. For these highly specific domains, even skewed attribute sets give positive results.

We wanted to see how the rule induction technique would do in a domain where each item had relatively few attributes. In such a case, we expected to find lower recall, and were more interested in the achievable precision. We obtained the unified catalogue records of the IBM libraries and attempted to infer rules for the first two characters of the Library of Congress classification codes assigned to each volume. This data set is decidedly sub-optimal, being heavily weighted toward science and engineering. We used as attributes only words and word pairs which would be found on a book title page; author, title, publisher, series, and conference name. On a well-represented topic like "physics", we achieved 47% recall with 71% precision. For "linguistics", with only 35 test cases, we had 11% recall with 50% precision, a result which probably signifies nothing.

The NTIS data set available to us consisted of approximately 100,000 abstracts in all areas of technology, classified into one or more of 40 major categories, each of which has from four to twenty subcategories. For testing purposes, we used samples of about 20,000 drawn from the complete set.

Our results with this data base were rather disappointing. Using selected features from the full

Attributes	Recall	Precision
Feature Extraction	73%	82%
Top 1000 Attributes	73%	84%
Top 2500 Attributes	72%	84%

Table 1: Reuters Results

Subject	Recall	Precision
Air Transportation	57%	89%
Football	87%	95%
Hockey	84%	91%
Mergers and Acquisitions	39%	71%

Table 2: UPI Results with Feature Extraction

set of approximately 10,000 features, for the three major classes we tried, the precision was about 75% but the recall was only from 5 to 50%. Using the full set of most frequent 2500 features, the precision was a little worse and the recall a little better. In attempt to see what simple improvement might be made, we chose the rule set with minimum error, ignoring the size of the rule set. In this case, the precision was a little less than 75% and the recall ranged from 18 to 55%. Inspection of the false negatives gave no simple answer for the low recall. The attributes in most cases included some that one would have supposed would be characteristic of the set, but which apparently would also have generated excessive false positives. This data set needs to be investigated further.

6 Concluding Discussion

From these experiments, it appears that rule induction is quite competitive with other machine learning techniques for document classification and may well be better than others which have been tried. Preliminary results on some data sets suggest somewhat stronger results than those reported for weighted methods such as nearest-neighbor case-based reasoning [9], and for some topics, not much worse than those reported for a hand-built expert system [3]. However, this preliminary conclusion can only be supported by rigorous and exacting comparisons. Given the very large volumes of data, and the sometime proprietary nature of documents, it is not surprising that few if any comparisons have been reported in the literature. The 1987 Reuters stories have recently been widely circulated and should prove helpful in objective comparisons.

Machine induced rule based models permit efficient analytical investigations, since rule sets can be inspected and modified easily either by human or machine. This process has been found to be

Subject	Recall	Precision
Air Transportation	52%	80%
Football	80%	96%
Hockey	73%	91%
Mergers and Acquisitions	26%	79%

Table 3: UPI Results with Top 2500 Attributes

useful when attempting to understand why documents get misclassified, and allows experiments with fine-tuning of the induced models. Often, this inspection detects erroneous classifications in the existing document database, i.e., what appears as misclassified documents are actually correct classifications, and there was a human error in assigning the initial classification.

For example, we have begun to understand the reasons for some of the poor results that we have obtained for certain document families (e.g., NTIS), or for certain categories within a document family (e.g., corporate-news in Reuters). A trend that we observe across these experiments is the presence of poor human classification or poorly organized classes in the training data. The NTIS document family was discovered to be widely populated with documents that had incorrect human assignments of topics. In the case of the Reuters newswires, certain classes like corporate-news are by nature very hard to credibly define (a whole spectrum of newswires can fall into this category). It is very hard for any classification algorithm to discover generalizable patterns across this widely variable set.

The explosive growth of electronic documents has been accompanied by an expansion in availability of computing. It is unlikely that such information can be managed without extensive assistance by machine. Some processes once thought of as requiring understanding and reading comprehension may fall to the superior ability of the machine to discover patterns that characterize document topics. Initially, however, machine learning and discovery systems may be combined with human developed systems for document classification. These, in turn, could be conceivably coupled as knowledge filters with commercial tools like newswire and information feeds and alerts to provide superior information retrieval services to the end user.

References

[1] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees.* Wadsworth, Monterrey, Ca., 1984.

[2] K.W. Church and P. Hanks. Word Association Norms, Mutual Information, and Lexicography. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 76–83, 1989.

[3] P. Hayes and S. Weinstein. Adding Value to Financial News by Computer. In *Proceedings of the First International Conference on Artificial Intelligence Applications on Wall Street*, pages 2–8, 1991.

[4] P.J. Hayes, P.M. Andersen, I.B. Nirenburg, and L.M. Schmandt. TCS: A Shell for Content-Based Text Categorization. In *Proceedings of the Sixth IEEE CAIA*, pages 320–326, 1990.

[5] N. Indurkhya and S. Weiss. Iterative Rule Induction Methods. *Journal of Applied Intelligence*, 1:43–54, 1991.

[6] D. Lewis. An Evaluation of Phrasal and Clustered Representations on a Text Categorization Task. In *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 37–50, June 1992. Edited by Nicholas Belkin, Peter Ingwersen, and Annelise Mark Pejtersen.

[7] D. Lewis. Feature Selection and Feature Extraction for Text Categorization. In *Procceedings of the Speech and Natural language Workshop*, pages 212–217, February 1992. Sponsored by the Defense Advanced Research Projects Agency.

[8] S. Lin and B. Kernighan. An efficient heuristic for the traveling salesman problem. *Operations Research*, 21(2):498–516, 1973.

[9] B. Masand, G. Linoff, and D. Waltz. Classifying News Stories using Memory Based Reasoning. In *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 59–65, June 1992. Edited by Nicholas Belkin, Peter Ingwersen, and Annelise Mark Pejtersen.

[10] J.R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, 1993.

[11] S. Weiss and N. Indurkhya. Reduced Complexity Rule Induction. In *Proceedings of the Twelfth IJCAI*, pages 678–684, 1991.

[12] S.M. Weiss and C.A. Kulikowski. *Computer Systems That Learn.* Morgan Kaufmann, 1991.

[13] P. Willett. Recent trends in hierarchic document clustering: a critical review. *Inf. Process. Manage.*, 24(5):577–597, 1988.