Hypertext Categorization using Hyperlink Patterns and Meta Data

Rayid Ghani RAYID.GHANI@CS.CMU.EDU
Center for Automated Learning & Discovery, Carnegie Mellon University, Pittsburgh, PA 15213

Seán Slattery SEAN.SLATTERY@CS.CMU.EDU
Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213

Yiming Yang YIMING.YANG@CS.CMU.EDU
Language Technologies Institute & Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213

Abstract

Hypertext poses new text classification research challenges as hyperlinks, content of linked documents, and meta data about related web sites all provide richer sources of information for hypertext classification that are not available in traditional text classification. We investigate the use of such information for representing web sites, and the effectiveness of different classifiers (Naive Bayes, Nearest Neighbor, and FOIL) in exploiting those representations. We find that using words in web pages alone often yields suboptimal performance of classifiers, compared to exploiting additional sources of information beyond document content. On the other hand, we also observe that linked pages can be more harmful than helpful when the linked neighborhoods are highly "noisy" and that links have to be used in a careful manner. More importantly, our investigation suggests that meta data which is often available, or can be acquired using Information Extraction techniques, can be extremely useful for improving classification accuracy. Finally, the relative performance of the different classifiers being tested gives us insights into the strengths and limitations of our algorithms for hypertext classification.

1. Introduction

As the size of WWW expands rapidly, the need for good automated hypertext classification techniques becomes more apparent. The Web contains over a billion pages inter-connected via hyperlinks, making the task of locating specific information on the Web increasingly difficult. Chen and Dumais (2000) conducted a study which showed that users often prefer navigating through directories of pre-classified content, and that providing a categorical view of retrieved documents enables them to find more relevant information in a shorter time. The common use of category hierarchies for navigation support in Yahoo! and other major web portals has also demonstrated the practical needs for hypertext categorization.

Hypertext classification poses new research challenges because of the rich representation of a "document" and the connectivity between documents. Hyperlinks, content of linked documents, and meta-data (such as relational attributes) about related web sites, all provide rich sources of information for hypertext classification that are not major concerns in traditional text classification. Given this, the question of how to effectively use such information becomes important. More specifically, how useful or reliable are hyperlinks when they are used in predicting topic labels of web pages or web sites? What learning algorithms are more powerful for discriminating informative links from noisy links with respect to classification? What algorithms are more robust in terms of dealing with a noisy vocabulary of hundreds of thousands of words? What kind of meta-data about web sites could be exploited and how useful are they if available?

These open questions have just begun to be addressed in current hypertext classification research. Chakrabarti et al. (1998) reported their experiments with a patents database and a subset of Yahoo. They treated the text of a document's neighbors (the documents having a link to/from that document) as if they were local terms and observed worse performance of their classifier compared with ignoring the links. In

contrast, our results show that these techniques lead to improved performance on one of our three hypertext classification tasks. They also developed an iterative algorithm that labeled test documents using the labels of surrounding documents. The labels were either given to the learner, or guessed from the text or a previous iteration. Improved accuracy of their classifier was observed, compared to not having such an iterative labelling process. It is not clear how well their results would generalize to web page and web site classification problems since most of their hypotheses concerning hypertext classification were tested on a Patents database which could have different properties than a web corpus.

Oh et al. (2000) also developed an algorithm for exploting hyperlink information. They used a single-pass algorithm to label documents which used neighboring documents that were similar to the test document. Both the guessed class, and the text of the similar neighboring documents were used for classification. Improved performance of their classifier on an encyclopedia corpus (with hyperlinks) was observed.

Slattery and Mitchell (2000) studied the use of hyperlinks from a different angle. They employed FOIL (First Order Inductive Learner) (Quinlan, 1990), a relational learner to exploit the relational structure of the web, and a Hubs & Authorities style algorithm (Kleinberg, 1998) to exploit the hyperlink topology. By combining these two algorithms they had improved classification accuracy over FOIL, on a corpus of university web pages in three classes.

There are also published results of traditional text classification methods applied to datasets consisting of web pages by treating each page independently and ignoring the links and category distribution among linked pages (Ghani, 2000; McCallum & Nigam, 1998).

While the work listed above suggests interesting ideas with some observations, hypertext categorization remains an under-explored area. More exploration of alternative approaches and different applications is needed before we can draw general conclusions about hypertext categorization problems. In this paper, we focus on how hypertext categorization differs from traditional text categorization, and how various sources of information beyond document content can be effectively used to improve classification performance. We use three well-known classification methods (Naive Bayes, Nearest Neighbor and FOIL) and an assortment of document representations designed to exploit the relationships among web pages and web sites in the form of links, words in a linked neighborhood and meta-data. We test our hypotheses on three different classification tasks, two consisting of classifying company web sites and the third one classifying university web pages. Our study shows that naively using hyperlink information can hurt the classification performance and that carefully using the linked neighborhood can be extremely beneficial. We also find that meta-data available for hypertext is a useful source of information and that combining meta-data with the text using very simple combination methods results in better performance.

2. Methodology

The purpose of the experiments presented in this paper is to explore various hypotheses about the structure of hypertext, especially as it relates to hypertext classification. While the scope of the experimental results presented is necessarily confined to our datasets, described in the Section 3, we hope that the analysis that follows will help future research into hypertext classification by providing some ideas about various types of regularities that may be present in other hypertext corpora and how one should construct a classifier to take advantage of them.

2.1 Regularities

Here are six kinds of regularities one might hope to find in a hypertext corpus. These are not mutually exclusive and various combinations of them may be found together in any single classification task. And of course these are orthogonal to any regularity in the content of the documents being classified.

Along with each regularity we discuss how a classifier that takes advantage of each might be designed.

2.1.1 NO REGULARITY

If the documents are linked at random, or at least independent of the document class, then we do not expect to be able to use hyperlinks to build better classifiers. The classification task reduces to standard text classification and we can then use standard text classifiers. Since we believe that hypertext linking is rarely random, this approach doubles as a baseline classifier with which to compare classifiers designed to use hyperlink regularities. We expect any classifier which takes hyperlinks into account will improve on the performance of a standard text classifier.

2.1.2 "ENCYCLOPEDIA" REGULARITY

If the class of a document is the same as the class of the majority of the linked documents, we expect hyperlinks to be a useful addition to the classifier. The

ETRI-Kyemong encyclopedia corpus used in Oh et al. (2000) may well exhibit this property since encyclopedia articles generally reference other articles which are topically similar.

With this kind of regularity, augmenting the text of each document with the text of its neighbors should produce better classification, since we should be adding more topic-related words to the document. Chakrabarti et al. (1998) applied this approach to a database of patents and found that classification performance suffered indicating that the corpus does not have this structure.

2.1.3 "Co-Referencing" Regularity

This is similar to the "encyclopedia" regularity, but in this case, documents with the same class tend to link to documents not of that class, but which are topically similar to each other. An example might be university student index pages which tend not to link to other student index pages, but do link mostly to home pages of students. This regularity still gives us predictive power, but we need to be careful when building the classifier so that it deals with the documents separately from the linked documents.

Using the words of linked documents, but treating them as if they come from a separate vocabulary should help classification with this kind of regularity. A simple way to do this is to prefix the words in the linked documents with a tag, such as `linked-word:`. Chakrabarti et al. (1998) also tried this approach on the patent database and again found that performance suffered, again indicating that the patent database does not have this structure.

2.1.4 Partial "Co-Referencing" Regularity

This is a "co-referencing" regularity where we might have more than a few "noisy" links. We might find this kind of regularity with student home pages, where many students may point to their own pages about their hobbies, but also link to a wide variety of other pages which are less unique to student home pages. If we discover that hobby pages are predictive of student home pages, then we can build a more accurate classifier.

With this kind of regularity, we have to search for the topically similar linked pages (such as the "hobby" pages above) in order to use them for classification. At the top level, this is a clustering problem to find similar documents among all the documents linked to documents in the same class. While the previous approaches were centered on various representations with standard text classification algorithms, this approach requires a more elaborate algorithm. One such algorithm is the Foil algorithm described in the next section. Craven et al. (1998) applied this algorithm to a corpus of university web pages and found that it did improve classification performance, indicating that this corpus does have this kind of regularity.

2.1.5 Preclassified Regularity

Our hypertext corpus may already have the perfect (or at least reasonably good) classifier buried inside of it. Either one page, or some small set of pages, may contain lists of hyperlinks to pages that are mostly members of the same class. A good example of this regularity is any page from the Yahoo topic hierarchy. Locating these predictive "hub" pages should improve classification performance.

If the classification scheme of our corpus is already embedded in the hyperlink structure, we have no need to look at the text of any document. We just need to find those pages within the hypertext "graph" that have this property. We can search for these pages by representing each page with only the names of the pages it links with. Our classifier can then see if any of these linked pages are correlated with a class label.

2.1.6 Meta-Data Regularity

Since we are dealing with hypertext, for many classification tasks that are of practical and commercial importance, there is meta-data available from external sources on the web that can be exploited in the form of additional features. If these features are sufficiently rich and predictive, we can build classifiers that can use them alone or combine them with the hyperlinks and textual information. For web pages, text within the <title> and <meta> tags is usually available and can be used as meta-data. Other domain-specific examples of these types of meta-data include movie reviews for movie classification, online discussion boards for various other topic classification tasks (such as stock market predictions or competitive analysis).

When meta-data are available, we can collect them using information extraction techniques. In particular, we look for features that relate two or more entities/documents being classified. Following the approaches outlined above for hyperlinks, these extracted features can then be used in a similar fashion by using the identity of the related documents and by using the text of related documents in various ways. Any information source from the Web about the entity being classified can be used as a meta-data resource and

the availability and quality of such resources will certainly depend on the classification task. Cohen (2000) described some experiments where he automatically located and extracted such features for several (non-hypertext) classification tasks.

2.2 Learning Algorithms Used

Our experiments used three existing classifiers. Naive Bayes and kNN have been well evaluated for text classification on benchmark collections and offer a strong baseline for comparison. FOIL is a relational learner which has shown promise for hypertext classification.

2.2.1 NAIVE BAYES

Naive Bayes is a simple but effective text classification algorithm for learning from labeled data alone (Lewis, 1998; McCallum & Nigam, 1998). We use the multinomial model as defined in (McCallum & Nigam, 1998) where each word in a document is assumed to be generated independently of the others given the class and use Laplace smoothing to calculate word probabilities.

2.2.2 K-NEAREST NEIGHBOR (KNN)

kNN, an instance-based classification method, has been an effective approach to a broad range of pattern recognition and text classification problems (Dasarathy, 1991; Yang, 1999; Yang et al., 2000). Our kNN uses the conventional Vector Space Model where each document is represented as a vector of term weights, and the similarity between two documents is measured using the cosine value of the corresponding vectors.

2.2.3 FOIL

Quinlan's FOIL (Quinlan, 1990) is a greedy covering algorithm for learning function-free Horn clauses. It induces each clause by beginning with an empty tail and using a hill-climbing search to add literals. An MDL-based stopping function trades off the cost of adding another literal with the gain of excluding additional negative examples. The evaluation function used to guide the hill-climbing search is an information-theoretic measure.

FOIL has already been used for text classification to exploit word order (Cohen, 1995) and hyperlink information (Craven et al., 1998). Here FOIL is used as described in Craven et al., with one `has_`*`word`*`(page)` relation for each word and a symmetric `link_to(page,page)` relation for hyperlinks between pages. To perform n-class classification, FOIL was run on n binary classification problems and each rule's prediction was given a score based on that rule's performance on the test set. The scored predictions from each binary problem were then merged to obtain an n-class prediction.

3. Dataset

To test our proposed approaches to hypertext classification, we needed datasets that would reflect the properties of real-world hypertext classification tasks. We wanted a variety of problems so we could get a general sense of the usefulness of each regularity described in the previous section.

We found three hypertext classification problems for this study: two of them are about classification of company web sites, and the third one is a classification task for university web pages.

3.1 Hoovers-28 and Hoovers-255

The corpus of company web pages was assembled using the Hoovers Online Web resource (`www.hoovers.com`) which contains detailed information about a large number of companies and is a reliable source of corporate information. Ghani et al. (2000) obtained a list of the names and home-page URLs for 4285 companies on the web and used a custom crawler for extracting information from company Web sites. This crawler visited 4285 different company Web sites and searched up to the first 50 Web pages on each site (in breadth first order), examining just over 108,000 Web pages.

Two sets of categories are available from Hoover Online: a coarse classification scheme of 28 classes (industry sectors such as Oil & Gas, Sporting Goods, Computer Software & Services) and a more fine grained classification scheme consisting of 255 classes. These categories label companies, not particular web pages. For this reason, we constructed one synthetic page per company by concatenating all the pages (up to 50) crawled for that company and ignoring the inner links between the pages of that company. Therefore our task for this dataset is web-site classification rather than web-page classification due to the granularity level of the categories in this application.

For meta-data about those web sites, we consulted Hoovers Online which provided information about the company names, and names of their competitors. We constructed several kinds of wrappers (from simple string matcher to statistical information extraction techniques) to extract additional information about the relationships between companies from the web pages in our dataset, such as whether one company

name is mentioned by another in its web page, whether two companies are located in the same state (in U.S.) or the same country (out side of U.S.). In the results section, we only report our experiments using the competitor information, titles, and meta-tags.

The resulting corpora (namely, Hoovers-28 and Hoovers-25) consist of 4,285 pages (synthetic) with a vocabulary of 256,715 unique words (after removing stop words and stemming), 7,762 links between companies (1.8 links per company) and 6.0 competitors per company. Each web-site is classified into one category only for each classification scheme. The most populous (majority) class for Hoovers-28 contains 8% of the documents and the one for Hoovers-255 contains 2% of the documents.

3.2 Univ-6 Dataset

The second corpus also comes from the Web→KB project at CMU and consists of web pages from several universities in the US.

The dataset consists of 4,165 pages with a vocabulary of 45,979 unique words (after removing stop words and stemming). There are 10,353 links between pages in the corpus (2.5 links per page).

The pages were manually labelled into one of 7 classes: student, course, faculty, project, staff, department and other. The department class was ignored in these experiments as it had only 4 instances. The most populous class ("other") is a catch-all class which is assigned to documents (74% of the total) that do not belong in any of the defined classes of interest.

4. Results

Tables 1 and 2 show the macro- and micro-averaged F_1[1] results for various combinations of learner and representation on our three tasks. Time constraints prevented us from completing the FOIL experiments on two of the representations and from doing a full set of preclassified regularity experiments.

We use F_1 score as the evaluation measure which is equivalent to accuracy under the conditions that (1) each document only belongs to one category and (2) the classifier assigns only one category to each document. Since both of these conditions hold true in our experimental setup, F_1 and accuracy can be used interchangeably; F_1 has been commonly used in text classification literature and is also applicable for multi-label classification problems (van Rijsbergen, 1979). All the results reported are for optimal vocabulary sizes for each algorithm, although each algorithm had it's own associated feature selection technique.

Chi-squared scoring was used for feature selection with kNN, while the Naive Bayes experiments used information gain. Since features of related documents can prove useful for classification, feature selection based on class labels is not a good choice for FOIL. Instead we use document frequency for the FOIL experiments presented here.

Referring back to the regularities listed in Section 2.1, we expect that if a given regularity is present in our corpus, then the associated method should outperform the baseline.

4.1 No Regularity

The *Words on Page Only* results show the baseline performance possible using only the text of the documents themselves and not looking at hyperlink information or meta-data information. On the hoovers-28 and hoovers-255 datasets, Naive Bayes and kNN have comparable results which are significantly better than the FOIL results. In contrast, on the Univ-6 dataset, kNN and FOIL perform comparably and significantly better than Naive Bayes.

4.2 Hyperlink Regularities

4.2.1 "ENCYCLOPEDIA" REGULARITY

The rows labeled *All Words from Linked Pages* look for this pattern in the corpus. The kNN and Naive Bayes results on the two Hoovers datasets show that performance suffers badly when compared to the baseline. It's quite clear that these two datasets do not exhibit this regularity. Chakrabarti et al. (1998) also reported decreased performance when using this method on their datasets. This result is not surprising since the word distribution of the neighbors is not similar to the distribution of the class that the company belongs to. This becomes even more evident when we analyze the dataset and see that out of all the hyperlinks linking different companies, only 7% link companies of the same class.

The algorithm proposed by Oh et al. (2000) for exploiting this regularity calculates the likelihood for each document belonging to a certain class by multiplying the class probability (using the words on the web page) by the fraction of neighbors that are in the same class. Applying this to our dataset would result in very low accuracy since a very small fraction of the companies have neighbors in the same class and multiplying the likelihood for the "correct" class by zero

[1] $F_1 = (2 * r * p)/(r + p)$ where r=recall and p=precision

	Hoovers28			Hoovers255			Univ6		
	NB	kNN	FOIL	NB	kNN	FOIL	NB	kNN	FOIL
Words on Page Only	54.3	**55.3**	31.6	**24.6**	19.8	8.0	37.8	41.1	**45.4**
All Words from Linked Pages	**40.3**	35.1	-	**14.8**	12.0	-	32.7	**46.6**	-
All Words from Linked Pages Tagged	**49.0**	45.4	33.1	**17.9**	15.9	9.3	39.5	45.8	**52.6**
HTML Title	37.6	**39.9**		11.3	**14.5**		**37.3**	37.1	
HTML Meta Tags	45.5	**47.4**		17.7	**18.7**		22.0	**35.8**	
Competitor Names	70.0	**74.2**	33.7	40.7	**44.5**	8.3	-	-	-

Table 1. Macro-averaged F_1 results for each classifier on each representation. Best results for each dataset with each representation are shown in bold.

	Hoovers28			Hoovers255			Univ6		
	NB	kNN	FOIL	NB	kNN	FOIL	NB	kNN	FOIL
Words on Page Only	55.1	**58.1**	31.5	**32.5**	32.0	11.6	70.5	**83.0**	82.7
All Words from Linked Pages	**40.1**	38.5	-	18.9	**20.4**	-	74.1	**85.7**	-
All Words from Linked Pages Tagged	**49.2**	48.3	32.9	26.2	**26.9**	12.8	76.4	**87.4**	85.8
HTML Title	40.8	**43.3**		18.8	**22.6**		78.9	**81.6**	
HTML Meta Tags	48.6	**49.8**		25.1	**28.3**		73.6	**78.6**	
Competitor Names	**75.4**	74.5	33.7	52.0	**53.0**	12.0	-	-	-

Table 2. Micro-averaged F_1 results for each classifier on each representation. Best results for each dataset with each representation are shown in bold.

(the fraction of companies in the same class) would classify most of the documents wrongly.

On the other hand, we find that on the Univ6 dataset, both kNN and Naive Bayes improve their performance by using words from the linked neighborhood. This suggests that the Encyclopedia regularity holds true to some extent for the Univ6 dataset and in fact, we found that 24% of the hyperlinks in this dataset link pages of the same class which is significantly higher than the 7% for the hoovers-28 dataset.

4.2.2 "Co-Referencing" Regularity

The *All Words from Linked Pages Tagged* method looks for this pattern. Unlike previous results reported in the same studies, where tagging the words from the neighbors (treating them as if they're from a separate vocabulary) doesn't affect performance, we find interesting results. For Univ-6 dataset, this representation results in higher accuracy than using the words on the web page and naively using all the words from linked neighbors.

On the other hand, for the Hoovers datasets, this variation results in significant performance degradation from the baseline. Interestingly the performance drop is less severe than with the previous method, but ideally we'd like a method for exploiting potential hyperlink regularities that did not result in poorer performance when such regularities were not present in the data. If our classifiers could handle noise "perfectly", then the performance would be at least as good as the baseline which ignore linkage information. This suggests that the classifiers have problems filtering/handling noise which could be overcome given more training data.

4.2.3 Partial "Co-Referencing" Regularity

The *All Words from Linked Pages Tagged* results under FOIL look for this regularity. On all three datasets, FOIL was able to get some improvement over its baseline results, although for Hoovers-28 and Hoovers-255, the initial results were still well below those of Naive Bayes and kNN. This indicates that all three datasets have some partial "co-referencing" regularities and that FOIL was able to find it in each.

4.2.4 Preclassified Regularity

By only using the names of the hyperlinked neighbors (company names) and ignoring the word information, we do not see any improvement but the fact that Naive Bayes still gets some leverage (15% accuracy for 28-classes and 5% for Hoovers-255 are both higher than baseline) from using the names of the companies while completely ignoring the text on their websites suggests that the preclassified regularity exists in our dataset to some extent. (not shown in Table above)

4.3 Meta-Data Regularities

4.3.1 Title and Meta-Tags

We find that titles and meta-tags contain useful information for all three classification tasks and result in performance that is better than default (assigning the majority class to all examples). In particular, for Univ6, both title words and meta-tags perform better than using all the words on the web pages with Naive Bayes with respect to microaverage F_1.

4.3.2 Competitors

The domain-specific meta-data we report results from is the Competitors data. While we tried several other features in our study, we only report one due to space limitations. We can treat the competitor information in the same way as we do the hyperlinks and use this meta-data to search for regularities in the names of the competitors or in the content of their web sites. The Naive Bayes and kNN results use the names of the competitors and this approach produces a sharp boost in accuracy for both 255-classes and 28-classes.

Not shown are results for Naive Bayes and kNN using the content of the competitors web sites. Using this approach, performance doesn't improve over using only the names of the competitors for the 28 class problem but slightly increases accuracy for the 255 class problem. The Foil results use the content of the competitor web sites to search for a partial "co-referencing" regularity and achieve roughly the same performance as in the other Foil experiments on the hoovers datasets.

Another noteworthy point is that tagging the words from hyperlinked neighbors results in an increase in accuracy over the non-tagged version, while tagging the words from competitors actually hurts the accuracy. The usefulness of the competitor information can be explained by the high proportion of competitor pairs in the same class. Unlike the hyperlinked company pairs of which 7% are in the same class, around 70% of pairs of competitors share the same class label.

Apart from competitor information, we also experimented unsuccessfully (in terms of accuracy) with location information about companies. We extracted the cities, states and countries where a company was located, both by using information extraction techniques on the text of the web site, and also from external sources such as Hoovers. Although this information did not result in performance improvement, it may be combined with other features and provide an orthogonal view of the classification problem. Other potentially useful meta-data might include products of a company which again can be extracted automatically from web pages using a dictionary of various product names and types.

4.4 Combining Hypertext and Meta-Data

In previous sections, we showed that text on the web pages and meta-data both provide useful information for hypertext classification. We hypothesize that if they capture different information about the classifications tasks, we can combine them to achieve better performance. The results in Table 3 show that we indeed get better performance by combining Page and Title classifications and Page and Meta-Tag classifications in all but three cases. Since our goal is not to design an elaborate voting/combination algorithm but to demonstrate that combining text and meta-data improves performance, we simply combine the scores from the two presentations by adding them together. It is likely that weighting the predictions from each classifier would result in better performance and we plan to explore this in future work.

5. Concluding Remarks

In this paper, we addressed the open questions in hypertext categorization, regarding how to effectively use the rich information which is typically available in hypertext and makes the task significantly different from traditional text classification. We specified six hypotheses about the regularities in hyperlinks, topic distribution a linked neighborhood, and meta-data about web sites which provide a framework in which to think about regularities in hypertext which we hope will aid researchers to design algorithms aimed at exploiting the richness of hypertext. We examined these hypotheses with three well-known supervised classification algorithms on a collection of web pages with practical classification schemes. Our major findings include:

- Using words in web pages alone often yielded suboptimal performance of classifiers, compared to exploiting additional sources of information.
- Hyperlinks can be highly "noisy" and can be more harmful than helpful for hypertext classification, as evident in this study on the two Hoovers datasets and the previous study by Chakrabarti et al. They can also be helpful when the "Encyclopedia" regularity holds, as observed in this study on the Univ-6 dataset and also previously by Oh et al. Careful examination of this regularity is crucial for the design of classification algorithms.
- Meta-data about web pages or web sites can be extremely useful for improving the classification

	Hoovers28		Hoovers255		Univ6	
	NB	kNN	NB	kNN	NB	kNN
Words on Page + Title	56.0	59.1	33.0	32.6	**72.2**	85.0
Words on Page + Meta-Tags	59.2	60.9	34.7	33.4	**73.2**	**81.2**

Table 3. Micro-averaged F_1 results for combination (with equal weights) of meta-data and text with kNN and Naive Bayes. Scores less that the highest of the two scores being combined are shown in bold.

accuracy, as evident in this study. This suggests the importance of examining the availability of meta-data in real-world applications, and using Information Extraction techniques for automated acquisition of meta-data.

- Standard text classification algorithms such as Naive Bayes and Nearest Neighbor algorithms can be used successfully to examine various hypotheses about hyperlinks and meta-data. FOIL, on the other hand, has the power for discovering relational regularities that cannot be explicitly identified using the other algorithms.

While the scope of the experimental results presented is necessarily confined to our datasets, we hope that our analysis will help future research into hypertext classification by providing some ideas about various types of regularity that may be present in a hypertext corpus and how one should construct a classifier to take advantage of each.

References

Chakrabarti, S., Dom, B., & Indyk, P. (1998). Enhanced hypertext categorization using hyperlinks. *Proceedings ACM SIGMOD International Conference on Management of Data* (pp. 307–318). Seattle, Washington: ACM Press.

Chen, H., & Dumais, S. (2000). Bringing order to the web: Automatically categorizing search results. *Proceedings of CHI'00, Human Factors in Computing Systems.*

Cohen, W. (1995). Learning to Classify English Text with ILP Methods. In L. D. Raedt (Ed.), *Advances in inductive logic programming.* IOS Press.

Cohen, W. (2000). Automatically extracting features for concept learning from the web. *Seventeenth International Conference on Machine Learning.*

Craven, M., Slattery, S., & Nigam, K. (1998). First-order learning for web mining. *Tenth European Conference on Machine Learning.*

Dasarathy, B. V. (1991). *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques.* McGraw-Hill Computer Science Series. Las Alamitos, California: IEEE Computer Society Press.

Ghani, R. (2000). Using error-correcting codes for text classification. *Seventeenth International Conference on Machine Learning.*

Ghani, R., Jones, R., Mladenic, D., Nigam, K., & Slattery, S. (2000). Data mining on symbolic knowledge extracted from the web. *Workshop on Text Mining at the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Kleinberg, J. (1998). Authoritative Sources in a Hyperlinked Environment. *Proceedings of the Nineth Annual ACM-SIAM Symposioum on Discrete Algorithms.*

Lewis, D. D. (1998). Naive (Bayes) at forty: The independence assumption in information retrieval. *Proceedings of ECML-98.*

McCallum, A., & Nigam, K. (1998). A comparison of event models for naive Bayes text classification. *AAAI-98 Workshop on Learning for Text Categorization.* Tech. rep. WS-98-05, AAAI Press.

Oh, H.-J., Myaeng, S. H., & Lee, M.-H. (2000). A practical hypertext categorization method using links and incrementally available class information. *Proceedings of the Twenty Third ACM SIGIR Conference* (pp. 264–271). Athens, Greece.

Quinlan, J. R. (1990). Learning logical definitions from relations. *Machine Learning, 5,* 239–266.

Slattery, S., & Mitchell, T. (2000). Discovering test set regularities in relational domains. *Seventeenth International Conference on Machine Learning.*

van Rijsbergen, C. (1979). *Information retrieval.* London: Butterworths.

Yang, Y. (1999). An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval, 1,* 67–88.

Yang, Y., Ault, T., & Pierce, T. (2000). Combining multiple learning strategies for effective cross validation. *The Seventeenth International Conference on Machine Learning (ICML'00)* (pp. 1167–1182).