Proceedings of the AAAI-2004 Workshop on Adaptive Text Extraction and Mining (ATEM-2004),
pp. 27-32, San Jose, CA, July 2004.

Using Soft-Matching Mined Rules to Improve Information Extraction

Un Yong Nahm and Raymond J. Mooney
Department of Computer Sciences
The University of Texas at Austin
1 University Station C0500
Austin, TX 78712-1188
{pebronia,mooney}@cs.utexas.edu

Abstract

By discovering predictive relationships between different pieces of extracted data, data-mining algorithms can be used to improve the accuracy of information extraction. However, textual variation due to typos, abbreviations, and other sources can prevent the productive discovery and utilization of hard-matching rules. Recent methods for inducing *soft-matching* rules from extracted data can more effectively find and exploit predictive relationships in textual data. This paper presents techniques for using mined soft-matching association rules to increase the accuracy of information extraction. Experimental results on a corpus of computer-science job postings demonstrate that soft-matching rules improve information extraction more effectively than hard-matching rules.

Introduction

Information extraction (IE) and knowledge discovery and data mining (KDD) are both useful tools for discovering knowledge from textual corpora. However, there has been relatively little research on exploring the productive integration of traditional IE and KDD methods. Nahm & Mooney (2000) introduced a mutually-beneficial framework for integrating IE and KKD. IE benefits KDD by extracting structured data from textual documents, which can then be mined using traditional methods. KDD benefits IE by discovering rules that support predictions that can improve the accuracy of subsequent information extraction.

The predictive relationships between different IE slot fillers discovered by KDD can provide additional clues about what information should be extracted from a document. For example, suppose we discover the following rule from data on programming languages and topic areas extracted from a corpus of computer-science job postings: "SQL" $\in language \rightarrow$ "Database" $\in area$. If the IE system extracted "SQL" for the $language$ slot but failed to extract "Database" for the $area$ slot, we may want to assume there was an extraction error and add "Database" to the $area$ slot. Since typically the *recall* (percentage of correct slot fillers extracted) of an IE system is significantly lower than its *precision* (percentage of extracted slot fillers which are correct) (DARPA 1998), such predictive relationships can be productively used to improve recall by suggesting additional information to extract.

Nahm & Mooney (2000) employed C4.5rules (Quinlan 1993) to induce predictive rules which were then used to improve the recall of subsequent information extraction. Unfortunately, extracted text often exhibits variations that can prevent mining algorithms from discovering important regularities. Variations can arise from typographical errors, misspellings, abbreviations, as well as other sources. For example, in data on local job offerings that we automatically extracted from newsgroup postings, the Windows operating system is variously referred to as "Microsoft Windows", "MS Windows", "Windows 95/98/ME", etc.. To address such textual variation, we have developed two new KDD algorithms, TEXTRISE (Nahm & Mooney 2001) and SOFTAPRIORI (Nahm & Mooney 2002), that induce *soft-matching* rules appropriate for variable text data. The resulting rules use a text-similarity measure such as string edit-distance (Gusfield 1997) or vector-space cosine similarity (Salton 1989) to flexibly match textual items.

In this paper, we demonstrate that using such soft-matching rules to predict potential extractions improves the accuracy of IE more than using hard-matching rules (as in our previous work). Specifically, we compare the hard-matching rules mined with APRIORI (Agrawal & Srikant 1994) to soft-matching rules mined with SOFTAPRIORI with respect to their ability to improve information extraction from a corpus of job postings.

Background

Information Extraction

The goal of an IE system is to locate specific data in natural-language text. The data to be extracted is typically given by a template which specifies a list of slots to be filled with substrings taken from the document. IE is useful for a variety of applications, particularly given the recent proliferation of Internet and web documents. In particular, machine learning techniques have been suggested for extracting informa-

- city: Austin
- state: TX
- language: Java; C; C++
- platform: MS Windows
- area: Technical writing; QA; technical support
- application: Microsoft Office 97/2000; Outlook 97/98; WordPerfect; Lotus 1-2-3
- hardware: Laptops; Printers; Palm Pilots
- major: Computer Sciences
- required_degree: BS

Figure 1: Sample filled template

tion from text documents in order to create easily searchable databases from the information, thus making the online text more accessible (Califf & Mooney 1999). For instance, information extracted from job postings on company websites can be used to build a searchable database of jobs.[1]

In this paper, we consider the task of extracting a database from postings to the USENET newsgroup, `austin.jobs` with BWI (Boosted Wrapper Induction) (Freitag & Kushmerick 2000). Figure 1 shows a filled computer-science job template where several slots may have multiple fillers. For example, slots such as languages, platforms, applications, and areas usually have more than one filler, while slots related to the city or state have only one.

BWI learns extraction rules by boosting a *wrapper induction* system. A *wrapper* is a contextual pattern that is simple but highly accurate. Wrapper induction, the automated process of learning wrappers, has been traditionally used to extract data from highly structured text such as web pages generated by CGI scripts. Since individual patterns typically have high precision and low recall, BWI uses boosting (Freund & Schapire 1996) to create accurate extractors by combining many high-precision patterns. In BWI, IE is treated as a classification problem, and contextual patterns are learned to identify the beginning and end of relevant text segments. Boundary patterns are repeatedly learned and training examples are reweighted using the ADABOOST algorithm (Schapire & Singer 1998) so that subsequent patterns identify examples missed by previous rules. BWI has been shown to perform reasonably well in several domains, from traditional free text to structured HTML documents.

The SOFTAPRIORI Algorithm

Data mining methods generally require terms in discovered rules to exactly match database entries. Normal variation in textual data items can therefore prevent the discovery of important regularities. Variations can arise from typographical errors, misspellings, abbreviations, as well as other sources. Variations are particularly pronounced in data that is automatically extracted from unstructured or semi-structured documents or web pages. SOFTAPRIORI (Nahm & Mooney 2002) discovers *soft matching* association rules whose antecedents and consequents are evaluated based on sufficient similarity to database entries.

[1]`http://flipdog.monster.com/`

- database (databases, database sys.) $\in$ area $\Rightarrow$ oracle (oracle7) $\in$ application [3.2%, 43.2%]
- mfc $\in$ area $\Rightarrow$ windows (windows nt, windows 95, windows 3.1, windows 3.x, windowsnt, windows95, windows'95) $\in$ platform [2.7%, 39.0%]
- linux (linux-pc) $\in$ platform $\Rightarrow$ html (dhtml, d/html, shtml) $\in$ language [11.0%, 50.0%]

Figure 2: Sample soft association rules mined from 600 job postings

SOFTAPRIORI generalizes the standard APRIORI algorithm for discovering association rules (Agrawal & Srikant 1994) by allowing soft matching based on a given similarity metric for each field. Similarity of textual entries in SOFTAPRIORI is measured using standard edit-distance or bag-of-words measures. In our experiments, we used edit-distance with affine gap cost (Needleman & Wunsch 1970), incurring one penalty for starting a new gap (i.e. sequence of character deletions) and a smaller penalty for continuing an existing gap (i.e. contiguous character deletions).

In SOFTAPRIORI, soft relations on items sets are defined, assuming that a function, $similarity(x, y)$, is given for measuring the similarity between two items x and y. Let $I = \{i_1, i_2, ..., i_m\}$ be a set of literals, called *items*. Let $\mathcal{D}$ be a set of baskets, where each basket $B \subseteq I$ is a set of items. A soft association rule is an implication of the form $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$ and no item in X is similar to an element of Y. The problem of mining soft association rules is finding all soft association rules, $X \Rightarrow Y$, such that their *soft-support* and the *soft-confidence* are greater than user-defined minimum values. Formal definitions of soft-support and soft-confidence are straightforward generalizations of the traditional ones that allow items to match as long as their similarity exceeds a pre-specified threshold.

In order to find association rules for extracted data, we first map each extracted filler to an item. A document is represented as a basket of items where each item is a slot-filler pair extracted from the document. By applying SOFTAPRIORI to job postings, we mined relationships between items such as "If a computer-related job posting requres knowledge of MFC then it also lists Windows in the list of required skills." Sample rules mined from a database of 600 job postings extracted from a USENET newsgroup are shown in Figure 2. Items found to be similar to a given item during training are shown in parentheses, and values for soft-support and soft-confidence are shown in brackets.

Using Soft Association Rules to Improve Information Extraction

The general DISCOTEX framework described in (Nahm & Mooney 2000) serially combines an information extraction system and a KDD module. After constructing an IE system that extracts the desired set of slots for a given application,

Parameter: $minconf$, $minsup$ - minimum confidence/support.
T_{sim} - similarity threshold.
T_{ex} - extraction threshold.
Input: D_{train} - set of labeled documents.
D_{test} - set of n unlabeled documents.
Output: L - set of $n$ew labels for D_{test}.

Function InformationExtraction (D_{train}, D_{test})
Build an information extraction rule base, RB_{IE}
(by applying BWI to D_{train})
Let L_{train} := set of labeled slot fillers of D_{train}.
Build a soft association rule base, RB
(by applying SOFTAPRIORI to L_{train}
with parameters $minconf$, $minsup$, and T_{sim})
For each unlabeled document $D_{test}(i)$ **do**
Extract slot fillers from $D_{test}(i)$ using RB_{IE}.
Let $L(i)$:= set of extracted slot fillers of $D_{test}(i)$.
Until no change obtained on $L(i)$ **Do**
For each rule R $(X \Rightarrow Y) \in RB$ **do**
If R fires on $L(i)$
For each matching substring Y' in $D_{test}(i)$ **do**
(with $similarity(Y, Y') \geq T_{sim}$)
$score(Y') := similarity(Y, Y') \times conf(R)$.
If $score(Y') \geq T_{ex}$
add Y' to $L(i)$.
Let L := ($L(1)$,$L(2)$,...,$L(n)$).
Return L.

Figure 3: Algorithm specification

a database is constructed from a corpus of texts by applying the extractor to each document to create a collection of structured records. A standard rule mining algorithm is then applied to the resulting database to discover interesting relationships. Specifically, we mine rules for predicting the presence or absence of each database item given information about all other items.

However, for data sets with significant textual variation, hard-matching rules discovered by standard data mining algorithms may not work. We propose mining soft-matching rules instead, which allow non-standardized database entries to match antecedents and consequents based on relevant similarity metrics. A benefit of association-rule mining instead of classification-rule induction is that consequents of rules are not predetermined, resulting in efficient mining of all potential associations as part of a single process. When inducing classification rules, a separate learning step must be run for predicting each possible item. For instance, classification rule induction is not as efficient for our job-postings data set with as many as 700 items in 600 documents.

Tests of IE systems usually consider two accuracy measures, *precision* and *recall*. Precision is the number of correctly extracted items divided by the total number of extractions while recall is the number of correct extractions divided by the total number of items actually present in the documents. Also, *F-measure*, the harmonic mean of precision and recall was introduced to combine them. Many extraction systems provide relatively high precision, but recall is typically much lower. Currently, BWI's search is also biased toward high precision. Experiments in the job postings domain shows BWI's precision (e.g. high 50%'s) is higher than its recall (e.g. low 40%'s) (Freitag & Kushmerick 2000). Although several methods have been developed that allow a rule learner to trade-off precision and recall (Cohen 1996), this typically leaves the overall F-measure unchanged.

By forward-chaining on extracted data using mined soft-association rules, we can derive additional probable extractions, thereby improving recall without unduly sacrificing precision. For example, suppose we discover the rule "MS Access 2000" $\in$ `application` $\Rightarrow$ "Database" $\in$ `area`. If the IE system extracted "Access 2000" $\in$ `application` but failed to extract "Database" $\in$ `area`, we may want to assume there was an extraction error and add "Database" to the area slot, potentially improving recall. Therefore, after mining knowledge from extracted data, the discovered soft-matching rules are used to predict additional potential extractions during subsequent extraction. Pseudocode shown in Figure 3 describes the use of mined rules in information extraction.

The final decision whether or not to extract a predicted filler is based on whether the filler (or any of its "synonyms") occurs in the document. If a string equal or similar to the predicted filler is found in the text, the extractor considers its prediction confirmed and extracts the string. In the previous example, even if the string "Database" is not found in the document, a similar string such as "databases" is still considered for extraction since $similarity$("Database","databases") $\geq T_{sim}$ where T_{sim} is the prespecified threshold for determining a match. The confidence of the rule is also considered in confirming that the rule is strong enough to extract the filler, combined with the similarity information indicating how close the actual string is to the predicted one.

In summary, documents which the user has annotated with extracted information are used to create a database. The SOFTAPRIORI rule miner then processes this database to construct a knowledge base of soft-matching rules for predicting slot values. These rules are then used during testing to improve the recall of the existing IE system by proposing additional slot fillers whose *similar* strings are confirmed to be present in the document before adding them to the final extraction template. The overall architecture of the final system is presented in Figure 4.

Evaluation

Dataset and Experimental Methodology

To test the overall system, 600 computer-science job postings to the newsgroups `austin.jobs` and `misc.jobs.offered` were annotated with information to be extracted. We used a simple web-crawler for spidering the `groups.google.com` web site in order to collect documents from the newsgroups.

Not all the documents collected from the Internet are relevant documents. Non-computer-science job postings or resumé postings, spam, and other irrelevant documents were filtered out by a trained text categorizer. A bag-of-words Naive-Bayes text categorizer (McCallum & Nigam 1998) is

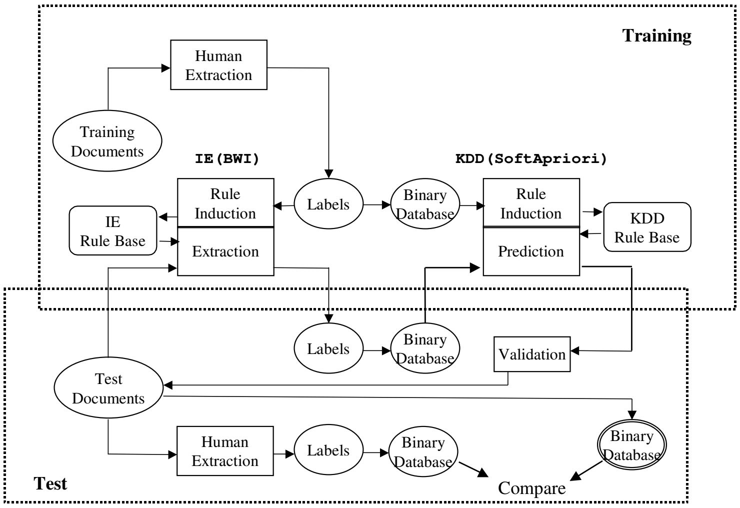

Figure 4: The system architecture

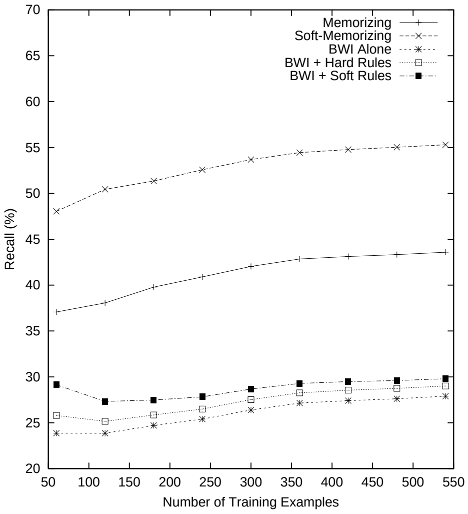

Figure 5: Recall on job postings

Slots	AvgNumFiller	AvgNumDoc	NumFiller
language	0.13	2.30	80
platform	0.17	7.11	104
application	0.30	3.76	179
area	0.60	1.17	361
total	1.21	1.38	724

Table 1: Statistics on slot-fillers

used to identity relevant documents. Classification accuracy as measured by ten-fold cross-validation is 91.17%, with 91.99% precision and 86.0% recall.

Ten-fold cross validation was used to generate training and test sets for extraction from the set of documents. Rules were mined for predicting the fillers of the `languages`, `platforms`, `applications`, and `areas` slots, since these are usually filled with multiple items that have potential predictive relationships. Statistics on these slot-fillers are shown in Table 1, including the average number of fillers per document, average number of documents per filler, and the total number of distinct filler strings in the corpus.

Parameters for BWI are set to the default values, a lookahead of 3 and 20 boosting iterations. The default set of wildcards is used. The similarity threshold, minimum support, and minimum confidence for APRIORI and SOFTAPRIORI were set to 0.70, 5%, and 10%, respectively. Association rules without antecedents (e.g. $\Rightarrow$ `C++`) are also employed. The minimum support and confidence values are determined by validating on the training data set. The minium confidence value is set to a low value because the final extraction of a filler is confirmed by checking if same (hard rules) or similar (soft rules) strings are found in the document or not. The match cost, mismatch cost, gap-start cost, and gap-extend cost parameters for the affine-gap edit distance were set to 0, 3, 3, and 1, respectively. All white spaces in strings are considered as blank characters and upper and lower cases are distinguished only in the IE phase.

Results

To evaluate our system, we compared the performance of BWI alone, BWI aided with hard-matching rules mined by standard APRIORI, and BWI with soft-matching association rules. Figures 5 and 6 and show the learning curves for recall and and F-measure for the job-postings data. The same set of human-annotated training data was provided to both BWI and the rule miner as shown in Figure 4.

As a benchmark, we show the performance of a simple baseline (Memorizing) for increasing recall that always extracts substrings that are known fillers for a particular slot. This baseline remembers all slot-fillers that appear at least once in the training data. Whenever a known filler string, e.g. `Java`, is contained in a test document, it is extracted as a filler for the corresponding slot, e.g. `language`. This method has good recall but limited precision since a filler string contained in a document is not necessarily the correct filler for the corresponding slot. For instance, “www” can appear in a document, not in a list of required skills but in a URL of the company’s homepage.

We also tested a “soft” version of this baseline (Soft-Memorizing) that extracts all strings that are sufficiently *similar* to known items in the training data. Although this increases recall, it decreases precision even further. For example, “Peoplesoft” remembered as a filler for the `application` slot can cause the system to extract the spurious filler “people”. The fact that the F-measure of these baselines are worse than the proposed system demonstrates the additional value of rule mining for improving extraction performance.

As hypothesized, BWI with soft-matching rules provides higher recall, and in spite of decreasing precision somewhat, overall F-measure is moderately increased. For each training set size, systems were compared to determine if their differences in recall were statistically significant us-

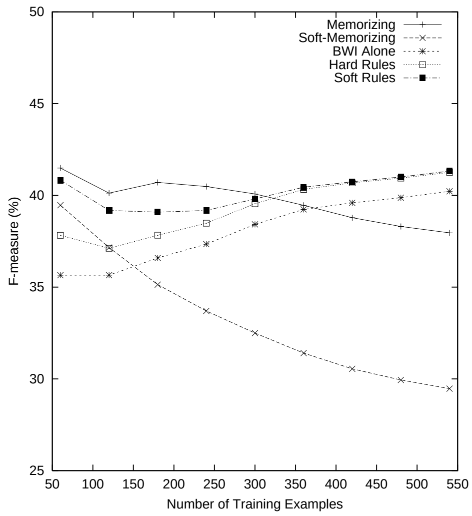

Figure 6: F-measure on job postings

ing a two-tailed, paired *t*-test ($p < 0.05$). For all set of training examples, using soft-matching rules is significantly better than both using hard-matching rules and unaided extraction while using hard-matching rules is also significantly better than unaided extraction. Although the differences are somewhat small, these results demonstrate the advantage of mining soft-matching rules for improving extraction accuracy. We believe that the small gains especially in F-measure are due to the relatively low accuracy of the underlying IE learner, BWI, and therefore plan to apply our method to a more accurate IE learning system, such as RAPIER (Califf & Mooney 1999).

Related Research

Although there is a growing interest in the general topic of text mining (Berry 2003; Grobelnik 2003; Hearst 2003), there has been relatively little research exploring the integration of IE and KDD. Several rule-induction and association-rule-mining algorithms have been applied to databases of corporations or product reviews automatically extracted from the Web (Ghani *et al.* 2000; Ghani & Fano 2002; Pierre 2002); however, these projects do not use the mined knowledge to improve subsequent extraction. Recently, a probabilistic framework for unifying information extraction and data mining has been proposed (McCallum & Jensen 2003). A general approach for using statistical relational models to integrate IE and KDD is presented, but an actual implementation and experimental results for this approach are still forthcoming. A boosted text classification system based on link analysis (Cohen 2003), and a feedback combination of an HTML parser and a high-level wrapper (Cohen, Hurst, & Jensen 2002) are related to our work in spirit since they also attempt to improve the underlying learners by utilizing feedback from KDD modules. The use of Web-based statistics (search-engine hit counts) to improve the precision of information extraction has been also proposed recently (Soderland *et al.* 2004).

Future Work

Our current experiments mine associations only from human-annotated data. An interesting question is whether the performance of an IE system can be further improved by discovering associations in automatically extracted data. Although adding information extracted from unannotated documents to the database may result in a larger database and therefore some better prediction rules, it may also create noise in the database due to extraction errors and consequently cause some inaccurate prediction rules to be discovered as well.

Currently, we only consider improving the recall of IE. Methods for using mined knowledge to improve extraction precision are also needed. Simply eliminating extracted fillers that are not predicted is too coarse and would likely severely damage recall. One possible solution is to use a variation of *negative* association rules (Savasere, Omiecinski, & Navathe 1998; Wu, Zhang, & Zhang 2002). By confidently predicting the *absence* of certain slot values given other extracted information, both precision and recall could potentially be improved. A related issue is combining the confidence measures for prediction rules with the extraction confidences from the IE system to produce an overall confidence in final extraction decisions.

The procedure for selecting the slots to be used in rule mining needs to be automated. In the current experiments, we manually chose four slots from the computer-science job template. By identifying and quantifying the correlations between slot values, this decision could be automated. Automatic learning of threshold values for mining soft rules and dynamic setting of minimum support and confidence values on each training data set could also be explored.

Conclusions

In previous work, we introduced an approach to using predictive rules mined from extracted data to improve the recall of information extraction. However, this approach was limited by the requirement that the antecedents and consequents of mined rules exactly match textual items. The normal variation that occurs in textual information frequently prevents such an approach from effectively exploiting many of the potentially-useful predictive relationships in the data. In more recent work, we have developed techniques for mining *soft-matching* rules that employ standard text-similarity metrics to discover more subtle relationships in variable textual data. By combining these ideas, we have developed a method for using soft-matching mined rules to further improve the accuracy of information extraction. Empirical experiments on a real text corpus showed that our new method can more effectively use automatically-discovered knowledge to improve the recall (and F-measure) of information-extraction.

Acknowledgements

This research was supported by the National Science Foundation under grant IIS-0117308.

References

Agrawal, R., and Srikant, R. 1994. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Databases (VLDB-94)*, 487–499.

Berry, M. W., ed. 2003. *Proceedings of the Third SIAM International Conference on Data Mining(SDM-2003) Workshop on Text Mining*.

Califf, M. E., and Mooney, R. J. 1999. Relational learning of pattern-match rules for information extraction. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99)*, 328–334.

Cohen, W. W.; Hurst, M.; and Jensen, L. S. 2002. A flexible learning system for wrapping tables and lists in HTML documents. In *Proceedings of the Eleventh International World Wide Web Conference (WWW-2002)*, 232–241. Honolulu, HI: ACM.

Cohen, W. W. 1996. Learning to classify English text with ILP methods. In De Raedt, L., ed., *Advances in Inductive Logic Programming*. Amsterdam: IOS Press. 124–143.

Cohen, W. W. 2003. Improving a page classifier with anchor extraction and link analysis. In Becker, S.; Thrun, S.; and Obermayer, K., eds., *Advances in Neural Information Processing Systems 15*, 1481–1488. Cambridge, MA: MIT Press.

DARPA., ed. 1998. *Proceedings of the Seventh Message Understanding Evaluation and Conference (MUC-98)*. Fairfax, VA: Morgan Kaufmann.

Freitag, D., and Kushmerick, N. 2000. Boosted wrapper induction. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, 577–583. Austin, TX: AAAI Press / The MIT Press.

Freund, Y., and Schapire, R. E. 1996. Experiments with a new boosting algorithm. In Saitta, L., ed., *Proceedings of the Thirteenth International Conference on Machine Learning (ICML-96)*, 148–156. Morgan Kaufmann.

Ghani, R., and Fano, A. E. 2002. Using text mining to infer semantic attirbutes for retail data mining. In *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM-2002)*, 195–202. Maebash City, Japan: IEEE Computer Society.

Ghani, R.; Jones, R.; Mladenić, D.; Nigam, K.; and Slattery, S. 2000. Data mining on symbolic knowledge extracted from the Web. In Mladenić, D., ed., *Proceedings of the Sixth International Conference on Knowledge Discovery and Data Mining (KDD-2000) Workshop on Text Mining*, 29–36.

Grobelnik, M., ed. 2003. *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence(IJCAI-2003) Workshop on Text Mining and Link Analysis (TextLink-2003)*.

Gusfield, D. 1997. *Algorithms on Strings, Trees and Sequences*. New York: Cambridge University Press.

Hearst, M. A. 2003. What is text mining? http://www.sims.berkeley.edu/~hearst/text-mining.html.

McCallum, A., and Jensen, D. 2003. A note on the unification of information extraction and data mining using conditional-probability, relational models. In *Proceedings of the IJCAI-2003 Workshop on Learning Statistical Models from Relational Data*.

McCallum, A., and Nigam, K. 1998. A comparison of event models for naive Bayes text classification. In *Papers from the AAAI-98 Workshop on Text Categorization*, 41–48.

Nahm, U. Y., and Mooney, R. J. 2000. A mutually beneficial integration of data mining and information extraction. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, 627–632.

Nahm, U. Y., and Mooney, R. J. 2001. Mining soft-matching rules from textual data. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001)*, 979–984.

Nahm, U. Y., and Mooney, R. J. 2002. Mining soft-matching association rules. In *Proceedings of the Eleventh International Conference on Information and Knowledge Management (CIKM-2002)*, 681–683.

Needleman, S. B., and Wunsch, C. D. 1970. A general method applicable to the search for similarities in the amino acid sequences of two proteins. *Journal of Molecular Biology* 48:443–453.

Pierre, J. M. 2002. Mining knowledge from text collections using automatically generated metadata. In Karagiannis, D., and Reimer, U., eds., *Proceedings of the Fourth International Conference on Practical Aspects of Knowledge Management (PAKM-2002)*, 537–548. Vienna, Austria: Springer. Lecture Notes in Computer Sicnece Vol. 2569.

Quinlan, J. R. 1993. *C4.5: Programs for Machine Learning*. San Mateo,CA: Morgan Kaufmann.

Salton, G. 1989. *Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer*. Addison-Wesley.

Savasere, A.; Omiecinski, E.; and Navathe, S. B. 1998. Mining for strong negative associations in a large database of customer transactions. In *Proceedings of the 14th International Conference on Data Engineering (ICDE-98)*, 494–502. Orlando, FL: IEEE Computer Society.

Schapire, R. E., and Singer, Y. 1998. Improved boosting algorithms using confidence-rated predictions. In Piatetsky-Shapiro, G.; Agrawal, R.; and Stolorz, P. E., eds., *Proceedings of the 11th Annual Conference on Computational Learning Theory*, 80–91. Madison, WI: ACM.

Soderland, S.; Etzioni, O.; Shaked, T.; and Weld, D. S. 2004. The use of Web-based statistics to validate information extraction. To appear in Papers from the AAAI-2004 Workshop on Adaptive Text Extraction and Mining(ATEM-2004) Workshop, San Jose, CA.

Wu, X.; Zhang, C.; and Zhang, S. 2002. Mining both positive and negative association rules. In Sammut, C., and Hoffmann, A. G., eds., *Proceedings of the Ninteenth International Conference on Machine Learning (ICML-2002)*, 658–665. Sydney, Australia: Morgan Kaufmann.