Discovering Knowledge from Relational Data Extracted from Business News

A. Bernstein, S. Clearwater, S. Hill, C. Perlich, F. Provost

Contact author: Abraham Bernstein, bernstein@stern.nyu.edu

May 27, 2002

Abstract

Thousands of business news stories (including press releases, earnings reports, general business news, etc.) are released each day. Recently, information technology advances have partially automated the processing of documents, reducing the amount of text that must be read. Current techniques (e.g., text classification and information extraction) for full-text analysis for the most part are limited to discovering information that can be found in single documents. Often, however, important information does not reside in a single document, but in the relationships between information distributed over multiple documents.

This paper reports on an investigation into whether knowledge can be discovered automatically from relational data extracted from large corpora of business news stories. We use a combination of information extraction, network analysis, and statistical techniques. We show that relationally interlinked patterns distributed over multiple documents can indeed be extracted, and (specifically) that knowledge about companies' interrelationships can be discovered. We evaluate the extracted relationships in several ways: we give a broad visualization of related companies, showing intuitive industry clusters; we use network analysis to ask who are the central players, and finally, we show that the extracted interrelationships can be used for important tasks, such as classifying companies by industry membership.

Introduction

Text-processing technologies have received increasing attention and use, as the deluge of text information increases. Methods from information retrieval (Salton and McGill 1983) have seen tremendous increases in usage, most often embedded in search engines. Researchers have focused attention on text classification (Yang 1999), information triage (Macskassy et al. 2001; Marshall and Shipman 1997), and information extraction (Califf and Mooney 1999).

Much less research has addressed the extraction/discovery of knowledge that resides not in a single document, but in a corpus of documents. For example, we extract knowledge about the relationships between businesses from large collections of business news stories. Any given news story may (or may not) contain partial information, and some news stories may even contain misleading information. However, presumably, if one were to read and remember all the news stories, general knowledge would become clear: which companies are related to each other? Which companies are central players? To what groups (e.g., industries) do different companies belong?

This paper presents a pilot study showing that network analysis techniques and statistical approaches combined with state-of-the-art information extraction techniques can be used to discover interlinked patterns automatically in large corpora of business news stories. The discovery of knowledge from multiple documents has been called "Text Data Mining" (Hearst 1999), which according to Hearst (at the time) had "…a fair amount of hype but as yet no practitioners." We have found only a few closely related studies, including the building of a knowledge base of company information from web sites (Craven et al. 1998), the discovery of medical knowledge from multiple articles (Swanson and Smalheiser 1994), and the discovery of knowledge from business news stories (Feldman and Dagan 1995)—upon which our work builds. Additionally, the U.S. Government has become critically interested in the extraction/discovery of relational patterns from collections of text documents, because they believe it would increase the effectiveness/productivity of intelligence analysts seeking clues to terrorist activity.[1,2]

[1] http://www.darpa.mil/ipto/research/eeld/index.html

The main question of this paper is: can we discover knowledge about *relationships* between businesses from large corpora of business news stories (where that knowledge is distributed over a large number of documents)? To that end we collected four months' worth of business news, which we processed first using a state-of-the-art information-extraction tool, and then processed further using data-mining methods. The goal of this study is to establish that we can discover non-trivial knowledge that is distributed across news stories. For example, one of our domain experts (a business researcher) would like to be able to determine automatically the relatedness of a company to an industry.

We first briefly describe the process for extracting entities and relationships from business news. Next we turn to the analysis of the extracted, relational data. We show visually that information about relationships between businesses can be extracted from the corpus (after some minor noise filtering). Then we apply more involved techniques to determine the "centrality" of the companies in an industry, as well as the relatedness of a company to any given industry.

Data Preparation

We based our analysis on a corpus of 22,170 business news stories from the 4-month period of 4/1/1999 to 8/4/1999, including press releases, earnings reports, stock market news, and general business news. As Figure 1 shows, we first applied an information extraction system by ClearForest, Ltd. (see www.clearforest.com) to extract both entities and relationships between them from the news stories and export them into a standardized XML-format. For this paper, we will concentrate only on business entities and only on the simple relationship: two businesses "co-occur" (are mentioned together) in a news story. This simple problem has analogies to many other problems involving text documents (two papers are cited in the same research paper, two potential terrorists are mentioned in the same intelligence report, etc.). The ClearForest information extraction engine is ideal for this extraction task: it outperformed all other information extraction tools in an evaluation by the Automatic Content Extraction program run by NIST that took place in February of this year (see see: http://www.itl.nist.gov/iad/894.01/tests/ace/),

[2] Such capabilities would be useful to a variety of different analytical jobs (e.g., noticing a new relationship between

particularly for the task of extracting entities from business news. The overall extraction resulted in a large database that contained among other things approximately 45,000 occurrences of company names. We also had to disambiguate the company names (for example, merging mentions of "HP," "H.P.," and "Hewlett Packard," but not "H. P. Hood"), resulting in a total of 1790 distinct companies.

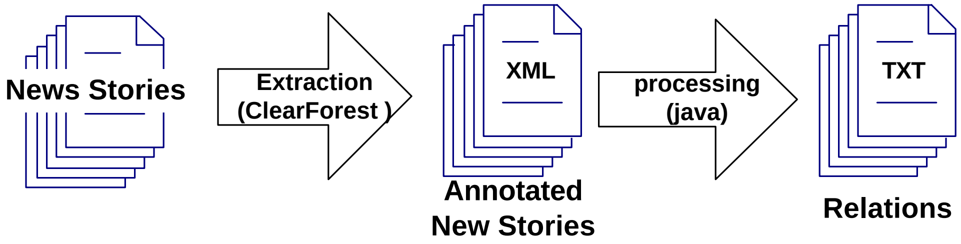

Figure 1: The Data Preparation Process

Compiling knowledge across documents

Assuming that we have a large set of simple, syntactic relationships between entities (extracted from news stories), we would like to be able to answer the following questions.

1. Can we identify which entities are (semantically) related to each other, for example because they belong to the same group? (Specifically, can we identify companies that are closely related to each other?)
2. If we can do (1), can we use this ability to identify the key players in an industry? (Specifically, can we find those who are in some sense "central" in the web of relationships?)
3. If we can do (1), can we use this ability to identify related entities in order to characterize entities by the different groups they are related to? (Specifically, can we use this information to create a relatedness measure that can act as a surrogate to industry membership?)

The method we use is similar, fundamentally, to many knowledge-technology successes, such as statistical natural language processing and traditional data mining. Namely, by taking advantage of a very large corpus of data, the aggregation of purely syntactic information can lead to the extraction of (shallow) semantic knowledge. For example, for machine translation, simply aggregating co-occurrences of words in a massive corpus of translated documents can yield remarkable translation performance—

a company and (say) the computer industry could improve the effectiveness of a financial analyst.

even when compared to manually constructed (and semantically based) translation systems. Because (by nature) the extraction is not perfect, such systems have had success in tandem with human analysts.

For our task, the knowledge of business relationships is "created" by aggregating simple co-occurrence relationships between entities, and by drawing conclusions statistically about semantic relatedness (e.g., group membership). Simple co-occurrence of two companies in a single news story does not necessarily mean that the companies are related in a general sense. For example, there are many stories that mention pairs of companies that are related in a way very specific to the current news ("Enron and Tyco both have questionable earnings practices") or that mention companies that are not related at all (e.g., "IBM and John Deere both issued earnings statements..."). However, statistical aggregation allows unimportant co-occurrences to act as noise, and important relationships to act as signal. Presumably the noise will be random (and will be cancelled out) and the signal will be regular (and will be amplified).

As a first evaluation of our conjecture, we visualize the relations inferred from the corpus, using a moderate level of noise filtering. Figure 2 illustrates the "important" relationships that remain after noise relationships are eliminated. Specifically, the nodes represent companies; the links relate companies that are mentioned together in 20 or more different stories. (Without the filter on noisy relationships, the graph is a mesh of spaghetti from which little can be inferred visually—but see below.) Recall that these relationships were created based on only the co-occurrences—with no explicit knowledge of industries. As a crude evaluation consider the shading of the nodes, which represents "industry" class as determined by two-digit SIC codes. Despite the (known) acute inaccuracies of SIC codes, similarly shaded nodes are related more often than would be expected if the relationships were random.[3]

As a second, more intuitive evaluation, consider the clusters of companies defined by the relationships. Moving counter-clockwise from the top center of the figure, one can see many clusters with clear semantic relationships, including rating agencies, a telecom cluster, investment banks, publishers, a

[3] If you have access to an electronic version of this paper, the shades are much clearer in color.

medium-sized airlines cluster[4] at the bottom center. Even in the heavily connected section on the right, there are clear sub-clusters. Continuing around counter-clockwise and spiraling in we see automobile manufacturers, a large telecom cluster, and the very large tech cluster. Within the tech cluster, we see dominant players (the hubs) IBM and Microsoft.

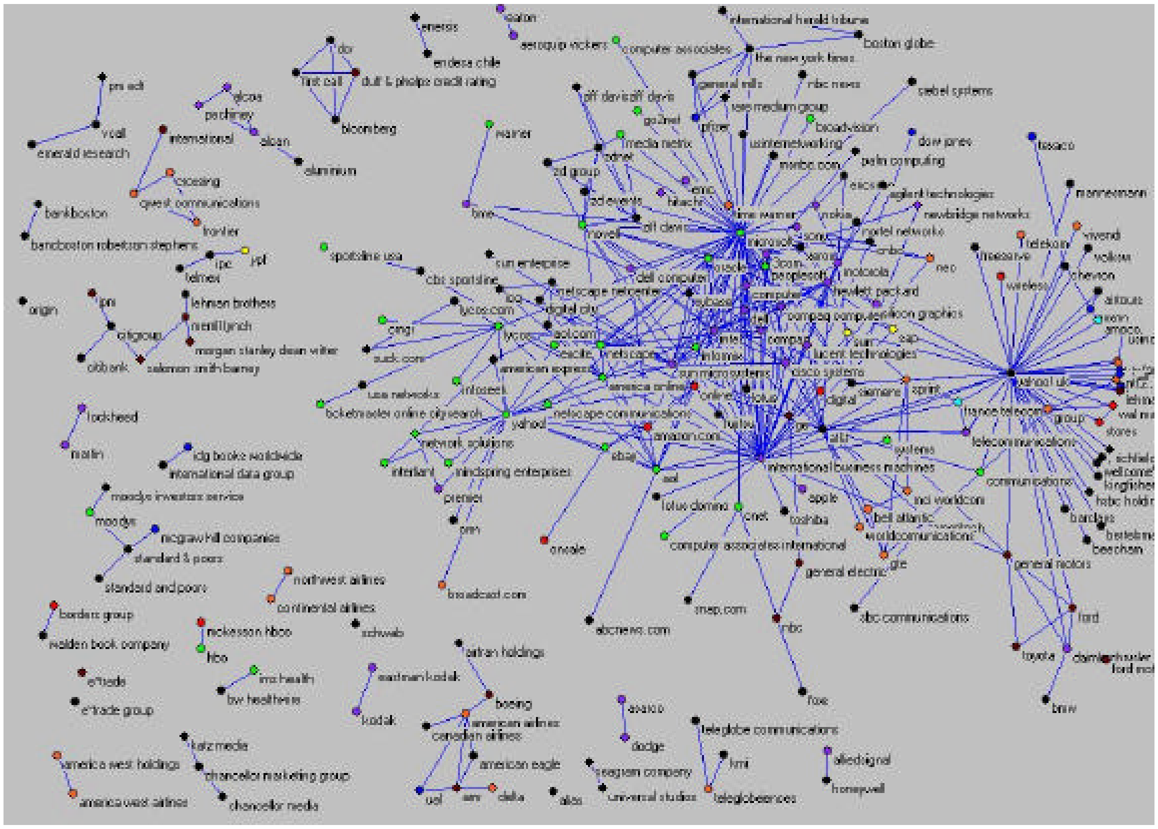

Figure 2: Filtered and labeled document-level co-occurrence data

This demonstration shows that the aggregation of relationships extracted from many documents, combined with a simple method for eliminating noise, results in the creation of knowledge about related companies.

We would like to go beyond this intuitive evaluation and provide more rigorous evaluations assessing whether the extracted knowledge indeed is meaningful. To this end the following two sub-sections

[4] The medium-sized airline cluster at the bottom illustrates a problem with using SIC codes to evaluate the quality of the relationships: Delta, AMR, American Eagle, and UAL have four different SIC codes.

provide numerical evaluations with increasing degrees of formality to confirm further that meaningful knowledge has been “discovered.” The first evaluation investigates whether a centrality measure applied to the co-occurrence graph can be used to identify central players of an industry. The results are surprisingly accurate, but remain difficult to evaluate formally. Finally, we show how a co-occurrence vector-space model for determining industry relatedness, analogously to the models used in information retrieval (Salton and McGill 1983) and collaborative filtering (Goldberg et al. 1992), can lead to a more rigorous confirmation.

Company centrality

The industry clusters shown in Figure 2 illustrate nicely how certain companies could be seen as central players in an industry (like IBM and Microsoft, the “hubs” to the right of center) and others are more peripheral (like Onsale, by itself, just below the center). Social network analysis (Scott 1991; Wasserman and Faust 1994), which was developed by scientists at the confluence of anthropology, sociology, and mathematics, provides a set of tools and measures for analyzing inherently relational data. In particular it provides so-called *centrality* measures, which describe the interconnectedness of social actors. In our context, social networks are composed of the social actors (the companies) and social relations (co-occurrences in news stories) represented as nodes and edges of a graph. We measure centrality simply as node degree—the number of relations any given actor is engaged in.

In analogy to Figure 2, we generate a graph to investigate the centrality of companies given the co-occurrence relationship. To filter noise we only consider relationships that are supported by more than one story (we no longer need to visualize the results, so we use a much more liberal threshold than we did for the visualization) resulting in a network with 315 companies and 1047 edges. Table 1 shows the 30 top-ranked companies in the computer industry (industry membership was determined by Hoover’s classification (Hoover’s 2002)), along with their centrality measures.[5] First, just by looking at the companies mentioned, note (intuitively) how well the centrality measure selects the more important players in the industry without any knowledge about the industry structure beyond the companies' co-occurrences in news stories. The final column (with the "X" marks) shows which of the companies are Fortune-1000 companies—note that these top-30 companies as ranked by centrality contain 16 of the total 24 Fortune-1000 companies in the technology-company list.[6] In sum, more than 50% of the top-30 "most central" technology companies are Fortune-1000 companies (the top-5 all are), as compared to less than 15% of the rest of the technology companies. This analysis provides a complementary view of the knowledge contained in the interrelationships extracted and aggregated from many documents.

Company Name	Centrality	Fortune
INTEL CORPORATION	500	X
MICROSOFT CORPORATION	406	X
COMPAQ COMPUTER CORPORATION	344	X
HEWLETT PACKARD COMPANY	336	X
AMERICA ONLINE INCORPORATED	322	X
NOVELL INCORPORATED	227	
NATIONAL SEMICONDUCTOR CORPORATION	212	X
3COM CORPORATION	183	X
CISCO SYSTEMS INCORPORATED	166	X
ADVANCED DIGITAL INFO CORPORATION	152	
ORACLE CORPORATION	146	X
INTEGRATED SILICON SOLUTION INCORPORATED	114	
MTI TECHNOLOGY CORPORATION	109	
META GROUP INCORPORATED	97	
SUN MICROSYSTEMS INCORPORATED	91	X
BROADVISION INCORPORATED	83	
HYPERION SOLUTIONS CORPORATION	70	
INTUIT INCORPORATED	64	
CABLETRON SYSTEMS INCORPORATED	61	X
INTERNATIONAL BUSINESS MACHS COR	59	X
ADOBE BUILDING CTRS INCORPORATED	56	
INGRAM MICRO INCORPORATED	53	
MICROSTRATEGY INCORPORATED	48	
DELL COMPUTER CORPORATION	44	X
PEOPLESOFT INCORPORATED	42	X
SILICON GRAPHICS INCORPORATED	39	X
ELECTRONIC DATA SYS CORPORATION	36	X
INTERGRAPH CORPORATION	36	
NETWORK APPLIANCE INCORPORATED	35	
3DFX INTERACTIVE INCORPORATED	32	

Table 1: Top 30 companies in terms of centrality from the computer industry

[5] In total, there were roughly 90 computer-industry companies that had co-occurrence relationships supported by more then one story.

[6] Recall that the news stories were collected in 1999. Consequently, we also chose the 1999 Fortune-1000 list.

Determining industry relatedness using a vector-space model

One of our domain experts (a business researcher) identified the determination of industry membership as being important knowledge to be able to discover in a timely fashion. Although our co-occurrence-based relationships are more general than simple industry membership, as we now show, a vector-space model can be applied to the relationships to create an effective proxy for industry membership.

In order to determine a company's relatedness to an industry we examine the similarity between the (normalized) vector representing a company's co-occurrences and an average vector for the industry. This approach is analogous to the vector-space model used in text classification, information retrieval, and collaborative filtering. The biggest difference to those approaches is that we use *relationships* between entities (which most probably came form a large number of different documents) as the elements of the vectors. Some advantages of this method are that it (1) allows the comparison between a company and an "average" vector for a whole industry, (2) allows one to look at the relatedness of whole industries in terms of their "average" vectors, and (3) provides a "relaxed" specification of a cluster of companies that allows flexible definitions of "industries" (or other groups).

For this analysis let us first identify a number of exemplar companies from different industries (see Table 2) using Hoover's classification.

Industry	Companies
Computer Software	Microsoft, IBM, ORCL, SAP, Computer Associates, Compuware, Seibel Systems, PeopleSoft, BMC Software
Computer Hardware	Compaq, Hewlett-Packard, IBM, Dell, NEC, Gateway, Apple, Acer
Integrated Oil	Hess, BP Amoco, Chevron, Texaco, Conoco, Exxon, Mobil, Shell, Elf Aquitaine
Major Drug Manufacturers	Squibb, Merck, Pfizer, Schering Plough, Warner Lambert, Johnson & Johnson, Smithkline Beecham, Glaxo Wellcome, Astrazeneca, Novartis, Abbott Laboratories, American Home Products.

Table 2: Exemplar Companies and their industries

Next we define how the co-occurrences are normalized and coded into vectors. In particular, we define the *direct* dot product (cosine) of an industry A with an industry B as a measure of relatedness between two industries. We use the term *direct* because it compares how companies in one industry directly co-occur with companies in another.[7]

The direct cosine between two companies is defined as follows:

$$\cos_{direct}(a,b,t) = a \bullet_t b = \frac{c_b(a\,|\,t)}{\sqrt{\sum_{a \in a} c_a^2(a\,|\,t)}} \qquad \textbf{(1)}$$

where *a* is a vector representing company *a* and the entries in the vector are the numbers of occurrences of company *a* with (all) other companies, $c_b(a|\tau)$ is the number of occurrences of company *b* with company *a* such that the number of occurrences is greater than or equal to the threshold τ (i.e., it is a function that chooses *b*'s entry in the vector *a* if it is greater than τ, and zero otherwise). Note that the sum in the denominator normalizes over all the companies co-occurring with *a*, which we will call the "basis set."

This definition leads to the definition of the dot product of a company with respect to an entire industry:

$$\cos_{direct}(a,B,t) = a \bullet_t B = \frac{1}{N_B} \sum_{b \in B} \cos_{direct}(a,b,t) \qquad \textbf{(2)}$$

where N_B is the number of companies in industry *B* and τ is the threshold as before. Finally, we can define the dot product of an industry with another industry as:

$$\cos_{direct}(A,B,t) = A \bullet_t B = \frac{1}{N_A N_B} \sum_{a \in A} \sum_{b \in B} \cos_{direct}(a,b,t) \qquad \textbf{(3)}$$

The above formula defines the dot product with respect to the first argument because the companies need not have the same companies in the basis set. Finally, note that *A•B?B•A* (necessarily), because the normalization allows relatedness to be asymmetric. For example, a small auto-parts supplier may be very strongly related to its single customer, General Motors; GM, on the other hand, may have a much weaker (relative) relationship to this small supplier than it does to its main competitor, Ford.

[7] We also have experimented with an indirect measure, which can compare two companies based on their vectors of

We can now compute a “relatedness” vector of a company with respect to a set of industries. The relatedness vector is defined as the vector of dot products over a set of industries:

$$\text{Relatedness}_t(X) = (\cos(X, A, t), \cos(X, B, t), \cos(X, C, t), \ldots) \quad \mathbf{(4)}$$

where *A*, *B, C*, are the industries and *X* is the company we are computing the relatedness for. Intuitively, relatedness should capture competition, collaborators (e.g., partners such as supply-chain relationships), and perhaps other sorts of relatedness. It is a more general relationship than a purely intra-industry relationships (e.g., competitors). However, as we will now show, it can be used remarkably effectively as a surrogate to industry-group membership.

Given these definitions let us now examine the direct cosine between our four industries. As Table 3 shows, the intra-industry relatedness values ($A \bullet_{t=1} A$) are higher than the inter-industry relatedness values ($A \bullet_{t=1} B$). The signal-to-noise ratios, $A \bullet A / A \bullet B$˜ 10, are impressive. Companies are most closely related to companies in the same industry.

Industry	Software	Computer hardware	Major drug manufacturer	Integrated oil
Software	.081	.055	.001	.002
Compute Hardware	.060	.094	.001	.001
Major Drug Manufacturer	.009	.008	.029	.006
Integrated Oil	.004	.005	.002	.075

Table 3: Average direct cosine between industries with t = 1

A notable exception is between the Software and Hardware groups. Although intra-group relatedness still is higher generally, it is clear that these industry groups are closely related to each other. Individual companies may in fact be more closely related to groups besides their own. For example, Microsoft’s relatedness vector (see equation (5)) shows that the company is slightly more closely related to the hardware industry than to the software industry. This observation makes sense: Microsoft’s main customers are in the computer hardware industry and it (arguably) has closer business relationships with hardware manufacturers than with software manufacturers.

co-occurrences—but we don’t have the space to present it here.

Relatedness(MSFT) =
(cos(MSFT, Software), cos(MSFT, Hardware),cos(MSFT, Drugs), cos(MSFT, Oil)) = **(5)**
(.041, .054, .002, .001)

Figure 3 shows the effect of the noise-filtering threshold τ on the signal-to-noise ratio for three of the industries (*S*=software, *O*=oil, and *H*=hardware). We can see that with the exception of the software and hardware industries, the ratio improves (sometimes dramatically) as we raise the threshold. The tradeoff of course is that that some signal is lost as the threshold is raised.

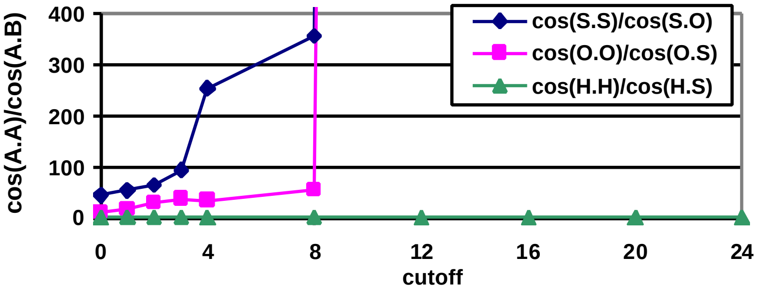

Figure 3: Signal to noise of three industries as a function of the threshold t

Using a Kuipers test (Kuipers and Niederreiter 1974), the null hypothesis that the two distributions (i.e., "signal" and "noise") are drawn from the same distribution is rejected for the software and drug industries (at p = 7x10^{-7}) as well as for the software and oil industries (at p=.0004). Only for the comparison between the software and the hardware industry can we not reject the null hypothesis (p=.6860); this shows (more rigorously) that those two industries are not as distinct from each other in terms of our relatedness measure.

Given the results shown here we can conclude that the direct cosine measure based on co-occurrences can serve as an adequate (surrogate) measure of industry membership, when industries are distinct (e.g., as in the oil vs. software case). When industries are strongly intertwined (as in the hardware vs. software industry) the measure is not very discriminative. In this particular case, however, we have to ask for what purposes the strong distinction between hardware and software industry companies is meaningful. For example, many companies produce both software and hardware; furthermore, for tasks like financial analysis, closely related hardware and software companies arguably will have similar stock-market performance (for example) than distantly related companies in one group or the other. Consequently, the vector-space model presented here may be more useful for some tasks (we have not shown this). We have shown that it can used to take advantage of relational information, which was initially distributed over a number of documents, to produce meaningful "knowledge."

Discussion

The analysis above confirms that using purely syntactic and statistical processing, we have automatically extracted non-trivial semantic knowledge about company interrelationships. This knowledge had been distributed over a large number of business-news documents. Similarly to the case (discussed above) with machine translation, we have in effect compiled the knowledge of the many authors of the news stories—in this case, about business interrelationships.

There are a number of limitations (and potential improvements) to our analysis. First, we have been limited by the data set. In particular, the time-period of its collection (in the middle of the Internet bubble) provided us with a distribution highly skewed to technology-related news. Some analyses (which we did not have the space to present) into other methods for industry classification gave encouraging results, until they were compared to simply predicting all companies were technology companies. To address this problem we have begun to experiment with a larger corpus of news from a different time period.

We have not yet produced satisfactory results using more sophisticated relational data-mining techniques, such as inductive logic programming (Dzeroski and Lavrac 2001) or probabilistic relational modeling (Friedman et al. 1999). We have only just begun to investigate this. Although relational learning approaches have been applied to text, for example, to create information extraction systems (Califf and Mooney 1999; Cohen 1995), we are not aware of their application to relational learning based on the aggregation of relationships from many documents.

In the present work, we have used existing industry classifications (mainly Hoover's) as a gold standard against which to compare the "relatedness" mined from the business news corpus. One criticism of the work presented here might be that the determination of industry relatedness is a rather simple task.

One simply could look to Hoover's, or could examine SEC documents. While this true to a certain extent, the approaches presented can compute the (approximate solution) very fast and cheaply, for any time period for which news is available, and could be used (for example) to monitor for structural changes not yet reflected in any manually created database (e.g., a company enters a new industry). It also was a discovery task for which we had two vital elements: expert interest and a gold standard for comparison.

Although this news-relatedness can be used as a(n approximate) surrogate for the task of industry classification, it is not identical. It would be interesting to investigate further the actual knowledge comprised by this relationship. However, it is important to point out that the vector-space model is more flexible than a precompiled industry classification. The basic industry vectors could be defined arbitrarily by users, and the system would give a distribution of relatedness to whatever vectors are given (e.g., the companies present in different sector mutual funds). The efficacy of doing so would be dependent on the particular task at hand.

Obviously, this list of limitations is incomplete, but it does highlight that the investigation of the automatic discovery of relational knowledge based on extractions from large textual corpora is a promising field for much research. Consider, for example, the massive "knowledge base" of business relationships that would be created if this study were scaled up to millions of documents as well as additional types of relationships (which the extraction software generates, but we ignored for this study).

In summary, in this pilot study, we have shown that automated techniques can be used discover knowledge from relationships distributed over a large number of business-news documents. Techniques for extracting information automatically from massive collections are becoming increasingly important as people face larger amounts of information they cannot absorb. We have concentrated here on validating the techniques by comparing them to existing "knowledge" compiled separately (e.g., by Hoover's). An intriguing direction for future work is to search for new knowledge (discoveries). By compiling and linking knowledge originally held by multiple, different news-story authors, can we discover new things? Although we have no evidence yet that there are such discoveries to be made, the fact that the method can recreate existing knowledge is promising.

Acknowledgements

Thanks to ClearForest, Ltd. for the use of their information extraction system, and to Ronen Feldman for useful discussions.

References

Bron, C., and Kerbosch, J. "Finding all cliques of an undirected graph," *Communication of the ACM* (16), 1973, pp. 575-577.

Califf, M. E., and Mooney, R. J. "Relational Learning of Pattern-Match Rules for Information Extraction," Proceedings of the 16th National Conference on Artificial Intelligence (AAAI-99),, Orlando, FL, 1999, pp. 328-334.

Cohen, W. W. "Learning to Classify English Text with ILP Methods," Proceedings of the 5th International Workshop on Inductive Logic Programming, L. De Raedt (Ed.), Leuven, Belgium, 1995, pp. 3-24.

Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., and Slattery, S. "Learning to Extract Symbolic Knowledge from the World Wide Web," Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98),, Madison, WI, 1998, pp. 509-516.

Dzeroski, S., and Lavrac, N. *Relational data mining*, Springer, Berlin; New York, 2001.

Feldman, R., and Dagan, I. "KDT - knowledge discovery in texts," Proceedings of the First Annual Conference on Knowledge Discovery and Data Mining (KDD),, Montreal, Canada, 1995.

Friedman, N., Getoor, L., Koller, D., and Pfeffer, A. "Learning Probabilistic Relational Models," Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI),, Stockholm, Sweden, 1999, pp. 1300-1307.

Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. "Using collaborative filtering to weave an information tapestry," *Communication of the ACM* (35:12), December 1992, 1992, pp. 61-70.

Hearst, M. A. "Untangling Text Data Mining," Proceedings of the 37th Annual Meeting of theAssociation for Computational Linguistics,, University of Maryland, June 20-26, 1999.

Hoover's. *Hoover's company profile database*, [Internet], Hoover's, Inc, Available: http://www.hoovers.com/ [2002, May 2](2002).

Kuipers, L., and Niederreiter, H. *Uniform Distribution of Sequences*, Wiley, New York, NY, 1974.

Macskassy, S. A., Hirsh, H., Provost, F., Sankaranarayanan, R., and Dhar, V. "Intelligent Information Triage," Proceedings of the 24th Annual International Conference on Research and Development in Information Retrieval,, 2001.

Marshall, C. C., and Shipman, F. M. "Spatial hypertext and the practice of information triage," Proceedings of the Eighth ACM conference on Hypertext and Hypermedia,, Southampton, United Kingdom, 1997, pp. 124 - 133.

Salton, G., and McGill, M. J. *Introduction to modern information retrieval*, McGraw-Hill, New York, 1983.

Scott, J. *Social Network Analysis: A Handbook*, Sage Publications, Newbury Park, CA, 1991.

Swanson, D. R., and Smalheiser, N. R. "Assessing a gap in the biomedical literature: Magnesium deficiency and neurologic disease," *Neuroscience Research Communications* (15), 1994, pp. 1-9.

Wasserman, S., and Faust, K. *Social network analysis : methods and applications*, Cambridge University Press, New York, 1994.

Yang, Y. "An evaluation of statistical approaches to text categorization," *Information Retrieval* (1:1-2), 1999, pp. 69-90.