Discovering Relationships by Feature Extraction from Text Databases

W. Bruce Croft & Mary H. Utt

Computer Science Technical Report 93-61
July 1993

Discovering Relationships by Feature Extraction from Text Databases

W. B. Croft and M.H. Utt
Computer Science Department
University of Massachusetts, Amherst

Abstract

An approach to accessing text-based information that emphasizes domain-specific features rather than documents is presented. The basis of this approach is the ability to automatically extract features from large text databases, and generate the significant associations or relationships between those features. The techniques for doing this are discussed, and examples of the operation of these techniques on a database of 46,449 Wall Street Journal articles are presented. In this application, the features extracted are company and person names and associations between them. A technique for searching for features to use as starting points for browsing is also discussed. The tests with the database of newspaper articles show that feature extraction is reliable, and that the relationships generated are reasonable. They also demonstrate that it is more difficult to evaluate systems of this type than standard information retrieval systems.

Keywords

indexing, feature extraction, link generation, information retrieval

1 Introduction

Information retrieval systems are primarily *document-oriented* in that the architecture of these systems is based on the storage, retrieval, and display of text documents, such as newspaper articles and scientific abstracts. Many hypertext systems have inherited this view of information in that hypertext nodes are typically short text documents, possibly derived from longer sources. In some systems, browsing of index terms associated with the text documents is also supported [9, 1], but this is generally regarded as a secondary activity in relation to the primary task of locating the relevant documents (or text nodes).

From the users' point of view, it is nearly always the information contained in the text documents that is the goal of the search, not the documents themselves. In some application domains, this information is sufficiently well-defined that it should be possible to build retrieval and browsing systems based on this information rather than the text documents that contain it. Such systems should make it easier to answer important classes of queries and may be able to support types of access that would be impossible in a document-based system.

The techniques described in this paper are the basis for an information system that moves away from a document-oriented perspective by focusing on the recognition of domain-

specific features and relationships between those features in text. These features and relationships become the primary focus of the system, with the text documents being used as backup data that provides more context for the relationships that are discovered.

The specific application studied here is the recognition of companies and people, and relationships between them, in newspaper texts. In this application, the types of information access that should be supported include retrieval of companies and people by name and by activity, and browsing networks of companies and people to discover relationships that may not be obvious in a document-oriented system.

In the following sections, we describe techniques and experiments in three major areas needed to support this application. These are

- *Automatic feature extraction* - techniques used to recognize features in large, free-text databases.
- *Generating links* - techniques for quantifying the relationship between features based on their association in free text.
- *Finding starting points* - techniques for "indexing" features and retrieving possible starting points for browsing in a network.

The experiments use a database of one year (1987) of Wall Street Journal articles. This consists of 46,449 articles containing 249 words on average, and is a part of the TIPSTER document collection [6]. Evaluating some of the proposed techniques is more difficult than a typical information retrieval experiment, and this issue will be discussed throughout the paper.

2 Automatic Feature Extraction

The problem of feature or fact extraction from unrestricted text has been studied by a number of researchers in the context of the Message Understanding Conferences [7] and now the TIPSTER project. The basic approach has been to use a variety of natural language processing and statistical techniques to extract predetermined types of facts for a specific domain. In the TIPSTER joint venture domain, for example, the goal of extraction is to identify information such as the companies forming the joint venture, the name of the new company, the location of the new company, the products of the new company, and the amount of money involved.

Accurate extraction of some types of information requires either sophisticated analysis or significant amounts of training data. There are, however, a significant number of important, and fairly general, features which can be recognized using relatively straightforward techniques. These include, for example, the names of companies, the names of people, locations, monetary amounts, and dates. Much of this could be described as the recognition and categorization of proper nouns. High rates of accuracy are possible because of the relative simplicity of the task. It is, for example, much easier to recognize the presence of a company name in an article about a joint venture than to identify the role that company is

playing. It is our contention that the ability to recognize these simple features can be used to develop powerful new approaches to accessing information.

For the application we are concerned with in this paper, the two feature recognizers needed are for company names and peoples' names. The techniques used for these recognizers are typical of those we would use for most simple features, which is a combination of lexical scanners built using *lex* or a similar tool, and table lookup.

The company recognizer scans the text for proper nouns (capitalized words) that have the appropriate format for a company name. Company names often include special words such as *Inc.*, *Corporation*, or *Pty. Ltd.* that are particularly useful for recognition [8]. In a given text document, the recognizer will use these special words to recognize the first mention of a company name and store it in a temporary table. This table permits the recognition of subsequent uses of that company name, even if the special words are not used. In the Wall Street Journal, for example, the first use of a company name in a story often uses the full form.

In a simple test of the company recognizer, we ran it on a sample of the Wall Street Journal database and compared the results to company names identified manually. The test database consisted of 139 articles containing 29,000 words. The manual scan of the database identified 334 company names. In this test, the precision (percentage of names identified as companies that actually were companies) was 89% and the recall (percentage of company names in the sample that were identified as companies) was 79%. Many of the precision errors were caused by two difficult company name formats where names are combined used 'and' and 'or', such as in *X, Y and Z Corporation* and *X of Y Inc.* Although these can be valid formats (e.g. Mutual of Omaha), they tend to introduce too many errors. We are currently revising the company recognizer to improve the recall by introducing a company name table that will contain common names and synonyms (e.g. IBM, Digital Equipment/DEC). This modification is based on the observation that references to well-known companies are more likely to not use the full form of the name.

The person name recognizer works in a similar fashion but places more emphasis on table lookup. A name is recognized when a capitalized sequence of words contains a title such as *Mr.*, *Senator*, *President*, and so forth. In addition, lists of first names and last names are used to identify names that do not contain titles. As in the case of the company recognizer, later references to people in the same story are recognized, even if the full name is not used. Checks are made to ensure that a recognized name is not a company name (for example, that John Blair Company, later in a story referred to as "John Blair" is not recognized as a person name.

Finally, because sequences of capitalized words may contain other words in addition to a name, a stop list of common problem words is maintained. For example, the Wall Street Journal frequently begins sentences with constructions such as *Added Joe Smith ...* or *Investor Jane Doe ....* We are investigating whether dictionary lookup may be more effective (and general) for this purpose.

To test the person name recognizer, we used the same database as the evaluation of the company recognizer. The manual scan identified 269 person names in the sample text. The

name recognizer achieved 92% precision and 93% recall. Many of the errors were due to inappropriate names in the first and last name tables. As an example, we are modifying the recognizer to avoid identifying locations (Santa Monica, Carson City) as person names.

Recognizing synonymous names is also a problem. For example, Bill Clinton and William Clinton are currently recognized as two different people. For two names in different stories to be recognized as the same, we have specified that the first name, middle initial (if any) and last name have to be the same. Given that we want to make connections between people and companies, the resolution of name ambiguity needs to be addressed. A table of common synonyms will help, but it is not the whole solution. Fortunately, the company-person connections themselves can provide significant additional evidence that two names are the same. If, for example, Roger Smith and Roger B. Smith are both highly correlated with GM, it is likely that they are the same person. This technique is used to conflate person-company links after they are generated using the techniques described in the next section.

Despite the complexities involved in recognizing company and person names, it seems reasonable to conclude from our test data (and other peoples' experiments) that these names can be reliably recognized in text.

3 Generating Links

Having identified companies and people in the text, the next step is to discover relationships or associations between them. These relationships can be used as the basis for the links in a hypertext network of companies, people, and source texts. Relationships can be identified using either direct or indirect associations. A direct association occurs when the company and/or people names occur 'close' to each other in the text. Closeness can be measured either by a simple distance measure (how far apart are the names) or using some linguistic context (for example, in the same sentence or in a subject-object relationship). In our previous work [5], word distance has been shown to be the strongest evidence for the presence of phrasal relationships, so for this study we have concentrated on name distance as the primary measure of association.

An indirect association occurs when the company and people names have similar words associated with them. In the next section, we discuss how word contexts can be used to derive these associations and support the direct retrieval of companies and people.

In this study, we used text *windows* to define direct associations. The window refers to the number of words on either side of a target feature (company or person name). For example, a window of size 11 would include the target feature, the five preceding words, and the five succeeding words. The window sizes used here were 51 words and 201 words. These sizes were chosen to compare paragraph-level and document-level associations.

The strength of a relationship between two features depends on the number of associations (co-occurrences in a window) and how common the features are in the whole database. For example, a single co-occurrence of GM and IBM in a text window is not likely to be significant given how frequently these companies are mentioned in the database. To determine

significant associations (or to rank the discovered relationships by presumed importance), we use two statistical measures, the expected mutual information measure (*EMIM*) [4] and ϕ^2 [3].

The expected mutual information measure compares the probability of observing two features, x and y, together to the probability of observing the two features independently. In this paper, we use a simplified version of this measure that ignores terms involving probabilities that features do not occur. The measure used is

$$EMIM(x,y) = \log \frac{P(x,y)}{P(x)P(y)}$$

When a strong relationship exists between the features, the joint probability (P(x,y)) will be greater than chance and $EMIM(x,y)$ will be greater than 0.

The calculation of *EMIM* (and ϕ^2) makes use of the so-called contingency table. This table can be represented as follows:

	y	$\bar{y}$
x	a	b
$\bar{x}$	c	d

The upper-left-hand cell (a) records the number of times features x and y co-occur in a window. Cell (b) records the number of times x occurs but y does not. Similarly, cell (c) records the number of times y occurs but x does not. Finally, cell (d) records the number of times neither feature occurs. Given this table to estimate probabilities, the *EMIM* calculation is:

$$EMIM(x,y) = \log \frac{a}{(a+b)(a+c)}$$

The ϕ^2 measure has been suggested as an alternative to *EMIM* and will tend to favor high-frequency events more. This is calculated as follows:

$$\phi^2 = \frac{(ad-bc)^2}{(a+b)(a+c)(b+d)(c+d)}$$

The Experiment

The two measures described were used to compute associations between all features (companies and people) that co-occurred in text windows. The following is general data about the Wall Street Journal database (WSJ87):

- Number of words in the collection: 11,550,222
- Number of person names identified by recognizer: 45,285
- Number of company names: 25,816

Two different window sizes were used. For window size 51, there were 226,475 windows. For size 201, there were 54,464 windows. Since the second window size is close to the average size of a document, the number of windows is similar to the number of documents in the database.

Tables 1 and 2 list the number of pairs of features (n) that co-occurred m times within windows of size 51 and 201 respectively, for $m = 1$ to 10. The tables also list the maximum number of co-occurrences.

Evaluating the associations produced is difficult. The top ranked associations for *EMIM* and ϕ^2 appear to be very similar. For example, 58 of the top 68 ϕ^2 scores for company-person associations are also among the top 68 *EMIM* scores. We wrote a program that displays the top ranked associations for a given company or person for either *EMIM* or ϕ^2, and experience with this program confirmed that, although there were not large differences between the measures, the ϕ^2 measure did seem to favor high frequency associations. Since this often corresponds with intuition about relationships, the ϕ^2 measure is probably the better choice.

This type of evaluation is rather unsatisfactory. It also does not answer the question of whether the relationships discovered are "good". It is not clear whether it would be possible to develop a list of "relevant relationships" as is done with relevant documents in IR tests. Certainly, it is clear that the approach works to the extent that obvious relationships are found (such as between CEOs of a company and the company) and that all the relationships discovered have a basis in the text documents. Spurious relationships only occur at low frequency co-occurrences (mainly $m = 1$) and with larger window sizes. The data for the window size of 201 shows that many more co-occurrences are found. Some of these will be useful and some will be chance co-occurrences. The most useful window size will probably depend on the application and the users.

As an indication of the type of relationships discovered, the following are examples of the simple associations display program. This program is not intended as an example of the type of interface that should be used to discover relationships, but instead was used to determine if the associations seemed reasonable.

The first interaction shows the types of choices that are available. It shows the companies that are highly associated with "Donald Trump". The first column in the list of companies is the frequency of co-occurrence, the second column is the association measure (either *EMIM* or ϕ^2.

```
Choose one of the following:

        1. Person-person associations
        2. Person-company associations
        3. Company-company associations
        4. Company-person associations
        5. Exit

Enter the number of your choice: 2
```

Number of times (m) a pair of features co-occurred	Number of pairs of terms (n) that co-occurred m times		
	person-company and company-person	person-person	company-company
1	74,950	81,890	102,404
2	28,880	28,924	20,622
3	11,084	11,654	9,206
4	6,336	7,264	4,780
5	3,512	3,790	2,534
6	2,618	2,776	1,774
7	1,740	1,908	1,182
8	1,362	1,434	916
9	1,118	992	670
10	852	720	562
maximum value of m	149	145	150

Table 1: WSJ87 Co-occurrence Data (Window size = 51)

Number of times (m) a pair of features co-occurred	Number of pairs of terms (n) that co-occurred m times		
	person-company and company-person	person-person	company-company
1	135,112	129,430	197,642
2	60,902	65,382	36,442
3	23,584	26,944	15,112
4	16,576	22,122	9,646
5	8,204	9,898	5,616
6	7,960	10,600	4,986
7	4,544	5,410	3,228
8	4,572	5,390	2,824
9	2,890	3,578	2,134
10	2,632	3,078	1,626
maximum value of m	200	200	199

Table 2: WSJ87 Co-occurrence Data (Window size = 201)

```
Person-Company Name Associations

Enter the person name: Donald Trump

        The person name to search for is Donald Trump

Choose one of the following orders for output:
        1. EMIM
        2. Phi-squared
Enter the number of your choice: 1

78 matches
Enter threshold for co-occurrences: 8

       47      8.360000        Bally Manufacturing
      136      8.280000        Golden Nugget
       39      8.180000        Resorts International
       11      7.480000        Elsinore
       96      6.780000        UAL
       11      6.770000        Pratt Hotel
       24      5.150000        Holiday
        8      4.680000        Marriott
       14      4.306184        Lazard Freres And Salomon Brothers
       15      3.966487        Caesars World
       24      3.280000        Allegis
       17      2.850000        Pan Am
        9      2.390000        Bear Stearns
13 displayed
```

The second interaction starts from one of the companies that are connected to Trump ("Golden Nugget") and finds related companies.

```
Company-Company Name Associations

Enter the first company name: Golden Nugget

Choose one of the following orders for output:
        1. EMIM
        2. Phi-squared
Enter the number of your choice: 2

13 matches
Enter threshold for co-occurrences:

      24      0.300016        GNF
```

```
      33      0.066514        Bally Manufacturing
      20      0.015471        Golden Nugget Finance
       8      0.008500        Resorts International
       4      0.004600        Elsinore
       7      0.002000        Caesars World
       2      0.000500        Janney Montgomery Scott
       2      0.000500        Hilton Hotels
       5      0.000200        Bear Stearns
       2      0.000100        Holiday
       4      0.000100        Allegis
       2      0.000000        Investors Service
       3      0.000000        Drexel Burnham Lambert
13 displayed
```

The list of companies contains some of the same ones as before, but also points in new directions. The third interaction looks for people associated with Golden Nugget, and of course we expect to see Trump again.

```
Company-Person Name Associations

Enter the company name: Golden Nugget

Choose one of the following orders for output:
        1. EMIM
        2. Phi-squared
Enter the number of your choice: 2

30 matches
Enter threshold for co-occurrences: 5

      30      0.221947        Stephen Wynn
     136      0.185900        Donald Trump
      31      0.150100        Michael Wynn
      14      0.104724        Steven Wynn
      24      0.099700        Frank Sinatra
       6      0.019200        Trump Castle
       5      0.006100        Marvin B Roffman
       7      0.005000        Henry Gluck
      10      0.002300        Martin T Sosnoff
       5      0.001000        Kirk Kerkorian
10 displayed
```

Note that the two spellings of "Stephen Wynn" are not conflated, and that "Trump Castle" has been mistaken for a person name. The fourth interaction repeats the same query, but uses *EMIM* instead of ϕ^2.

```
Choose one of the following orders for output:
        1. EMIM
        2. Phi-squared
Enter the number of your choice: 1

30 matches
Enter threshold for co-occurrences: 5

       14       10.725982       Steven Wynn
       30       10.710040       Stephen Wynn
       31       10.100000       Michael Wynn
       24        9.880000       Frank Sinatra
        6        9.500000       Trump Castle
      136        8.280000       Donald Trump
        5        8.130000       Marvin B Roffman
        7        7.360000       Henry Gluck
       10        5.730000       Martin T Sosnoff
        5        5.490000       Kirk Kerkorian
10 displayed
```

Note that the lists are very similar but ϕ^2 favors the higher frequency associations. The next interaction shows the people who are strongly associated with Trump. Again, we see some of the same names and some new ones.

```
Person-Person Name Associations

Enter the person name: Donald Trump

 174 matches
Enter threshold for co-occurrences: 5

       17       0.741100        Bob Halloran
       18       0.383400        Don King
       37       0.143600        Willard C Howard
       56       0.098400        James M Crosby
       26       0.080385        Tony Schwartz
        6       0.061500        Bob Arum
       14       0.051400        Jerome Palma
       49       0.035600        Jack E Pratt
       15       0.032100        Steven Roth
        6       0.028300        Jacques Lou
       15       0.027200        Felix Rohatyn
        6       0.026300        Seth Abraham
       10       0.026200        I G Davis
       27       0.021499        Trump Tower
        8       0.020100        Charles E Murphy
        5       0.019700        George N Aronoff
        5       0.019700        G M Brown
```

```
        5       0.019700        Anthony Gliedman
        5       0.019700        Harvey D Myerson
       18       0.019546        Stephen Wynn
Type M for more, Q to quit..
```

The next interactions follow up some of the connections, based on a company seen in the first interaction and a person ("Kerkorian") seen more than once.

```
Company-Person Name Associations

Enter the company name: Bally Manufacturing

19 matches
Enter threshold for co-occurrences: 3

      47       0.067900        Donald Trump
       3       0.006764        Stephen Wynn
       4       0.001900        Kirk Kerkorian
3 displayed

Person-Company Name Associations

Enter the person name: Kirk Kerkorian

52 matches
Enter threshold for co-occurrences: 5

       69       0.821500        Tracinda
       61       0.262520        Mgm Grand Hotels
      144       0.152900        Pan Am
       22       0.100100        Mgm UA Entertainment
        6       0.074500        Regent Air
       14       0.039200        Mgm UA Communications
        5       0.025900        Summa
        5       0.017200        Dunes Hotels & Casinos
        5       0.002800        Hilton Hotels
        5       0.001100        Transamerica
        5       0.001000        Golden Nugget
        7       0.000900        Braniff
        6       0.000700        UAL
        9       0.000500        Drexel Burnham Lambert
14 displayed
```

4 Finding Starting Points for Browsing

In the previous section, the starting points used to examine relationships were company and person names. It is possible to develop a more general approach that can be used to locate starting points (companies and persons) by a search based on activities. The same approach can also be used to calculate indirect associations on demand.

The basic technique involves identifying all the paragraphs that a given feature occurs in for the entire database. The feature is then "indexed" using those paragraphs. For example, if Company X is mentioned in 35 paragraphs in the database, the words in those paragraphs are used to index Company X. In that sense, the feature is treated as a special type of document and a feature retrieval database can be created, analogous to a document retrieval database. In the experiments described here, the INQUERY text retrieval system [2] was used to store feature representations and to retrieve features.

The words that are used to represent features are all the non-stopwords in the paragraphs, weighted by their frequency of occurrence in the paragraphs and in the database as a whole. Currently, no limit is put on the number of words in the representation of a feature. Very frequent people and companies, therefore, may have thousands of words in their representations. It is an open research issue to determine if there are optimal representation sizes and whether the entire database or some subset of it should be used to derive these representations.

Given a "feature database", a number of possibilities exist. One is that the database may be used to locate people and companies by activities. Queries can be submitted directly to the feature retrieval system (in our case, INQUERY) and ranked lists of companies and people will be produced. The features retrieved will be those whose representations contain the words in the query with high weights. We could, for example, ask for "companies active in oil exploration and alternative energy" and retrieve companies whose names will have co-occurred frequently with the phrases "oil exploration" and "alternative energy" in paragraphs in the database.

Another possible use of the feature retrieval system would be to use the representations of features as queries in order to retrieve features with similar representations. We could, for example, generate company-company *indirect* associations this way. This can be done efficiently enough to be carried out on demand during the user's interaction with the system.

A feature retrieval system was generated for the WSJ87 database using the INQUERY system. As before, the output of this system is difficult to evaluate. It appears that the features retrieved are reasonable. For example, companies retrieved using queries are related to the topics mentioned in the queries. More rigorous evaluation will require the cooperation of users in a particular application of the techniques.

The following are examples of the operation of the feature retrieval system. Note that the format of the output of this system is quite different to the associations program shown before.

The first query is an attempt to find people and companies who are associated with gambling and entertainment. The first number on each line is the rank, the second is the

similarity to the query, and the number in parentheses after the feature is the number of occurrences of that concept in the database. Note that this list contains both companies and people.

```
Query: gambling entertainment

 1. 0.571  mgm (49)
 2. 0.564  mgm/ua communications (61)
 3. 0.563  tracinda (31)
 4. 0.544  stuart perlman (2)
 5. 0.544  flamingo hilton hotel-casino (3)
 6. 0.544  stan fulton (2)
 7. 0.544  dennis gomes (4)
 8. 0.544  john t. makuch (2)
 9. 0.519  prism entertainment corp. (2)
10. 0.519  home entertainment group (10)
11. 0.519  alan m. levin (3)
12. 0.519  ronald segel (2)
13. 0.519  mgm/ua entertainment unit (2)
14. 0.519  southbrook entertainment corp. (2)
15. 0.519  meadowlands hilton (2)
```

The next query is a refinement of the first in that it adds "real estate development" to the list of activities, and we limit the output to person names. Not surprisingly, this time we find Donald Trump.

```
Query: gambling entertainment real estate development

 1. 0.513   david b. hilder (109)
 2. 0.525   webb (174)
 3. 0.522   donald trump (107)
 4. 0.513   roy j. harris (94)
 5. 0.505   charles e. cobb (14)
 6. 0.503   roy e. disney (51)
 7. 0.495   minami shoji (7)
```

The next query shows the ability of the system to find indirect associations. Many of these are the same as those found with direct associations. The query is the company "Golden Nugget".

```
Query: golden nugget

 1. 0.691  clyde turner (2)
 2. 0.691  steven wynn (3)
```

```
 3. 0.691  stephen wynn (4)
 4. 0.691  joel r. jacobson (2)
 5. 0.691  park place casino (2)
 6. 0.691  gnf corp. (2)
 7. 0.659  sinatra (28)
 8. 0.647  trump castle (14)
 9. 0.630  harrah (31)
10. 0.629  trump plaza (6)
11. 0.629  sands (49)
12. 0.616  clyde t. turner (4)
13. 0.616  kenny rogers (4)
14. 0.611  donald trump (107)
15. 0.603  henry gluck (20)
16. 0.589  diana ross (8)
17. 0.585  kirk kerkorian(24)
```

The final interaction shows what happens when a person name is used as a starting point.

```
Query: donald trump

 1. 0.653  councilman harold mosee (2)
 2. 0.629  chairman beryl anthony (3)
 3. 0.618  harvey i. freeman (5)
 4. 0.615  ivana trump (6)
 5. 0.613  trump parc (7)
 6. 0.608  trump plaza (28)
 7. 0.606  taj (65)
 8. 0.605  harvey myerson (6)
 9. 0.605  david g. marshall (6)
10. 0.603  james m. crosby (46)
11. 0.580  mayor koch (19)
12. 0.579  nugget (255)
13. 0.573  castle (234)
14. 0.573  marvin b. roffman (19)
15. 0.553  janney montgomery scott (98)
16. 0.546  wynn (119)
```

5 Building a System

As a summary of the process we are suggesting for building a system based on extracted features and relationships, the following is a description of the steps involved.

1. The first step is to parse the document database, identifying features and words. The output of this process is a collection of transactions. The transactions identify the position in the texts of the features and the words that occur in the same paragraphs as features.

2. The first set of transactions is sorted and used to build the inverted lists for the calculation of the association measure (probably ϕ^2). Each inverted list corresponds to a particular feature and lists the documents and positions in the documents in which the feature occurs.

3. The association measures between all feature types are calculated.

4. The other set of transactions is sorted and used to create the feature retrieval system. In this system, there is an inverted list for each word that co-occurs with a feature, and the list records the features that the word co-occurs with and the frequency of co-occurrence.

5. The feature associations and feature retrieval system are used as the basis of a browsing and text retrieval system. This system would support retrieval of people and companies, browsing of links between people and companies, and accessing related texts.

A number of these steps are computationally expensive. For example, the parsing step took approximately 5 hours on a Sun 490 for the WSJ87 collection. Steps 3 and 4 are worse. The computation of the association measures for WSJ87 took approximately 30 hours. We are currently studying ways of improving the efficiency of these steps, including the use of parallel hardware.

6 Conclusion

The approach presented here attempts to support new ways of accessing text. By focusing on extracting features and relationships between features, we hope to be able to address information needs that could not be handled with queries in a typical text retrieval system. An example of this would be discovering that company X is related to person Y, who is in turn strongly related to company Z. Discovering the relationship between companies X and Z may be difficult in a text retrieval system because the two companies do not necessarily get mentioned in the same stories, and the user who must formulate the query will probably not know that they are even looking for stories about companies X and Z until they see this relationship exists.

The person-company application described here is addressing a real user need, and there are many other potential applications of these techniques in different domains, such as medical systems and intelligence analysis. The main problems that need to be addressed are the computational costs of generating associations and indexing features, and developing an evaluation method for tuning the performance of such systems.

Acknowledgments

This work was supported in part by the NSF Center for Intelligent Information Retrieval at the University of Massachusetts. Mark Pezarro came up with the idea for the company-

person application. Yufeng Jing and Jamie Callan contributed significantly to the implementation of the experiments described in the paper.

References

[1] P.D. Bruza and Th.P. van der Weide. Two level hypermedia. In *Proceedings of the International Conference on Database and Expert Systems Applications*, pages 76–83. Springer-Verlag, 1990.

[2] J. P. Callan, W.B. Croft, and S.M. Harding. The INQUERY retrieval system. In *Proceedings of the Third International Conference on Database and Expert Systems Applications*, pages 78–83. Springer-Verlag, 1992.

[3] K.W. Church and W.A. Gale. Concordances for parallel text. In *Seventh Annual Conference of the University of Waterloo Centre for the New OED and Text Research*, pages 40–62, 1991.

[4] K.W. Church and P. Hanks. Word association norms, mutual information, and lexicography. In *Proceedings of the 27th Meeting of the ACL*, pages 76–83, 1989.

[5] W. B. Croft, H.R. Turtle, and D.D. Lewis. The use of phrases and structured queries in information retrieval. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 32–45, 1991.

[6] D. Harman. The DARPA tipster project. *ACM SIGIR Forum*, 26(2):26–28, 1992.

[7] W. Lehnert and B. Sundheim. A performance evaluation of text-analysis technologies. *AI Magazine*, pages 81–94, 1991.

[8] L.F. Rau. Extracting company names from text. In *Proceedings of the Sixth IEEE Conference on Artificial Intelligence Applications*, 1991.

[9] Roger H. Thompson and W. Bruce Croft. Support for browsing in an intelligent text retrieval system. *International Journal of Man-Machine Studies*, 30:639–668, 1989.