Name Searching and Information Retrieval

Paul Thompson and Christopher C. Dozier
West Group
610 Opperman Drive
Eagan, MN 55123, USA
thompson@research.westlaw.com cdozier@westpub.com

Abstract

The main application of name searching has been name matching in a database of names. This paper discusses a different application: improving information retrieval through name recognition. It investigates name recognition accuracy, and the effect on retrieval performance of indexing and searching personal names differently from non-name terms in the context of ranked retrieval. The main conclusions are: that name recognition in text can be effective; that names occur frequently enough in a variety of domains, including those of legal documents and news databases, to make recognition worthwhile; and that retrieval performance can be improved using name searching.

1 Introduction

Name searching, matching, and recognition have been active areas of research for a number of years [Hickey 1981, Carroll 1985, Rau 1991, Borgman and Siegfried 1992, Paik et al. 1993, Hayes 1994, Proceedings 1995, Pfeiffer et al. 1996], but relatively little evaluation of either the effectiveness of name searching tools or of the effect of name recognition on retrieval performance has been published. In many retrieval contexts being able to retrieve on names, whether personal, institutional, geographic, or other names, is an important capability. Some applications [Jing and Croft 1994] use name searching to extend the traditional information retrieval paradigm. To date, however, the main application of name searching has been in determining whether a name of interest in a query matches a name in a database of names [Hickey 1981, Hermansen, 1985]. Two examples of companies that develop customized name matching systems of this sort for business and government clients are Language Analysis Systems, Inc. and Search Software America.

In this paper a different application of name searching is considered: using name recognition and matching to support ranked retrieval of free text documents. Although this application uses name matching techniques much like those used in conventional relational database name searching, and name recognition, or tagging, techniques much like those of information extraction applications; text retrieval is sufficiently different from those applications, as to present different problems and issues, calling for different name searching techniques. This paper describes a series of experiments exploring the retrieval application and draws some tentative conclusions about it and how it differs from database name matching and information extraction name recognition applications.

This study reviews the accuracy of personal name recognition as shown in the Named Entity Task of the Sixth Message Understanding Conference (MUC-6) [Proceedings 1995]; investigates the frequency of personal and other names in case law and in news database queries; and finally explores the effect on retrieval performance of searching for, personal names differently from other words, through a simulation of name searching based on proximity searching. The main conclusions of this study are: 1) that name recognition in text can be done effectively; 2) that names occur frequently enough in both texts and queries of legal and news databases to make their recognition worthwhile; and 3) that name searching can lead to improved retrieval for queries with personal names.

2 Definitions, Problems, and Issues

Name searching is a term that has been used in a variety of ways. It is useful to define for purposes of this paper what is meant by *name searching* and related terminology and to describe the application areas for which name searching systems have been developed. In their comprehensive review article of personal name-matching applications Borgman and Siegfried [1992] categorize applications as being: 1) name authority control, 2) information retrieval, and 3) duplicate detection.

Name-matching in a database context is the process of comparing two character strings and determining whether or not the two strings designate the same entity; in the applications Borgman and Siegfried considered, the same person, but more generally the same institutional, geographical, or other proper-named entities as well.

This determination might be made solely on the basis of a direct comparison of the two strings, or more knowledge might be used, e.g., models of a) variant spelling or representation of names, b) keying errors, c) phonetic models, or d) record-linkage. That is, if the names to be compared are part of records containing additional fielded information, e.g., age or social security number, this information can be used as additional evidence in the name-matching process.

Name-matching assumes that two character strings have been identified which are names and the question is only whether they are instances of the same name. Typically it is also important to determine if the names refer to the same entity. Another important class of algorithms is needed for name recognition in applications where the names are not already manually identified. Name recognition is the process of identifying that a given character string is in fact a name. Such techniques can be used to extract names from text in the case of an information extraction system [Proceedings 1992, 1995], or as part of the indexing process for an information retrieval system. The same, or similar, techniques can be used at retrieval time when parsing a user's query. Commercial products, such as Carnegie Group's NameFinder and IsoQuest's NameTag are available to support these sorts of applications.

Name matching in the context of information retrieval differs from name matching in either database or natural language understanding contexts. In all three types of applications what is ultimately of interest is not that two names match, whether exactly or approximately, as character strings, but that the entities to which they refer are identical. Such reference resolution is not generally possible without some additional context. In the case of database retrieval additional context is provided by the structured nature of the data. A name typically is one field of a record corresponding to the named entity. The other fields, e.g., age, or social security number, can be used to infer that the two names being matched do refer to the same individual. In the case of natural language understanding systems there is linguistic context, as well, perhaps, as domain knowledge representation which can be used to help infer that the two names being matched refer to the same individual. Information retrieval differs from both of these types of applications, because it has neither the structure provided by a database record, nor the linguistic depth or domain knowledge representation of the natural language understanding system. Practically name matching becomes a matter of determining whether the surface forms of the two names being matched are close enough as to indicate that it is plausible that they refer to the same individual.

Name searching can be defined as the process of using a name as part of a query in order to retrieve information associated with that name in a database. Name searching, in the general case, includes both name recognition and name-matching. If names are not already identified as such in the database's text records, e.g., when they appear as part of a free text field and have not been previously tagged as being names, then name recognition is required. Similarly in parsing the query, if the name has not been identified as a name by the syntax of the query, then it will be necessary to recognize it. Once names are recognized in query and database record, then name-matching algorithms are needed to determine whether the names are the same, or that they in fact designate the same individual, e.g., two instances of the lexical entity *Judge Smith* are the same name, but may not designate the same individual.

3 The Study

This study consists of three parts. The first is a review of the literature on the accuracy of name recognition, in particular the results from the MUC-6 Named Entity Task [Proceedings 1995]. The second part of the study measures retrieval performance with name searching simulated by probabilistic searching with a proximity operator against a standard test collection with associated relevance judgments. The third part of the study analyzes the frequency of occurrence of personal and company names in legal and newspaper text collections and queries.

3.1 Name Recognition Accuracy

The Message Understanding Conferences (MUC) have evaluated the information extraction performance of the leading extraction systems for several years [Proceedings 1992, 1995]. Extracting names has always been part of the extraction task for MUC, but with MUC-6 [Proceedings 1995], a specific Named Entity sub-task was developed to focus exclusively on name extraction from news text. Participating systems were evaluated on personal, organizational, and other name recognition, as well as on related tasks, such as recognizing time and numeric expressions. The leading systems achieved very high accuracy for personal name recognition.

3.2 Evaluation of Name Recognition and Retrieval Performance

To measure the gain in retrieval performance that might be achieved using name searching, a set of 38 queries containing personal names was developed by a domain expert and run against West's FED test collection. The FED collection consists of 410,883 federal case law documents. The expert also identified the set of relevant documents from the FED collection associated with each query.

There are several ways that name searching could be implemented in a document retrieval context. One way would be to use name recognition software to tag all personal names in the document collection and also in queries. Alternatively, the collection could be tagged, but the user might be required to specify names in the query. Either way, strings designated as being names in the query would be matched against strings tagged as names in the text. Strings tagged as names in the collection might also be indexed differently than other strings. In particular they might not be stemmed, since presumably the similarity in meaning assumed to obtain among strings stemming to a common stem for general terms, would not apply to names.

A different approach to name searching would be to leave the collection unchanged, but to handle name queries differently from other queries. A combination of these two approaches would also be possible, i. e., tagging names in text and queries, as well as handling name queries differently. The strong personal name recognition results from MUC-6 [Proceedings 1995] suggest that approaches using name tagging are likely to work well. In this study, however, names were not tagged. Rather, name searching was simulated by probabilistic searching with a proximity operator for multiple word names.

The 38 queries (shown in the appendix) were run against the FED. Retrieval performance using proximity-based name searching on this test collection, as described in section 4.2, was compared against a baseline provided by the WIN retrieval algorithm. WIN is West's probabilistic retrieval engine based on the inference network model (Turtle and Croft 1991).

The baseline searches treated each term in the query as a separate concept. The relevance score for each document was computed as the sum of the logged products of each term's term frequency(tf) and inverse document frequency (idf).

The proximity searches treated non-name terms in same way the baseline searches did. However, for name terms, the proximity searches used the tf and idf of the proximally ordered name terms. The proximity searches computed relevance for names using the tf and idf of occurrences in which the first name occurred 2 or fewer word positions before the last name. In this way advantage was taken of the fact that name terms are ordered and resist interruption by non-name terms.

For example, in the query *Cases involving jailhouse lawyer Joe Woods*, the baseline search treated *Joe* and *Woods* as independent concepts. *Joe* occurred in 7,669 documents within the 410,883 document test collection and had a normalized idf of 0.31. *Woods* occurred in 18,064 documents and had an idf of 0.24. The ordered proximity search treated *Joe Woods* as a single concept in which the terms comprising the concept were proximally ordered. *Joe +2 Woods* occurred in 17 documents and had an idf of 0.78. By treating *Joe Woods* in this manner, the proximity search boosted the scores of documents containing references to the person *Joe Woods* and thereby improved search performance.

Our search engine computes the normalized idf, nidf, in the following way:

$$\text{nidf} = \frac{\ln \frac{N}{n}}{\ln N},$$

where N = collection size and n = the number of documents containing the term.

Table 1 shows the frequency counts and normalized idf for the concepts in the query *Cases involving jailhouse lawyer Joe Woods.*

Concept	Frequency	Nidf
+2(joe woods)	17	0.78
joe	7669	0.31
woods	18064	0.24
jailhouse	316	0.55
lawyer	21251	0.23
involving	136201	0.09
cases	241108	0.04

Table 1. Term frequencies and normalized inverse document frequency values for a given query

3.3 Name Recognition Case Law Collection

A manually marked up case law name recognition test collection of 724 test documents was created for evaluating name recognition and name frequency analysis. Guidelines and example marked up pages from case law text were prepared for use by the manual markers. Personal and institutional, or company, names were tagged in an SGML-like manner. Other names, acronyms, and abbreviations were also tagged including: geographic; product; facility; and (court) case names.

4 Results

The MUC-6 Named Entity Task [Proceedings 1995] results show the effectiveness of name recognition for news text, if not directly for case law text. Support for the hypothesis that name searching can lead to retrieval performance improvement was provided by simulating name searching using a proximity operator, which required that query multiple word name terms occur within two non-stopwords of each other in the text of a document The name frequency analyses show that names occur frequently enough in case law to merit special handling. In news text and queries names occur with much greater frequency (see table 4).

4.1 Name Recognition Accuracy

The leading systems on the personal name recognition portion of the MUC-6 Named Entity Task, e.g., those developed by SRA and BBN, each had recall and precision scores of 98%, or higher [Proceedings 1995]. While this performance was achieved on news text, and may not necessarily generalize to other types of text, it is a very strong result. It suggests that comparable levels of performance may be achievable for other text types, as well. NameTag [NameTag 1996], for example, was able to obtain this high accuracy using two major knowledge sources: a representation of name structure, e.g., *first name last name*; and contextual knowledge about name occurrences, e.g., that a corporate executive's name often co-occurs with a title. These knowledge sources are implemented in a) name recognition rules consisting of a pattern and an action and in b) lexical resources, e.g., part of speech information.

4.2 Effect on Retrieval Performance

For the 38 queries with personal names (see section 3.2) run against the FED collection, proximity-based name searching led to significant improvement over the baseline WIN searching. Table 2 compares results for proximity-based to the baseline. The first column of table 2 shows eleven levels of recall, while the second and third columns show the precision scores for baseline and proximity-based name searching, respectively, for the corresponding level of recall. The final row shows the eleven point averages, and the numbers in parentheses are the percentage improvement of the proximity-based approach over the baseline. This method of recall/precision evaluation is widely used in information retrieval research, and in particular has been used in the Text REtrieval Conferences (TREC) [Harman 1996]. The proximity operator required that the name terms occur within two non-stopwords of each other in the text of a document.

Recall	Precision (38 queries)	
	baseline	proximity
0	85.2	91.0 (+6.9)
10	81.9	89.5 (+9.3)
20	81.2	88.9 (+9.6)
30	80.8	88.1 (+9.0)
40	78.5	86.9 (+10.6)
50	77.1	85.2 (+10.5)
60	74.8	84.2 (+12.6)
70	72.1	83.1 (+15.3)
80	67.5	80.5 (+19.3)
90	62.8	74.4 (+18.5)
100	61.4	70.9 (+15.5)
avg	74.8	83.9 (+12.1)

Table 2. Name Recognition and Retrieval for 38 Queries Containing Personal Names

4.3 Name Frequencies in the Case Law Collection

There were 58,585 personal name word tokens in the manually marked set of 720 cases constituting the Case Law Collection. This represents 2.05% of all word tokens in the collection (not counting stopwords). Table 3 shows counts and percentages for the various types of names manually marked in this set of documents. Table 4 shows that percentage of user natural language queries containing person, company, and other names to several news databases over periods of several days in 1995.

Name Tokens	Count	Percentage
Institution	73,654	2.58
Personal	58,585	2.05
Geographic	12,800	0.45
Product	1,113	0.04
Facility	2,257	0.09
All Names	148,709	5.20
All Tokens	2,858,460	100.00

Table 3. Names and Abbreviations in 720 Document Case Law Collection

Database	Company	Person	All
Wall St. Journal	36.23	13.57	67.83
Los Angeles Times	18	38.2	83.4
Washington Post	15	17.3	38.8
Allnews	34.65	29.3	91.6

Table 4. Percentage of Queries with Names

5 Discussion

This study suggests that the name recognition accuracy of name searching software is reasonably good and it seems safe to assume that that accuracy can be improved using domain-specific heuristics and tuning. For queries containing names there was retrieval performance improvement using name searching, as simulated by proximity operators. This study further shows that the frequency of occurrence of personal, and other names in cases is sufficient to warrant their separate treatment in document retrieval.

The performance improvement obtained by proximity searching against a collection which had not had names pre-tagged suggests that better retrieval performance improvement gains may be possible using simple name matching heuristics if the query name term is known, rather than relying on pre-processed name tagging. Whether pre-tagging the collection with name recognition software could give even better retrieval performance is an open research question. The MUC-6 results imply that recognition accuracy is very high, at least for news text, but whether this would help retrieval much, given that the name to be searched is already known, i.e., specified in the query, is uncertain.

This study supports the view that name recognition and matching in the context of information retrieval is a significantly different problem from either name searching, or matching, in relational databases, or name recognition, or extraction, i.e., tagging names in free text. Most research and development has focused on these latter two applications, rather than information retrieval. The prospect of adaptation for information retrieval of the name recognition and matching techniques developed for these applications, seems promising, however. For Boolean retrieval systems one approach would be to put the burden of query name recognition on the user by requiring that the user tag a query term as being a personal, company, or other name. Then name recognition techniques, much like those of information extraction, could be used to find candidate matching names in free text and name matching techniques, much like those of database applications, could be used to determine whether names identified in query and text matched.

For systems such as WIN, Freestyle, or Target, of West Publishing, Lexis-Nexis, and DIALOG, respectively, which take natural language queries as input, the approach to take is less clear. Although it would be possible to have the user, as in the Boolean situation, tag query terms as names, this would seem to violate the underlying philosophy of natural language input search systems, i.e., that the user communicate with the search engine in ordinary natural language. If the user does not provide query name recognition, then the system must do so automatically. It might be thought that the same query recognition software used to recognize names in text could do the same in queries. This is possible, but the nature of document and query text is quite different. Much less rich syntactic content is usually present in queries, which also tend to be quite short in commercial online systems [Lu and Keefer 1995]. This greatly changes the recognition problem, especially for software which finds patterns in text as the basis of its name recognition [Krupka 1995]. Software which relied much more on an exhaustive lexicon of names and variants might do better, but could not deal with names which were not contained in its lexicon.

6 Conclusions

This paper has discussed name searching in the context of ranked information retrieval. It has been argued that while the techniques of name recognition and matching used in database searching and in information extraction can be adapted to the text retrieval problem, that the retrieval application is sufficiently different from both of the other two applications as to require very different approaches. Existing research or commercial software

can be used as parts of an overall approach to name searching, but there are major adaptations that need to be made and gaps in the architecture to be filled, such as how to recognize names effectively in user queries. Once an effective approach for name searching has been developed, there should be large benefits, especially for business areas, such as newspaper databases, where a large proportion of queries contain personal, company, product, or other names.

References

Borgman, Christine L. and Susan L Siegfried. 1992. Getty's Synoname and its cousins: A survey of applications of personal name-matching algorithms *Journal of the American Society of Information Science*, 43: 459-476.

Carroll, John M. 1985. *What's in a name? An essay in the psychology of reference* New York: W. H. Freeman.

Fuhr, Norbert. 1996. Object-oriented and database concepts for the design of networked information retrieval systems In Barker, Ken and Ozsu, M. Tamar (eds.) *Proceedings of the Fifth International Conference on Information and Knowledge Management 96*, 12-16 November, Rockville, Maryland, 1996, pages 164-172.

Harman, Donna K. (ed.) 1996. *The Fourth Text REtrieval Conference (TREC-4)* NIST Special Publication 500-236.

Hermansen, John C. 1985 Automatic name searching in large data bases of international names Ph.D. thesis Georgetown University

Hickey, Thomas B. 1981. Development of a probabilistic author search and matching technique for retrieval and creation of bibliographic records OCLC Office of Planning and Research.

Hayes, Phil 1994. NameFinder: software that finds names *in text Proceedings RIAO 94*, vol. 1, 11-13 October, New York, pages 762-774

Jing, Yufeng and W. Bruce Croft. 1994. An association thesaurus for information retrieval *Proceedings RIAO 94*, vol. 1, 11-13 October, New York, pages 146-160.

Krupka, George 1995. SRA: Description of the SRA system as used for MUC-6 *Proceedings: Sixth Message Understanding Conference (MUC-6)* 6-8 November, Columbia, MD, 1995, Morgan Kaufmann, pages 221-235.

Lu, X. Allan and Robert B. Keefer. 1995. Query expansion/reduction and its impact on retrieval effectiveness *Overview of the Third Text REtrieval Conference (TREC-3)* In Harman, Donna K., (ed.) NIST Special Publication 500-225, pages 231-239.

NameTag™ Technical Overview 1996. IsoQuest technical report.

Paik, Woojin; Elizabeth Liddy, Edmund Yu, and Mary McKenna. 1993. Categorizing and standardizing proper nouns for efficient information retrieval Proceedings of the Workshop SIGLEX (The Lexicon) held at the Association for Computational Linguistics annual conference.

Pfeiffer, Ulrich; Thomas Poersch, and Norbert Fuhr. 1996. Retrieval effectiveness of proper name search methods *Information Processing & Management*, 32:667-679.

Proceedings: Fourth Message Understanding Conference (MUC-4) 16-18 June, McLean, VA, 1992, Morgan Kaufmann.

Proceedings: Sixth Message Understanding Conference (MUC-6) 6-8 November, Columbia, MD, 1995, Morgan Kaufmann.

Rau, Lisa F. 1991. Extracting company names from text *Proceedings of the Seventh Conference on Artificial Intelligence Applications.*

Turtle, Howard R. and W. Bruce Croft. 1991. Evaluation of an inference network-based retrieval model *ACM Transactions on Information Systems*, 9:187-222.

Appendix

The 38 queries with names italicized:

1. Cases discussing *Dennis Banks* and the occupation at Wounded Knee.
2. Cases mentioning *John Ehrlichman.*
3. Cases involving the business activities of *Ferris Alexander.*
4. Cases with *Roy Rogers Creasey.*
5. Cases mentioning a biography of *Howard Hughes.*
6. Cases involving the jailhouse lawyer, *Joe Woods.*
7. Testimony by *Kenneth Boudreaux.*
8. Litigation surrounding *Theodore Bundy.*
9. Cases from judge *Diana Murphy* addressing issues relating to attorney's fees.
10. Cases involving the estate of *Elvis Presley.*
11. Cases brought by *Rudolph Lucien* as a prisoner.
12. Cases involving *Donald Trump.*
13. Cases involving *Andrea Dworkin.*
14. Cases discussing national security council staff member *Oliver North*'s dealings with the contra rebels.
15. Cases involving PTL founder *Jim Bakker.*
16. Cases involving *Larry Flynt* that deal with defamation.
17. Cases mention *Weldon Carmichael* as an expert witness.
18. Cases which refer to the expertise of Dr. *Irving Selikoff.*
19. Holocaust expert *Raul Hilberg.*
20. Cases referencing the teachings of *Irving Younger.*
21. Cases quoting the opinions of Judge *Learned Hand.*
22. References to *Alfred Hitchcock..*
23. Cases involving attorney *Bruce Cutler.*
24. References to *Laurence Tribe.*
25. Referencing to the famous *Alger Hiss* case.
26. *Marvin Mitchelson* cases.
27. Cases referring to general *William Westmoreland.*
28. Cases mentioning the author *Stephen King.*
29. References to *Oliver Wendell Holmes.*
30. *Jerry Giesler* cases.
31. Bribery cases involving *Richard LeFevour.*
32. Securities advisor *Raymond Dirks.*
33. References to King *Solomon.*
34. Abscam cases involving Congressman *Richard Kelly.*
35. References to *Julius Rosenberg.*
36. Lawsuits involving *Vanessa Redgrave.*
37. References to the trial of *Aaron Burr.*
38. Cases mentioning the trial of Sir *Walter Raleigh.*