Document Indexing using Named Entities

Rada Mihalcea and Dan I. Moldovan
Southern Methodist University
Department of Computer Science and Engineering
Dallas, Texas, 75275-0122
{rada, moldovan}@engr.smu.edu

Abstract: Current text indexing and retrieval techniques have their roots in the field of Information Retrieval where the task is to extract documents that best match a query. With an ever increasing number of documents available due to the easy access through the Internet, the challenge is to provide users with *concise* and *relevant* information. We are proposing here a novel, yet simple approach, which indexes the named entities in the documents, such as to improve the relevance of documents retrieved. Experiments performed in finding information related to a set of 75 input questions, from a large collection of 125,000 documents, show that this new technique reduces the number of retrieved documents by a factor of 2, while still retrieving the relevant documents.

Keywords: Information Retrieval, Semantic Indexing, Question Answering

1. Introduction

With the tremendous amount of information available today, it becomes crucial to have fast and accurate retrieval systems. Current Information Retrieval (IR) systems that operate on a fixed set of documents, as well as the Internet search engines like [Altavista 2000], [Infoseek 2000], Metacrawler [Selberg and Etzioni 1997], [Google 2000], extract documents by finding keywords in documents.

The main problem of the traditional keyword-based approach to IR is that usually too many results, or wrong results, are returned to be useful. To overcome these impediments, a solution is to include more information when the documents are indexed. Recently, there is a growing interest to introduce semantics into the Internet search. Hellman [Hellman 1999] presents the benefits of a semantic Web and the industry interest in this. Still, it is not made very clear what kind of semantic tags could lead to a better search.

We describe in this paper a methodology for increasing the precision of current IR systems by indexing the Named Entities (NE) found in the documents. To our knowledge, this approach is novel, yet simple. The idea presented here relates basically to the incorporation of semantic tags into the input set of documents, and complements the work we are currently doing in the field of semantic indexing [Mihalcea and Moldovan 2000].

The inputs to IR systems usually consist of a question/query and a set of documents from which the information has to be retrieved.

Consider for example the question: ``*Who was the first president of the United States?*". Using the currently existing search techniques, one would extract some keywords, as for example "*first*", "president", "United States" and search the collection of documents with a query which combines these keywords. The problem arising here is that there is no way to specify that the **expected answer** to this question is a *person*, and thus several wrong results can be obtained in return to such a query. It would be useful to have the possibility to add to the query a new term, like a *PER* tag which would specify that we are also looking for a person name.

The technique we are proposing here is basically to enrich the texts with NE tags prior to document indexing. This enables the formation of a query that includes **answer type** in addition to the keywords specified in the input question. Several experiments have been performed using the AltaVista Discovery search tool. A set of 125,000 documents has been enriched with NE tags and indexed using AltaVista.

Next, a set of 75 questions was run against this index. The results obtained show that the number of documents retrieved was reduced by a factor of 2, while still retrieving relevant documents.

For the purpose of identifying NE, we used the named entity recognizer implemented in Minipar [Lin 1994], which addresses the following NE categories: *person, location, organization, date, number, percent, money, product* and *special collocations*.

RELATED WORK

The Natural Language Processing (NLP) tracks taking place within the TREC conferences proved that NLP techniques could actually improve IR systems. One of the main techniques used in this direction is the identification of semantic and lexical relations and properties associated with the terms found in documents. Attempts have been done in indexing concepts [Woods 1997], phrases [Zhai et. al 1996], or word senses [Schutze and Pedersen 1995], [Gonzalo et. al 1998], [Mihalcea and Moldovan 2000c].

[Pustejovsky et. al 1997] has shown that semantic tagging is one of the central components needed towards the goal of having a more efficient IR. The problem is how to perform this semantic tagging. A solution has been offered by [Hellman 1999]: referring to the search on the Web, he shows that a semantic Web would overcome the problem of having too many results, as usually obtained from the current search engines; he is mentioning the usage of languages such as XML or XHTML to perform the semantic tagging. The problem with this approach is that it will take long time to standardize these languages and actually have the people creating web pages to use them. Instead of having the semantic tagging accomplished by the authors of a text, this task can be performed at the IR engine level. Even if this will result in a lower precision, it is more feasible to have a semantic tagger implemented in a central engine, instead of requiring thousands of people to semantically tag the text they are writing.

In this context, the idea of enriching texts with named entity tags comes naturally. Indexing named entities complements the work performed so far by researchers in indexing word senses in order to increase the IR effectiveness .

2. Named Entities

Named Entities are words or word sequences which usually cannot be found in common dictionaries, and yet encapsulate important information that can be useful for the semantic interpretation of texts. The problem of recognizing NE in a text has constituted, for years, one of the main tasks addressed in the Message Understanding Conferences (MUC).

As we have outlined earlier, the main gain obtained by enriching the texts with NE tags prior to the indexing stage, is that this will enable the retrieval of **answer types**.

The approaches considered so far by researchers for NE recognition are mainly based on machine learning algorithms. The following systems learn to identify and automatically classify named entities from a training corpus: Nymble [Bikel et. al 1997], MENE [Borthwick et. al 1998], LTG [Mikheev et. al 1998], or the system described in [Collins and Singer 1999]. Most of these systems took into consideration the seven classes of named entities, as defined in the MUC NE task: *person, location, organization, date, monetary values, numbers, and percentages*. We have also tagged other special collocations, which prove to be useful for the IR task, as for example *distance-length*, *time-length*, *weight*. These are the NE classes on which we are currently focusing, but further extensions will consider other NE tags as well.

To recognize NE in free text, we have used the NE tagger implemented in Minipar [Lin 1994]. The reasons for choosing this tagger are: (1) the fact that is publicly available, and (2) its high precision: we have NE tagged 10 texts from the *LA Times* collection and manually checked them, obtaining a precision of about 89%.

Question type	Answer type	Example
who	_PER	Who was the first American in space?
where	_LOC	Where is Taj Mahal?
when	_DATE	When did French revolutionaries storm the Bastille?
how how-many how-tall how-far	 _NUM _SPEC _SPEC	 How many Grand Slam titles did Bjorn Borg win? How tall is the Mattherton? How far is Yaroslavl from Moscow?
what	_PER _LOC	What famous communist leader died in Mexico City? What is the tallest building in Japan?

Table 1: Examples of answer types for some question classes

There are 9 types of tags used in Minipar: _PER (person); _ORG (organization); _LOC (location); _DATE (date); _NUM (number); _MONEY (money); _PCT (percentage); _PROD(product); _SPEC (special collocation).

3. Document processing

Document processing consists of four main phases:

1. Tokenization
2. Part of speech tagging; this task is performed using Brill's tagger [Brill 1992]
3. NE tagging, as described above.
4. Word stemming, based on the WordNet dictionary. Knowing the part of speech of each word enables us to use the WordNet stemming functions.

EXAMPLE OF TEXT ENRICHED WITH NE TAGS

Figure 1 shows an example of text enriched with NE tags. Also, the words are stemmed based on the WordNet dictionary. Note that the tag is inserted in the text, before the named entity.

Original text

It turned out to be the other way around. O'Toole opened his Pan Am flight bag and Salo found himself staring at a bundle of bonds - $50,000 worth. "I want Planned Parenthood to have this money" O'Toole said. And so began a strange, 11-year acquaintanceship between Salo and the mysterious visitor, who died recently at the age of 91 and left the Planned Parenthood affiliate his life's fortune of nearly $200,000.

Text with stemmed words and NE tags

It turn out to be the other way around. _PER O'Toole open his _ORG Pan Am bag and _PER Salo find himself stare at a bundle of bond - _MONEY $50,000 worth. " I want Planned Parenthood to have this money " - _PER O'Toole say. And so begin a strange _SPEC 11 years acquaintanceship between _PER Salo and the mysterious visitor, who die recently at the age of _NUM 91 and leave the Planned Parenthood affiliate his life 's fortune of nearly _MONEY $ 200,000.

Figure 1. An example of text with stemmed words, enriched with NE tags.

4. Questions processing and answer types

Every question entering the system has to be processed such as to fit the format accepted by the IR system. As we are using here the AltaVista Discovery tool, the questions have to be transformed into queries formed with keywords connected with the boolean operators AND, OR, NEAR. Thus, we have to determine the keywords that are to be considered for the search.

We have to find also the **answer type** of each question. An extensive analysis of questions and their possible answer types was done in the LASSO Question Answering system; Table 1 shows an extract of this analysis, including examples of answer types. For a more detailed analysis, see [Moldovan et. al 1999].

QUESTION PROCESSING
First, we part of speech tag the question using Brill's tagger. Then, we determine its **answer type** and determine the keywords to be used for the search using the following rules:
- Use all the nouns in the question;
- Use all the adjectives in superlative form;
- Use all the numbers (cardinals);
- If more than 200 documents are returned, use the adjectives modifying the first noun phrase.

The purpose of the experiments reported in this paper is to show that indexing the documents using named entities can lead to a better search, namely the number of documents returned is significantly reduced, while still finding relevant information. The task of actually finding answers in text is much more difficult and it needs more complex heuristics for keyword selection, as those presented in [Moldovan et. al 1999].

Using the keywords selected by the rules above, and the expected answer, a query is formed in the following way: the **answer type** is connected with the NEAR operator with the head noun of the first noun phrase in the question. All the other keywords are connected with the AND operator. The rationale for this methodology of connecting the answer type to the query is that we are expecting the answer to be closed to one of the central concepts of the question. The experiments performed showed that choosing the head noun of the first noun phrase as a central concept enables us to retrieve relevant documents.

For each question, we are forming two types of queries: one denoted by Q_{NE} which includes the **answer type**, and one denoted by Q_{SS}, which is the classical query to be posed to a search engine and connects the keywords with the AND operator.

Consider the following example:
"Who may be best known for breaking the color line in baseball?"

The queries formed are:
Q_{NE}: *_PER NEAR line AND color AND baseball*
respectively
Q_{SS}: *line AND color AND baseball*.

This simple technique that indexes named entities, and incorporates the expected answer into the input query proves to lead to better searches; in the next section we are describing the experiments performed and present the results.

5. The experiments

To determine the effect of NE tagging prior to document indexing, and how this affects the precision of IR systems, we had performed a set of experiments using the AltaVista Discovery [Altavista 2000] tool.

The set of documents considered during these experiments are a subset of the data provided by NIST for the TREC conferences [Trec 1999]. We used the *L.A.Times* collection, which consists of nearly 500 Mbytes

of data, i.e. 125,000 documents. Each of the documents in this set has been processed, i.e. each document was tokenized, part of speech tagged, the words were stemmed using WordNet dictionary and NE tags have been added using the parser mentioned in a previous section.

It is important to notice that the SGML tags or the homogeneous characteristics of the document collection have not been used in any way. We have chosen this data for reasons of availability. Thus, the technique described in this paper is generally applicable, and can be used in the same way for Internet searches.

After processing the documents, they have been indexed using AltaVista Discovery. Speaking about time complexity, nearly 13 hours were needed to process the documents, while the indexing process took about 48 hours. This results in an average time of 1.56 minutes for processing, and 5.76 minutes for indexing for each Mbytes of data. An interesting observation to be made here is the fact that the time for documents pre-processing represents about a quarter of the time employed for indexing; for this experimental setup, preprocessing documents did not represent a significant overhead respect to the normal IR process.

Next, we had to choose a set of questions having an answer in the test collection we used during the experiments. For this purpose, we started with the 200 questions provided during the Question Answering track of the TREC-99 conference. The SMU team had successfully participated at TREC 1999 with the LASSO system [Moldovan et. al 1999].
We used the set of correct answers determined by the TREC assessors for the 200 questions, such as to determine a set of questions that have an answer in the *L.A.Times* collection. This resulted in a set of 75 questions for which we knew that a relevant document could be found in the *L.A.Times* collection.

No.	Question	Original documents		Processed documents	
		(a)	(b)	(a)	(b)
1	When did Nixon visit China?	110	1	52	1
2	Who may be best known for breaking the color line in baseball?	85	1	35	1
3	What is the legal blood alcohol limit for the state of California?	21	1	15	1
4	When did the Jurassic Period end?	1	1	1	1
5	What is the name of the highest mountain in Africa?	43	1	1	1
6	How far is Yaroslavl from Moscow?	4	1	1	1
7	When did the original Howdy Doody show go off the air?	3	0	1	0
8	How long did the Charles Manson trial last?	15	1	4	1
9	What was the monetary value of the Nobel Peace Prize in 1989?	26	1	5	1
10	What debts did Qintex group leave?	12	1	5	1
	Average question (for the 75 questions set)	38.3	0.9	20.6	0.85

Table 2: Sample of the results. 10 questions together with the results obtained by searches made using the original documents, respectively the documents with stemmed words and enriched with NE tags. (a) columns show the number of documents retrieved, (b) columns indicate if a relevant document was found (0/1).

The questions were processed and we automatically determined their answer types. Queries have been formed using the keywords determined from the questions, as well as the **answer type**. Table 2 shows 10 of the questions together with the results obtained by searches made using the original documents, respectively the documents with stemmed words and enriched with NE tags. We are showing in the (a) columns the number of documents returned by the search while (b) columns indicate if a relevant document was found in the set of retrieved documents (binary 0/1, 0=no relevant documents, 1= one or more relevant documents were found).

For the total set of 75 questions considered during our experiments, the average number of documents retrieved using the original documents and the Q_{SS} format of query was 38.3, and for 68 questions a relevant document was found. On the other hand, using the processed documents (i.e. NE tags are added and the words are stemmed based on WordNet) and the Q_{NE} format of query (which incorporates also the answer type), 20.6 documents were retrieved and a relevant document was found for 64 questions.

This means a reduction by a factor of 2 in the number of documents retrieved, while still finding relevant documents for 95% of the questions with respect to the number of questions answered using the original documents and the Q_{SS} query format.

The measurements used in classic IR systems, i.e. *precision* and *recall*, are not appropriate for Internet or for very large collection of texts, as it is impossible to know a priori *all* the documents that are relevant to an input question. A variation of the *precision* measurement, which addresses better the Internet environment, can be defined as:

$$precision_I = \begin{cases} 0, & \text{if no relevant documents are found} \\ \frac{1}{N}, & \text{otherwise} \end{cases}$$

where N is the number of documents retrieved (or the number of documents checked by a user).

The average $precision_I$ for the 75 questions, when the original documents and the Q_{SS} query format are used is 0.023. Using the documents enriched with NE tags and the Q_{NE} query format leads to a $precision_I$ of 0.04, i.e. an increase of 73%.

6. Conclusion and future work

We have presented in this paper a technique that indexes named entities with the purpose of increasing the quality of the search. Experiments have been performed in indexing and retrieving information from a collection of 125,000 documents. The results obtained show that this new technique can reduce the number of documents retrieved by a factor of 2, while still finding the relevant information.

There are some limitations of the method described here. For example, searches for *"why"* questions cannot be improved using this technique, and the solution is to use the keyword-based approach.

Nevertheless, adding NE tags to the texts prior to document indexing proves to be a good solution towards the goal of increasing the quality of the searches on Internet or on very large collections of texts.

As future work, we plan to study the effect of adding other types of semantic information to the texts prior to the indexing phase and measure the impact over the quality of the retrieval.

References

[Altavista 2000] Altavista, 2000. Digital Equipment Corporation, "http://www.altavista.com".

[Bikel et al.1997] D. Bikel, S. Miller, R. Schwartz and R. Weischedel. 1997. NYMBLE:, a high-performance learning text finder. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing* , pages 109-116, Providence, RI.

[Borthwick et al.1998] A. Borthwick, J. Sterling, E. Agichtein, and R. Grishman. 1998. NYU: Description of the MENE named entity system as used in MUC-7. In *Proceedings of the Seventh Message Understanding Conference*

[Brill 1992] E. Brill. 1992. A simple rule-based part of speech tagger. In *Proceedings of the 3rd Conference on Applied Natural Language Processing*, Trento, Italy.

[Collins and Singer 1999] M. Collins and Y. Singer. 1999. Unsupervised models for named entity classification. In *Proceedings of EMNLP/VLC.*

[Gonzalo et al.1998] J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarran. 1998. Indexing with WordNet synsets can improve text retrieval. In *Proceedings of COLING-ACL '98 Workshop on Usage of WordNet in Natural Language Processing Systems*, Montreal, Canada, August.

[Google 2000] Google. 2000. "http://www.google.com".

[Hellman 1999] R. Hellman. 1999. A semantic approach adds meaning to the Web. In *Computer*, pages 13--16.

[Infoseek 2000] Infoseek. 2000. "http://www.infoseek.com".

[Lin 1994] D. Lin. 1994. Principar - an efficient, broad-coverage, principle-based parser. In *Proceedings of COLING-94*, pages 42-48, Kyoto, Japan.

[Mihalcea and Moldovan 2000c] R. Mihalcea and D.I. Moldovan. 2000. Semantic indexing using WordNet senses. In *Proceedings of ACL Workshop on IR & NLP*, Honk Kong, October.

[Mikheev et al. 1998] A. Mikheev, C. Grover, and M. Moens. 1998. Description of the LTG system used for MUC-7. In *Proceedings of the Seventh Message Understanding Conference*.

[Moldovan et al. 1999] D. Moldovan, S. Harabagiu, M. Pasca, R. Mihalcea, R. Goodrum, R. Girju, and V.~Rus. 1999. LASSO: A tool for surfing the answer net. In *Proceedings of the Text Retrieval Conference (TREC-8),* November.

[Pustejovsky et al. 1997] J. Pustejovsky, B. Boguraev, M. Verhagen, P. Buitelaar, and M. Johnston. 1997. Semantic indexing and typed hyperlinking. In *Proceedings of the American Association for Artificial Intelligence Conference, Spring Symposium, "NLP for WWW"*, pages 120-128, Stanford, CA.

[Schutze and Pedersen 1995] H. Schutze and J. Pedersen. 1995. Information retrieval based on word senses. *In Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 161-175.

[Selberg and Etzioni 1997] E. Selberg and O. Etzioni. 1997. The MetaCrawler architecture for resource aggregation on the Web. In *IEEE Expert*.

[Trec 1999] TREC-8. 1999. Text retrieval conference. http://trec.nist.gov.

[Woods 1997] W.A. Woods. 1997. Conceptual indexing: A better way to organize knowledge. Technical Report SMLI TR-97-61, Sun Microsystems Laboratories, April. (available online at: http://www.sun.com/research/techrep/1997/abstract-61.html).

[Zhai et al.1996] C. Zhai, X. Tong, N. Milic-Frayiling, and D.A. Evans. 1996. Evaluation of syntactic phrase indexing - CLARIT NLP track report. In *Proceedings of the 5th Text Retrieval Conference TREC-5*, pages 347-359, Gaithersburg, Maryland, November.