Finding Answers in Large Collections of Texts: Paragraph Indexing + Abductive Inference

Sanda M. Harabagiu
Southern Methodist University
Dallas, TX 75275-0122
sanda@seas.smu.edu

Steven J. Maiorano
AAT
Washington, D.C. 20505
stevejm@ucia.gov

Abstract

This paper describes a methodology of answering questions by using information retrieved from very large collections of texts. We argue that combinations of information retrieval and extractions techniques cannot be used, due to the open-domain nature of the task. We propose a solution based on indexing techniques that identify paragraphs from texts where the answers can be found. The validity of the answers is obtained through a lightweight process of abduction.

Background

The process of answering a question starts with finding information from which the answer can be entailed. When the information is represented in axiomatic knowledge bases, logical resolution, refutation and theorem proving are the disciplines of AI that can help with the derivation of answers to questions posed in logical representations. In contrast, when the information is not represented in a formal language, but in natural language, answering questions has to deal with natural language ambiguities as well. Information retrieval (IR) and information extraction (IE) are the subfields of natural language processing (NLP) that approximate the complex problem of answering questions from large collections of texts.

Given a query, current IR systems allow us to locate documents that might contain the pertinent information, but most of them leave it to the user to extract the useful information from a ranked list. The extraction of the information of interest is the object of IE systems, provided that the information is to be mapped into a predefined, target representation, known as *template*[1].

Research in the area of Information Retrieval (IR) has been encouraged by the Text Retrieval Conference (TREC),[2]. The Message Understanding Conferences (MUCs) and the TIPSTER programs gave great impetus to research in IE. The systems that participated in the MUCs have been quite successful at extracting information from newswire messages and filling templates with the information pertaining to the events of interest. Typically, the templates model queries regarding *who* did *what* to *whom*, *when* and *where*, and eventually *why*.

However, combinations of IR and IE systems are impractical solutions for delivering answers for open-domain questions, due to the dependency of IE systems on domain knowledge. Systems that can do reliable question answering without domain restrictions have not been developed yet. Given the vast amount of information available in textual form, for example on the Web, it would be very useful if answers to users' questions could be found automatically from the underlying text.

To take a step closer to *information retrieval* rather than *document retrieval*, this year TREC has initiated an experimental track: the *Question/Answering* (Q/A) track, whose aim is to foster research in the area of textual Q/A, especially when pursued in a domain independent manner. Participation from many groups with different strengths in IR and IE will highlight whether:

1. Surface-text-based methods, like text-snippet extraction, can be effectively used for question answering.
2. Linguistic techniques can be used to enhance shallow methods for effective question answering.

Evaluation of fully automatic Q/A systems will measure the effectiveness of the technologies employed. Motivated by our initial experiments, we argue that the detection of potential answers is made possible by statistical and surface techniques that are not currently

[1]The format of templates is often imposed by interfaces to databases

[2]TREC is sponsored by the National Institute of Standards and Technology (NIST) and the Defense Advanced Research Projects Agency (DARPA)

employed either in IR or in IE systems. The assessment of the validity of an answer is provided by a weighted abductive inference, operating on an ad-hoc knowledge representation. This method contrasts with traditional ontology-based systems which employ symbolic reasoners, normally giving "all or nothing" types of answers.

Classification of questions

The TREC Q/A Track specifies two restrictions for the questions. First, questions should have an exact answer that occurs in some document from the underlying text collections. Table 1 lists the number of documents and their sources, whereas Figure 1 illustrates the format of documents through one example from the *LA Times* collection.

Source	Nr. Documents	Percentage
LA Times	22,715	0.34%
Foreign Broadcast Information Service	6,265,666	95.53%
Federal Register	60,089	0.91%
Financial Times	210,158	3.22%
TOTAL	6,558,628	100%

Table 1: Collections of documents

```
<DOC>
<DOCNO>LA042389-0067</DOCNO>
<DOCID> 47702 </DOCID>
<DAT>
<p>
April 23, 1989, Sunday, Home Edition
</p>
  ....
<p>
Two strangers, one with a broken leg arrived at the Maryland
farmhouse of Dr. Samuel Mudd early on April 15, 1865. Dr. Mudd
set the injured man's leg and afterward invited his patient and
guest to rest in a spare bedroom.
</p>
<p>
The patient was John Wilkes Booth, who hours earlier had fatally
shot President Abraham Lincoln at Ford's Theater in Washington
D.C., about 30 miles away.
</p>
  ....
</DOC>
```

Figure 1: SGML format of document

The second restriction applies to the length of the answer. There are two answer categories. The first one limits the length to 50 contiguous bytes of text whereas the second category comprises answers that either (a) represent a sentence or (b) are under 250 bytes of text. Table 2 lists a set of questions and their answers from the first category, whereas Table 3 illustrates two possible answers from the second category[3].

Question 1	Who shot President Abraham Lincoln?
Answer 1	*John Wilkes Booth*
Source	LA042389-0067
Question 2	How many lifes were lost in the Pan Am crash in in Lockerbie, Scotland?
Answer 2	*270*
Source	LA012589-0105, LA042689-0020
Question 3	How long does it take to travel from London to Paris through the Chunnel?
Answer 3	*three hours 45 minutes*
Source	FT944-8324
Question 4	Which Atlantic hurricane had the highest recorded wind speed?
Answer 4	*Gilbert (with wind speeds exceeding 200 m.p.h.)*
Source	LA120389-0130, LA092089-0027
Question 5	Which country has the largest part of the Amazon rain forest?
Answer 5	*Brazil (which has 60%)*
Source	LA032590-0089

Table 2: Short answers (first category).

Question	Who was Lincoln's Secretary of State?
Answer-1 (sentence)	*Booth schemed to kill Lincoln while his compatriots would murder Vice President Andrew Johnson and Secretary of State William Seward.*
Answer-2 (250 bytes)	*to kill Lincoln while his compatriots would murder Vice President Andrew Johnson and Secretary of State William Seward. Only Booth was successful,*

Table 3: Answers from the second category.

The simplicity of the task formulated for the Q/A TREC track is quite intentional, because it promotes large scale independent domain Q/A over systems that handle more sophisticated, domain dependent questions. Due to the length restrictions, questions that will be processed in the first experiments of the TREC Q/A track fall in Class 1 out of the five classes of questions[4], listed in Table 4.

Since the answers to questions from Class 2 need to comprise multiple sentences, they do not meet the length constraints set by the TREC Q/A guidelines. Moreover, answers are expected to indicate the document where they occur, thus answers from the Class 3 do not qualify, as they are derived from information distributed across several texts.

[3]These questions and answers were provided by the NIST organizers of the Q/A track of the TREC Conference.

[4]Classification proposed on the Q/A discussion mailing list.

Length and document uniqueness requirements do not apply to answers for questions from Class 4, since a coherent discourse spans several sentences, possible from several documents. Finally, questions from Class 5 impose answers produced by reasoning and implications, thus they would rarely have the answer occurring in some document.

Class 1	The answer is: *single datum or list of items found in verbatim* Characteristics: *who, when, where, how (old, much, large)* Examples: *Who is the president of France?* *Who were the American Presidents of the last Century?*
Class 2	The answer is: *multi-sentence, usually in a paragraph or two* Characteristics: *Extract facts not stated in a single datum or a list* Examples: *What happened after the Titanic sank?*
Class 3	The answer is: *across several texts* Characteristics: *Comparative/ contrastive response* Examples: *What are the various opinions about legal drug use?*
Class 4	The answer is: *an analysis of the retrieved information* Characteristics: *The answer is synthesized coherently from several retrieved fragments* Examples: *Create an argument to convince people to stop immigration* *Should the Fed raise interest rates at their next meeting in May?*
Class 5	The answer is: *the result of complex reasoning* Characteristics: *Need of extensive world knowledge and commonsense reasoning capabilities in conjunction with domain knowledge* Examples: *What should be the defense strategy in the O.J. Simpson murder case?*

Table 4: Classes of questions

Questions from the Class 5 are addressed by research techniques that rely on ontological resources, coupled with derivational devices. An example of such endeavors is provided by the evaluations taking place under the High Performance Knowledge Bases (HPKB) initiative[5] One of these evaluations was the *Crisis Management Challenge Problem* (CMCP), designed to address questions about international crises, such as " What will the US response be if Iran closes the Strait of Hormuz?" (Cohen et al.1998).

The scope of the CMCP[6] competition focused on the reasoning aspect of Q/A. This was made possible by the availability of: (1) commonsense ontologies[7]; and (2) by parameterizing the questions with a *question grammar*. Figure 2 illustrates some of the parameterized questions (PQ) and corresponding sample questions (SQ) used in CMCP.

PQ 5-6, 87, 89 [What {amount, fraction} of <InternationalAgent1>'s <EconomicMeasure1> comes from <EconomicMeasure2> (in <UnitOfMeasure>)]?

SQ5: How much oil does Japan purchase from the Persian Gulf states?
SQ87, SQ97: How much did Iran spend on military hardware inports in 1994 (in 1990 US$) ?

PQ22 [What effect might an <InternationalActionType> have on the {price, supply, demand} of/for <ProductType> on {the international, <Country>'s domestic} market?

<InternationaActionType>=
{terrorist attack on <InternationalAgent>'s <ProductType> facilities, {boycott, embargo} of <ProductType> by <InternationalAgent1> against <InternationalAgent2>, blockade of <Country>, <HPKB:BodyOfWater>} by <InternationalAgent>

SQ22: What effect on the price of oil on the international market are likely to result from the terrorist attacks on Saudi oil facilities on Days 22, 23 and 25?

Figure 2: Parameterized questions (PQ) and sample questions (SQ) for the CMCP tests

The knowledge sources employed in the CMCP competition are listed in Figure 3. Each of the knowledge sources are represented by Web pages, in which fragments of texts are manually tagged as relevant for the CMCP competition. Figure 3 illustrates one such annotation. The tagged fragments do not specify the parameterized questions to which they are relevant.

The open-domain characteristic of questions from the TREC Q/A track makes it impossible to generate a grammar of questions. Instead, we conjecture that all questions from Class 1 are either (1) requests for fillers of some thematic role of a structure defined by the question or (2) requests for subordinates of a

[5]HPKB is sponsored by DARPA (cf. *http://www.teknowledge.com*).

[6]CMCP was designed by a team from IET, Inc and PSR Corp., led by Robert Schrag.

[7]Ontologies were based on Cyk, the commonsense knowledge base developed by Cycorp for the past decade

concept specified by the question. The ultimate goal of the TREC Q/A evaluations is to assess whether the current technology can produce automatically annotations that represent answers from the TREC corpora. Unlike the annotations employed in the CPCM task, the TREC Q/A answers are very short fragments of text, unbound to any prespecified domain.

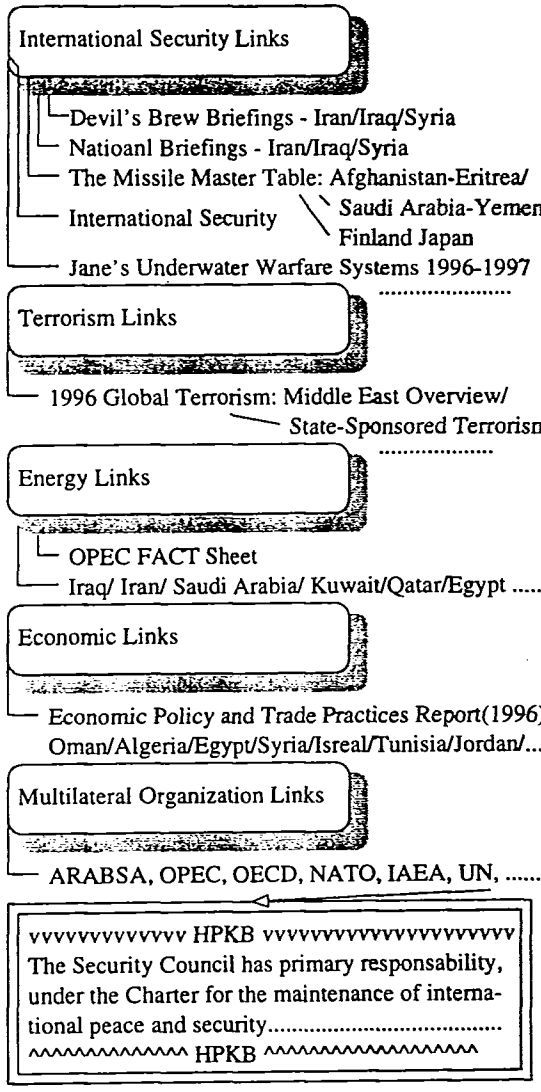

Figure 3: Knowledge Sources for CMCP

We distinguish three categories of questions from Class 1:

1. *Class 1.a* comprising questions about the subjects, objects, manner, time or location of a certain action or event. Questions 1 and 3 from Table 2 are elements of this class.
2. *Class 1.b* made of queries about the properties or attributes of a certain person, place, object or phenomenon. This class contains also questions that ask for the identification of objects with given properties. Examples are Questions 4 and 5 from Table 2.
3. *Class 1.c* consisting of questions of taxonomic nature. The answer should enumerate subordinate objects or entities. An example is "What are the forms of international crimes committed in Latin America?". Questions related to the cardinality of a set are also classified in this category. Question 2 from Table 2 is an example.

To be able to identify relevant answers to questions from any of the above categories, we have two choices: (a) to use IR techniques and extract the answers from the returned documents or (b) to parse all texts and derive conceptual indexes. The first method relies on indexing techniques developed for document retrieval. Semantic ambiguity is the obstacle that makes the second choice impractical at the moment, although we are aware of efforts for building open-domain semantic indexes (Woods 1997). Therefore, we have decided to index all texts such that we can return *paragraphs* where the answers might lie instead of documents that contain relevant information for the questions. This reduces significantly the inspection time.

Indexing for Q/A

There is no question that the performance of a Q/A system that operates on textual documents depends in a large measure on its indexing capabilities. To us, and many other users of IR systems and search engines over the Internet, it has become apparent that indexing for Q/A should be handled differently than in classical IR systems, because we should not retrieve relevant documents, but indexes to paragraphs from documents where the answer may be found.

There are a number of options for searching and indexing large and distributed collections of documents, each with its strengths and weaknesses. Searching solutions fall into two broad categories: single versus distributed search engines. A single engine builds a full index of the entire collection of documents, whereas distributed search indexes[8] combine results from several search engines.

The most popular search techniques are the vector-space search, widely used in IR (Buckley et al.1998) (Salton 1975) and boolean search, employed by several Internet search engines (e.g. Altavista). Indexing can be either based on word frequencies (weight indexing) or probabilistic. Another option for indexing is provided by *hyperlinks* that can be generated automatically when computing semantic similarity (cf. (Green 1997)). Hyperlink indexing provides with a representa-

[8] Also known as meta-search engines

tion of knowledge sources similar to the one employed in CMCP.

Vector-space indexing

Under the vector-space model, documents and queries are conceptually represented as *vectors* (cf. (Salton 1989)). If the vocabulary contains n words, a document D is represented as a normalized n-dimensional vector $D=<w_1,w_2,...,w_n>$, where w_i is the weight assigned to the word (term) t_i. If t_i is not present in D, then w_i is 0. The weight w_i indicates how statistically significant word t_i is.

One common way of obtaining document vector D is by first computing the un-normalized vector $D'=<w'_1,w'_2,...,w'_n>$, where each w'_i is the product of the *word frequency factor* (*tf*) with an *inverse frequency factor* (*idf*). This indexing scheme assigns the largest weight to those terms which occur with high frequency in individual documents, but are at the same time relatively rare in the collection as a whole. The *tf* factor is equal (or proportional) to the frequency of the i^{th} word within document d. The *idf* factor represents the content discriminating power of the i^{th} word, and is typically computed by $log\frac{N}{d_i}$, where N is the total number of documents in the collection and d_i is the number of documents containing the i^{th} word. Once D' is computed, the normalized vector D is typically obtained by dividing each element w'_i with $\sqrt{\sum_{i=1}^{n}(w'_i)^2}$.

Questions (queries) in the vector-space model are also represented as normalized vectors over the word space, $Q=<q_1,q_2,...,q_m>$, where $q_i=r_i*idf$, in which r_i is the number of times term t_i appears in the query.

The retrieval of a relevant document is based on the similarity (*Sim*) computation obtained as an inner product of the query and document vectors:

$$Sim(D_i,Q_j)=\sum_{k=1}^{n}(w_{ik}*q_{jk})$$

Another similarity measure used in the vector space model is the cosine similarity:

$$Cos(D_i,Q_j)=\frac{\sum_{k=1}^{n}(w_{ik}*q_{jk})}{\sqrt{\sum_{k=1}^{n}w_{ik}^2\sum_{k=1}^{n}q_{jk}^2}}$$

To be able to retrieve paragraphs relevant to a question instead of documents, we have modified the weighting scheme of the vector-space. Instead of the indexing based on *tf*idf* weights, we use a weighting method that promotes the proximity of query terms in the document. Weights are computed by the product *tr*idr*, where *tr* measures the term relevance. The term relevance is calculated using a formula employed by the *Inquirus* meta-search engine (Lawrence and Giles), whereas *idr* represents the content discriminating power of all the words from the same paragraph. The formulae are:

$$tr=c_1N_p+(c_1-\frac{\sum_{i=1}^{N_p-1}\sum_{j=i+1}^{N_p}min(d(i,j),c_2)}{c_2\sum_{k=1}^{N_p-1}(N_p-k)})+\frac{N_t}{c_3}$$

where N_p is the number of query terms that are present in the document (each term is counted only once), N_t is the total number of query terms in the document (each term is counted as many times as it appears), $d(i,j)$ is the minimum distance between the i-th and the j-th query terms which are present in the document. Constant c_1 controls the magnitude of *tr*, whereas c_2 specifies the maximum distance between query terms and c_3 specifies the importance of term frequency[9].

The *inverse relevance factor* can be computed as $idr=log(N/\prod_{k=1}^{n_p}d_k)$, where n_p is the number of terms in the same paragraph and d_k is the number of documents containing the k^{th} term.

Boolean indexing

The *Boolean* model is more primitive than the vector-space model, but it is important because many sources still use it for answering queries. Moreover, our initial experiments under the TREC Q/A track proved that the boolean index, built by *Altavista* was superior to the vector-model index, provided by the *PRIZE* search engine publicly available from NIST. We have post-processed the retrieved documents with a simple operator, called *Paragraph* (Moldovan and Mihalcea 1998), which identifies document paragraphs that contain the majority of the query terms or their WordNet (Miller 1995) synonyms. We were able to find the answer to a question in at least one of the paragraphs in at least 43% of the cases. We conclude thus that boolean indexing is a technique that should be considered for textual Q/A.

In Boolean indexing, documents are represented as words with position information. Queries are expressions composed of words and connectives such as "and", "or", "not" and proximity operators such as "within k words of". The answer to the query is the set of all the documents that satisfy the Boolean expression. Because the Boolean model does not rely on document-query similarities (i.e. a document does or does not satisfy a query), the indexing structure is made of a collection of document vectors. We can refine boolean indexing, by building clusters of documents that present high similarities, as proposed in (Li and Danzig 1997). Well known similarities, such as *Dice's coefficient*, *Jaccard's coefficient*, *Cosine similarity* and *Overlap similarity* (van Rijsergen 1979) can be used to provide for a cluster-based retrieval. Radecki

[9]In Inquirus c_1=100, c_2=5000 and $c_3=10c_1$

has employed several measures to rank similarities between boolean queries and clusters of documents. New similarities based on his formula were proposed in (Li and Danzig 1997). Given a boolean query Q and a cluster of documents R, their similarity is measured by:

$$S(Q,R) = \frac{|\ \psi_R(Q) \bigcap \pi(R)\ |}{|\ \psi_R(Q) \bigcup \pi(R)\ |}$$

where $\psi_R(Q)$ represent all the boolean matches of Q in the cluster R and $\pi(R)$ is the set of documents from cluster R. This similarity measure is used to estimate the validity of the clustered index, and thus indicates a relevance measure for a meta-search based on boolean indexing. Such retrieval methods have been employed by the *Gloss* system (Gavrano et al.1999) and the *Indie* system (Danzig et al.1992).

Indexing based on NLP

The idea of using NLP techniques to improve IR performance is not new (Strzalkowski 1995). NLP parsing is especially useful, since it detects syntactic phrasal terms that can improve indexing quality. Parses are especially useful for building Q/A systems, since subject-verb-object (SVO) patterns can be extracted as well as prepositional attachments. Syntactic information coupled with indexes to words in documents helps guiding the path to a question's answer. However, syntactic and semantic ambiguities are problems that make automatic NLP techniques hard to be used for Q/A.

START (Katz 1997), the Q/A system built at MIT, employs knowledge of verb alternations to index sentences from documents. Semantic and taxonomic information is used as well. However, the high performance of START is based on heavy manual annotations, thus a commodity that we cannot obtain before the evaluation of the TREC Q/A systems.

Another indexing method based on NLP was designed at Sun Microsystems (Woods 1997). *Conceptual indexing* is the result of automatically creating a conceptual knowledge index, through classification algorithms that combine semantic, morphological and syntactic information. This indexer is particularly interesting, since it provides with interpretation of complex nominals, and thus builds a taxonomy of concepts recognized from large collections of texts.

The anatomy of several new Internet search engines (e.g. Google (Brin and Page 1998)) shows that many indexes are created by using phrasal parsers and identifying terms as heads of phrases or complex nominals.

A third indexing method was introduced in (Green 1998), and is an automatic way of generating hyperlinks that is based on *lexical chaining*. Lexical chaining is a method of detecting semantically related words in a text, when using a thesaurus (originally the Roget's Thesaurus, later WordNet). Lexical chains are used to measure similarity between documents at paragraph level. The similarity is measured by overlap or cosine coefficients between weighted paragraph vectors. The weight of a paragraph is given by its density. The density of a chain c in a paragraph p is defined as $d_{c,p} = \frac{w_{c,p}}{w_p}$, where $w_{c,p}$ represents the number of words from chain c in paragraph p whereas w_p is the number of content words from p.

Indexing is important because it retrieves paragraphs or fragments of texts where the answer may be found, but the recognition of the answer has to rely on logical derivations. We argue that indexing provides with weight measures that can be used in the weighted abduction method reported in (Hobbs et al.1993).

Abduction of answers

Given a fragment of text, we infer that it is an answer of a question if an explanation, based on semantic knowledge can be found. The process of interpreting texts by providing explanations of why they are the answers to a question is known as abductive inference. We distinguish two forms of abduction. The first one applies to the interpretation of taxonomic information, used in explaining answers to questions similar to those listed in Table 5. The second form is based on weighted abduction, as presented in (Hobbs et al.1993).

Q1.1	Who was Lincoln's Secretary of State?
Q1.2	Who were the US presidents from last century?
Q1.3	What forms of international crime exist?
Q1.4	What are some negative phenomena in society?

Table 5: Questions asking for taxonomic information.

The abduction of questions similar to those from Table 5 is often based on the interpretation of appositions and on the recognition of lexico-syntactic patterns for hyponymy or troponymy. For example, the apposition `Secretary of State William Seward` from the answers listed in Table 3 indicates that William Seward is a Secretary of State, but does not indicate that he was Lincoln's Secretary of State, to provide with an answer to the question Q1.1. To be able to abductively infer this, we need to rely on the following pragmatic, general axioms:

action1 (e1, Person1) & action2 (e2,Person2) &
& related_events(e1,e2) => related (Person_1, Person_2)

same_time(e1,e2) => related_events(e1,e2)

isa(e1,e2) => related_events(e1,e2)

Using the davidsonian treatment of events, these axioms state that two events e_1 and e_2, if lexicalized by two different verbs or nominals ($action_1$ and $actions_2$)

entail a relation between the persons affected by the events when there is a relation between the events as well. Possible relations between events are either the fact that they occur at the same time, or the fact that one is the troponym of the other.

The abduction explains the relations between President Lincoln and Secretary of State William Seward. Backward chaining accounts for the explanation of the relations between the events of `killing` and `murdering`, because we can find an *isa* relation in WordNet 1.6 between the sense 1 of verb `murder` and the first sense of verb `kill`. The cue phrase `while` also indicates that the two events co-occur.

Similarly the abduction of the answer to question Q1.2 is based on axioms representing knowledge about the definition of the last century. Such axioms can be derived from the defining glosses of WordNet 1.6 (Miller 1995), since we use the same lexical knowledge base for retrieval of semantic information.

In contrast, the abduction of the answer to question Q1.3 is based on the recognition of *lexico-syntactic patterns* for hypernymy (subsumption) and meronymy (e.g. is-part, is-member relations). Several such patterns were presented in (Hearst 1998), together with an algorithm for the acquisition of such patterns from large corpora. Table 6 lists some of the patterns and examples of their recognition.

Pattern 1	$noun_0$ `such as` $noun_1$ {, $noun_2$ ..., (and|or) $noun_i$}
Example	*authors such as Herrick, Goldsmith and Shakespeare*
Pattern 2	$noun_1$ {, $noun_2$ ...} (or|and) `other` $noun_0$
Example	*bruises, broken bones or other injuries*
Pattern 3	$noun_0$ `including` {$noun_1$ ...} (or|and) $noun_n$
Example	*common-law countries, including Canada and England*
Pattern 4	$noun_0$ `especially` {$noun_1$ ...} (or|and) $noun_n$
Example	*European countries, especially France, England and Spain*

Table 6: Lexico-syntactic patterns.

The answer to question Q1.3 is infered because Pattern 3 from Table 6 could be recognized in the following paragraph:

From the point of view of guaranteeing the Slovak Republic's internal security, risks associated with the following are at the forefront:
- *the rise of nationalism and irredentism among a certain part of the population with possible tendencies toward violating the integrity of the state's territory.*
- *the rise of organized crime and [international forms of crime, including terrorism, blackmail and drug-related problems.]*

Similarly, the answer to question Q1.4 is infered because Pattern 2 from Table 6 could be recognized in the following paragraph:

It is important to motivate the people to participate in the campaing against [corruption, smuggling, the production of fake goods and other negative phenomenon of society.]

Lexico-syntactic patterns and appositions cannot account for the majority of answer abductions. However, they provide with a simple method of finding instances and subsumers of entities, states, events or attributes, and thus help proving relations between terms of the question. As we have seen in the abduction of the answer to Q1.1, some of the relations are more important than others. The fact that murdering is a kind of killing is more important in proving the relations between President Lincoln than Secretary of State William Seward than the fact that the two events take place at the same time. Therefore, we believe that the two relation predicates should have different weights in the explanation.

This assumption agrees with the principles of weighted abduction, presented in (Hobbs et al.1993). In that framework, axioms are represented in the format:

$$P_1^{w_1} \wedge P_2^{w_2} ... \Longrightarrow Q \wedge R$$

where $P_i^{w_i}$ are antecedents participating with different weights in explaining the conclusion Q. R is the background information, that helps the backchaining. The weights implement the intuitions of what predicates contribute more to the derivation of a certain conclusion.

The assignment of the weights has an ad-hoc nature in (Hobbs et al.1993), thus using the values of the weights from the index might be an interesting solution. We experiment now to see whether this solution gives better performance than the weighted abduction schemes proposed in (Charniak and Shimony 1990).

An alternative is to compute the weights with semantic density measures over lexico-semantic networks derived from WordNet 1.6 (Miller 1995). These networks are collections of paths that account for the abduction of answers to question. The paths are derived in a two step process. First paths between the terms of the questions are derived. Then, paths to terms co-occuring with question concepts in the same paragrapg are sought. The procedure that derives these paths is:

1. *For all C_i, concepts from the question*
2. *Search the glosses of all senses of every C_i*
3. *If there is a collocation (w_1, w_2)* such that
   w_1 is a synonym, hypernym, holonym of a C_1
   w_2 is a synonym, hypernym, holonym of a C_2
4. *Then found path(C_1, C_2)*
5. *For all glosses implemented in WordNet*
6. *If there is a collocation (w_1, w_2)* in
   the gloss of (S_k), such that
   w_1 is a synonym, hypernym, holonym of a C_1
   w_2 is a synonym, hypernym, holonym of a C_2
   S_k is related to some C_3
7. *Then found path(C_1, C_2)*
8. *found path(C_1, C_3)*

The paths obtained for the question "How many animals and plants are threatened by extinction?" are represented in Figure 4. To be able to infer that the answer to this question lies in the following fragment of a paragraph, we need to explain why sliding toward extinction is equivalent to being threatened by extinction:

Fifteen years after Congress enacted a law to protect endangered species, most of the nearly 500 types of animals and plants show no sign if recovery and many continue their slide toward extinction.

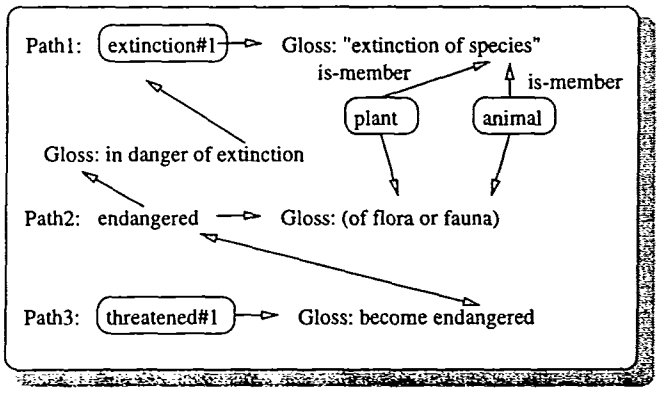

Figure 4: Semantic paths obtained from WordNet

The abductive inference follows the shortest path from the answer concepts to the question paths. For this example, because sliding is a kind of slowly changing, and threatening is a process of becoming endangered, we can use the knowledge that becoming is a form of changing as well, and assign a larger weight to the threaten predicate.

Conclusions

We believe that the nature and complexity of the task of textual Q/A will impact research in NLP in a major way. First, new indexing technologies will bootstrap existing IR systems. Second, the need of validating answers will boost knowledge based reasoning methods for NLP. After the IE field has used as its best the shallow methods of processing texts, we take a new step by incrementally adding techniques that rely on knowledge resources. This endeavor will bring closer the NLP and knowledge processing communities. Research in abduction, an inference technique believed to be well suited for NLP processing, is expected to take a central place in textual Q/A.

References

Sergey Brin and Lawrence Page. The anatomy of a Large-Scale Hypertextual Web Search Engine. In the *Proceedings of the Seventh International World Wide Web Conference*, 1998.

Chris Buckley, Mandar Mitra, Janet Walz and Claire Cardie. SMART Hight Precision: TREC 7. In the *Proceedings of the Text Retrieval Conference TREC-7*, 1998.

Eugene Charniak and S.E. Shimony. Probabilistic semantics for cost-based abduction. Technical Report CS-90-02, Department of Computer Science, Brown University, Providence RI, 1990.

Paul Cohen, Robert Schrag, Eric Jones, Adam Pease, Albert Lin, Barbara Starr, David Gunning and Murray Burke. The DARPA High Performance Knowledge Bases Project. In *AI Magazine*, Vol 18, No 4, pages 25–49, 1998.

Paul Cohen, Vinay Chaudhri, Adam Pease and Robert Schrag. Does Prior Knowledge Facilitate the Development of Knowledge-Based Systems? To be pubblished in the *Proceedings of AAAI-99*, 1999.

Peter Danzig, Shih-Hao Li and Katia Obracczka. Distributed Indexing of Autonomous Internet Services. *Computing Systems*, Vol 5, No 4, pages 433–459, 1992.

Luis Gravano, Hector Garcia-Molona amd Anthony Tomasic. Gloss: Text-Source Discovery over the Internet. To be pubblished in the *AI Journal*, 1999.

Stephen J. Green. Automatically generating hypertext by computing semantic similarity. PhD thesis, University of Toronto, 1997.

Stephen J. Green. Automatic link generation: Can we do better than term repetition? In the *Proceedings of the Seventh International World Wide Web Conference*, 1998.

Sanda Harabagiu and Dan Moldovan. A Parallel System for Text Inference Using Marker Propagations. *IEEE Transactions in Parallel and Distributed Systems*, Vol 9, no 8, pages 729–748, 1998.

Sanda Harabagiu and Dan Moldovan. Knowledge Processing on Extended WordNet. In *WordNet: An Electronic Lexical Database and Some of its Applications*, editor Fellbaum, C., MIT Press, Cambridge, MA, 1998.

Marti Hearst. Automated Discovery of WordNet Relations. In *WordNet: An Electronic Lexical Database and Some of its Applications*, editor Fellbaum, C., MIT Press, Cambridge, MA, 1998.

Lynette Hirschman, Marc Light, Eric Breck and John D. Burger. Deep Read: A Reading Comprehension System. In the *Proceedings of the 37th Meeting of the Association for Computational Linguistics (ACL-99)*, pages 325–332, University of Maryland, 1999.

Jerry Hobbs, Mark Stickel, Doug Appelt, and Paul Martin. Interpretation as abduction. *Artificial Intelligence*, 63, pages 69–142, 1993.

Boris Katz. From Sentence Processing to Information Access on the World Wide Web. *Proceedings of the AAAI Spring Symposium*, pages 77–86, 1997.

Julian Kupiec. MURAX: A Robust Linguistic Approach for Question Answering Using an On-line Encyclopedia. In the *Proceedings of the 16th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-93)*, pages 181–190, Pittsburg, PA, 1993.

P.S. Jacobs. Text Power and Intelligent Systems. In Lawrence Erlbaum Associates, *Text-Based Intelligent Systems*, pages 1–8. ,editor Jacobs, P.S., Hillsdale, N.J., 1992.

Steve Lawrence and Lee Giles. Inquirus, the NECI meta search engine. In the *Proceedings of the Seventh International World Wide Web Conference*, 1998.

D.B. Lenat. CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communication of the ACM*, vol 38: No11, pages 32–38, November, 1995.

Shih-Hao Li and Peter Danzig. Boolean Similarity Measures for Resource Discovery. *IEEE Transactions on Knowledge and Data Engineering*, Vol 9, No 6, pages 848-862, 1997.

G.A. Miller. WordNet: A Lexical Database. *Communication of the ACM*, vol 38: No11, pages 39–41, November 1995.

Dan Moldovan and Rada Mihalcea. A WordNet-based Interface to Internet Search Engines. In *Proceedings of the FLAIRS-98*, pages 275–279, 1998.

Gerard A. Salton and M.E. Lesk. Computer Evaluation of Indexing and Text Processing. *Journal of the ACM*, Vol 15, No 1, Pages 8–36, January 1968.

Gerard A. Salton. A theory of indexing. *Regional Conference Series on Applied MAthematics*, No 18, SIAM, Philadelphia, PA, 1975.

Gerard A. Salton. Automatic Text Processing: The transformation, analysis and retrieval of information by computer. Addison-Wesley, 1989.

Tomek Strzalkowski. Robust Text Processing in Automated Information Retrieval. In *Readings in Information Retrieval*, pages 317–322, 1995.

B. Sundheim. Proceedings. *Sixth Message Understanding Conference (MUC-6)* August, 1995.

C.J. van Rijsergen. Information Retrieval. London, Butterworth & Co, 1979. August, 1995.

William A. Woods. *Conceptual Indexing: A Better way to Organize Knowledge.* Technical Report of Sun Microsystems Inc., 1997.