MURAX: A Robust Linguistic Approach For Question Answering Using An On-Line Encyclopedia

Julian Kupiec

Xerox Palo Alto Research Center
3333 Coyote Hill Road, Palo Alto, CA 94304

Abstract

Robust linguistic methods are applied to the task of answering closed-class questions using a corpus of natural language. The methods are illustrated in a broad domain: answering general-knowledge questions using an on-line encyclopedia.

A closed-class question is a question stated in natural language, which assumes some definite answer typified by a noun phrase rather than a procedural answer. The methods hypothesize noun phrases that are likely to be the answer, and present the user with relevant text in which they are marked, focussing the user's attention appropriately. Furthermore, the sentences of matching text that are shown to the user are selected to confirm phrase relations implied by the question, rather than being selected solely on the basis of word frequency.

The corpus is accessed via an information retrieval (IR) system that supports boolean search with proximity constraints. Queries are automatically constructed from the phrasal content of the question, and passed to the IR system to find relevant text. Then the relevant text is itself analyzed; noun phrase hypotheses are extracted and new queries are independently made to confirm phrase relations for the various hypotheses.

The methods are currently being implemented in a system called MURAX and although this process is not complete, it is sufficiently advanced for an interim evaluation to be presented.

16th Int'l SIGIR'93/Pittsburgh PA USA-6/93

1 Introduction

The paper is organized as follows. First the motivation for the question-answering task is given and a description of the kind of questions that are its concern, and their characteristics. A description of the system components is given in Section 3. These include the encyclopedia and the IR system for accessing it. Shallow linguistic analysis is done using a part-of-speech tagger and finite-state recognizers for matching lexico-syntactic patterns.

Section 4 describes the analysis of a question by considering an example, and the system output is illustrated. Analysis proceeds in two stages. The first, primary query construction, finds articles that are relevant to the question. The second stage (called answer extraction) analyzes these articles to find noun phrases (called answer hypotheses) that are likely to be the answer.

Both stages require searching the encyclopedia. Queries made during the first stage are called primary queries, and only involve phrases from the question. The second stage creates secondary queries which are generated by MURAX to verify specific phrase relations. Secondary queries involve both answer hypotheses and phrases from the question.

Primary query construction is explained in Section 5, followed by a complete description of answer extraction in Section 6. An informal evaluation and discussion are then presented.

2 Task Selection

The task is concerned with answering general-knowledge questions using Grolier's on-line encyclopedia. The task is motivated by several criteria and and goals. Robust analysis is needed because the encyclopedia is composed of a significant quantity of unrestricted text. General-knowledge is a broad domain, which means that it is impractical to manually provide detailed lexical or semantic information for the words of the vocabulary (the encyclopedia contains over 100,000 word stems). The methods demonstrate that shallow syntactic analysis can be used to practical advantage in broad domains, where the types of relations and objects involved are not known in advance, and may differ for each new question. The analysis must capitalize on the information available in a question, and profit from treating the encyclopedia as a lexical resource.

The methods also demonstrate that natural language analysis can add to the quality of the retrieval process, providing text to the user which confirms phrase relations and not just word matches. The task also serves as a practical focus for the development of linguistic tools for content analysis and reveals what kind of grammar development should be done to improve performance.

The use of closed-class questions means that performance can be evaluated in a straightforward way by using a set of questions and correct answers. Given a correct noun phrase answer, it is generally easy to judge whether a noun phrase hypothesized by the system is correct or not. Thus relevance judgements are simplified, and if one correct hypothesis is considered as good as any other, recall measurements are not required and performance can be considered simply as the percentage of correctly hypothesized answers.

1. What U.S. city is at the junction of the Allegheny and Monongahela rivers?
2. Who wrote "Across the River and into the Trees"?
3. Who married actress Nancy Davis?
4. What 's the capital of the Netherlands?
5. Who was the last of the Apache warrior chiefs?
6. What chief justice headed the commission that declared: "Lee Harvey Oswald ... acted alone."?
7. What famed falls are split in two by Goat Island?
8. What is November's birthstone?
9. Who 's won the most Oscars for costume design?
10. What is the state flower of Alaska?

Figure 1: Example Questions

2.1 Question Characteristics

A closed-class question is a direct question whose answer is assumed to lie in a set of objects and is expressible as a noun phrase. Such questions are exemplified in Figure 1. These questions appear in the general-knowledge "Trivial Pursuit"[1] game and typify the form of question that is the concern of the task. They have the virtue of being created independently of the retrieval task (i.e. are unbiassed) and have a consistent and simple stylized form; yet they are flexible in their expressive power.

Who/Whose:	*Person*		
What/Which:	*Thing,*	*Person,*	*Location*
Where:	*Location*		
When:	*Time*		
How Many:	*Number*		

Table 1: Question Words and Expectations

The interrogative words that introduce a question are an important source of information. They indicate particular expectations about the answer and some of these are illustrated in Table 1. Notable omissions are the words *why* and *how*, expecting a procedural answer rather than a noun phrase [2] (e.g. "How do you make a loaf of bread?").

These expectations can be used to filter various answer hypotheses. The answers to questions beginning with the word "who" are likely to be people's names. This fact can be used to advantage because various heuristics can be applied to verify whether a noun phrase is a person's name.

A question introduced by "what" may or may not refer to a person; however, other characteristics can be exploited. Consider the following sentence fragments, where *NP* symbolizes a noun phrase: "What is the *NP*..." and "What *NP*...". The noun phrase at the start of such questions is called the question's *type phrase* and it indicates what type of thing the answer is. The encyclopedia can be searched to try to find evidence that an answer hypothesis is an instance of the type phrase (details are in Section 6.1.1). The verbs in a question are also a useful source of information as they express a relation that exists between the answer and other phrases in the question.

The answer hypotheses for "Where ..." questions are likely to be locations, which often appear with locative prepositions or as arguments to verbs of motion. Questions of the form "When ..." often expect answer hypotheses that are dates or times and the expectation of questions beginning "How many ..." are numeric expressions.

Closed-class questions are also addressed by a system [Wendlandt and Driscoll, 1991] for accessing public in-

[1] Copyright Horn Abbot Ltd., Trivial Pursuit is a Registered Trademark of Horn Abbot Ltd.

[2] Questions requiring procedural answers are not considered unimportant, but of more concern after initial goals have been attained.

formation documents at NASA Kennedy Space Center (e.g. "What are the dimensions of the cargo area in the shuttle?"). In the system, conventional word-based similarity measures are augmented with terms for thematic roles, obtained from a manually constructed lexicon.

3 Components

An on-line version of Grolier's Academic American Encyclopedia [Grolier, 1990] was chosen as the corpus for the task. It contains approximately 27,000 articles, which are accessed via the Text Database (TDB) [Cutting *et al.*, 1991], which is a flexible platform for the development of retrieval system prototypes and is structured so that additional functional components (e.g. search strategies and text taggers [Cutting *et al.*, 1992]) can be easily integrated.

The components responsible for linguistic analysis are a part-of-speech tagger and a lexico-syntactic pattern matcher. The tagger is based on a hidden Markov model (HMM). HMM's are probabilistic and their parameters can be estimated by training on a sample of ordinary untagged text. Once trained, the Viterbi algorithm is used for tagging. To assess performance, an HMM tagger [Kupiec, 1992b] was trained on the untagged words of half of the Brown corpus [Francis and Kučera, 1982] and then tested against the manually assigned tags of the other half. This gave an overall error rate of 4% (corresponding to an error rate of 11.2% on words that can assume more than one part-of-speech category). The percentage of tagger errors that affect correct recognition of noun phrases is much lower than 4%. The tagger uses both suffix information and local context to predict the categories of words for which it has no lexicon entries.

The HMM used for tagging the encyclopedia text was also trained using the encyclopedia. A benefit of such training is that the tagger can adapt to certain characteristics of the domain. An observation in this regard was made with the word "I". The text of the encyclopedia is written in an impersonal style and the word is most often used in phrases like "King George I" and "World War I". The tagger trained on encyclopedia text assigned "I" appropriately (as a proper noun) whereas the tagger trained on the Brown corpus (a mixture of different kinds of text) assigned such instances as a pronoun.

Given a sentence of text, the tagger produces a sequence of pairs of words with associated part-of-speech categories. These enable phrase recognition to be done. Phrases are specified by regular expressions in the finite-state calculus [Hopcroft and Ullman, 1979]. Noun phrases are identified solely by part-of-speech categories, but more generally categories and words are used to define lexico-syntactic patterns against which text is matched. This kind of pattern matching has also been exploited by others (e.g. [Jacobs *et al.*, 1991, Hearst, 1992]).

Initially, only simple noun phrases are identified because they are recognized with the greatest reliability. Analysis involving prepositional phrases or other coordination is applied subsequently as part of more detailed matching procedures. Word-initial capitalization was found to be useful for splitting a noun phrase appropriately, thus "New York City borough" is split into "New York City" and "borough". Such splitting improves the efficiency of boolean query construction (enabling direct phrase matches, rather than requiring several words to be successively dropped from the phrase).

3.1 Title Phrases

A multi-word phrase that is the title of a film, book, play, etc., is usefully treated as a single unit. Furthermore, it may not be a simple noun phrase (e.g. Play Misty for Me). Such phrases are readily identified when marked typographically by enclosing quotes or italics. However, title phrases may be marked only by word-initial capitalized letters; furthermore, some words (such as short function words) may not be capitalized. Thus, the correct extent of the phrase may be ambiguous and alternative possibilities must be accommodated. The most likely alternative is chosen after phrase matching has been done and the alternatives compared, based on the matches and frequency of the alternative interpretations.

4 Operational Overview

This section presents an informal description of the operation of the system, by tracing the analysis steps for an example question, shown in Figure 2.

"Who was the Pulitzer Prize-winning novelist that ran for mayor of New York City ?"

Pulitzer Prize	winning novelist
mayor	New York City

Figure 2: Example Question and Component NP's

4.1 Primary Document Matches

Simple noun phrases and main verbs are first extracted from the question, as illustrated in the figure. These question phrases are used in a query construction/refinement procedure that forms boolean queries with associated proximity contraints (Section 5). The queries are used to search the encyclopedia to find a list of relevant articles from which primary document matches are made. These are sentences containing one or more of the question phrases.

Primary document matches are heuristically scored according to the degree and number of matches with the question phrases. Matching head words in a noun phrase receive double the score of other matching words in a phrase. Words with matching stems but incompatible part-of-speech categories are given minimal scores. Primary document matches are then ranked according to their scores.

4.2 Extracting Answers

It is assumed that primary document matches contain answer hypotheses, so answer extraction begins by finding all simple noun phrases contained in them. Each noun phrase is an answer hypothesis distinguished by its components words, and the article and sentence in which it occurs. Answer hypotheses are themselves scored on a per-article basis according to the sum of the scores of primary document matches in which they occur. The purpose of this is to minimize the probability of overlooking the correct answer hypothesis if a subsequent non-exhaustive search is performed using the hypotheses.

For each answer hypothesis the system tries to verify phrase relations implied by the question. For the question in Figure 2, we note that the answer is likely to be a person (indicated by "who"). The type phrase indicates the answer is preferably a "Pulitzer Prize winning novelist", or at least a "novelist" as indicated by the head noun of the type phrase. The relative pronoun indicates that the answer also "ran for mayor of New York City". Phrase matching procedures (detailed in Section 6) perform the verification using the answer hypotheses and the primary document matches, but the verification is not limited to primary document matches.

It can happen that a pertinent phrase relation is not present in the primary document matches although it can be confirmed elsewhere in the encyclopedia. This is because too few words are involved in the relation in comparison to other phrase matches, so the appropriate sentence does not rank high enough to be in the selected primary document matches. It is also possible that the appropriate information is not expressed in any primary document match and depends only on the answer hypothesis. This is the case with one heuristic that the system uses to try and verify that a noun phrase represents a person's name. The heuristic involves looking for an article that has the noun phrase in its title; thus if the article does not share any phrases with the question, it would not be part of any primary document match.

Secondary queries are used as an alternative means to confirm phrase relations. A secondary query may consist of solely an answer hypothesis (as for the heuristic just mentioned) or it may also include other question phrases such as the question's type phrase. To find out whether an answer hypothesis is a "novelist", the two phrases are included in a query and a search yields a list of relevant articles. Sentences which contain co-occurrences are called secondary document matches. The system analyzes secondary document matches to see if answer hypotheses can be validated as instances of the type phrase via lexico-syntactic patterns.

The best matching phrase
for this question is: **Mailer, Norman**

The following documents were most relevant:

Document Title: **Mailer, Norman**
Relevant Text:

- "The Armies of the Night (1968), a personal narrative of the 1967 peace march on the Pentagon, won **Mailer** the **Pulitzer Prize** and the National Book Award."
- "In 1969 **Mailer** ran unsuccessfully as an independent candidate for **mayor** of **New York City**."

Document Title: novel
Relevant Text:

- "Among contemporary American **novelists**, Saul Bellow, John Dos Passos, John Hawkes, Joseph Heller, **Norman Mailer**, Bernard Malamud, Thomas Pynchon, and J. D. Salinger have reached wide audiences."

Next best: Edith Wharton, William Faulkner

Figure 3: Example Output

4.3 System Output

For the given question the system produces the output shown in Figure 3. The presentation is different from extant IR systems. Answer hypotheses are shown to the user to focus his attention on likely answers and how they relate to other phrases in the question. The text presented is not necessarily from documents that have high similarity scores, but those which confirm phrase relations that lend evidence for an answer. This behaviour is readily understood by users, even though they have not been involved in the tedious intermediate work done by the system.

In Figure 3, the first two sentences are from primary document matches. The last sentence confirming Norman Mailer as a novelist is a secondary document match. It was confirmed by a lexico-syntactic pattern which identifies the answer hypothesis as being in a list-inclusion relationship with the type phrase.

We next consider this approach in contrast to a common alternative, vector-space search. Vector-space search using full-length documents is not as well suited to the task. For the example question, a search was done using a typical similarity measure and the bag of content words of the question. The most relevant document (about Norman Mailer) was ranked 37th. Somewhat better results could be expected if sentence or paragraph level matching was done (cf. [Salton and Buckley, 1991]). However the resulting text matches do not have the benefit of being correlated in terms of a particular answer and they muddle information for different answer hypotheses.

5 Primary Query Construction

This section describes how phrases from a question are translated into boolean queries with proximity constraints. These are passed to an IR system which searches the encyclopedia and returns a list of matching documents (or *hits*). The following functionality is assumed of the IR system:

1. The boolean AND of terms, denoted here as: $[term_1, term_2, ...term_n]$
2. Proximity of a strict sequence of terms, separated by up to p other terms denoted here as: $\{p\ term_1, term_2, ...term_n\}$
3. Proximity of an unordered list of terms, separated by up to p other terms denoted here as: $(p\ term_1, term_2, ...term_n)$

The overall process is again illustrated via an example question:

"Who shot President Lincoln ?"

The question is first tagged and the noun phrases and main verbs are found. In the above case the only noun phrase is *President Lincoln* and the main verb is *shot*. Boolean terms are next constructed from the phrases. At the outset a strict ordering is imposed on the component words of phrases. For the preceding question, the first query is:

{0 president lincoln}

The IR system is given this boolean query and searches for documents that match. Depending on the number of hits, new boolean queries may be generated with the purpose of:

1. Refining the ranking of the documents.
2. Reducing the number of hits (Narrowing).
3. Increasing the number of hits (Broadening).

Iterative broadening and narrowing has been investigated for the situation where phrase structure is not considered [Salton *et al.*, 1983].

5.1 Narrowing

Items (1) and (2) above are performed by using title phrases (Section 3.1) rather than the noun phrases, or by adding extra query terms such as the main verbs and performing a new search in the encyclopedia. Including the main verb in the example gives:

[{0 president lincoln} shot]

Narrowing is done to try to reduce the number of hits. It also involves reducing the co-occurrence scope of terms in the query and constrains phrases to be closer together (and thus indirectly there is a higher probability of them being in some syntactic relation with each other). A sequence of queries with increasingly smaller scope are made, until there are fewer hits than some predetermined threshold. A narrowed version for the previous example is shown below:

(10 {0 president lincoln} shot)

5.2 Broadening

Broadening is done to try and increase the number of hits for a boolean query. It is achieved in three ways:

1. Increasing the co-occurrence scope of words within phrases, while jointly dropping the requirement for strict ordering of the words. E.g. (5 president lincoln) would match the phrase "President Abraham Lincoln". A sequence of queries with increasingly larger scope are made until some threshold on either the proximity or resulting number of hits is reached.
2. Dropping one or more whole phrases from the boolean query. Query terms, each corresponding to a phrase, are dropped to get more hits. It is efficient to drop them in an order that corresponds to decreasing number of overall occurrences in the encyclopedia.

3. Dropping one or more words from within multiple-word phrases in a query to produce a query that is composed of sub-phrases of the original. In the previous example, to increase the number of hits *president* could be dropped, and so might *lincoln*.

5.3 Control Strategy

The initial boolean query comprises all the noun phrases derived from the user's question. Broadening and/or narrowing are then applied. Although a strict prioritization of operations does not seem necessary, the following partial order is effective:

1. Co-occurrence scope is increased before terms are dropped.
2. Single phrases are dropped from a query before two phrases are dropped.
3. Higher frequency phrases are dropped before lower frequency ones.
4. Title phrases are tried before any of their component noun phrases.
5. Complete phrases are used before their sub-phrases.

The iterative process of broadening and/or narrowing terminates when either a threshold on the number of hits has been reached, or no further useful queries can be made. Upon termination the hits are ranked. In practice it is not necessary to provide elaborate ranking criteria and documents are ranked simply by the number of terms they have in common with the user's question.

6 Answer Extraction

This section completes the description of how the most likely answer hypotheses are found from the relevant sentences in the various hits. Phrase matching operations are considered first, followed by the procedure for constructing secondary queries to get secondary document matches. Generally several hypotheses may represent the same answer, so they must be linked together and their various phrase matches combined. They can then be ranked in order of likelihood.

6.1 Phrase Matching

Phrase matching is done with lexico-syntactic patterns which are described using regular expressions. The expressions are translated into finite-state recognizers, which are determinized and minimized [Hopcroft and Ullman, 1979] so matching is done efficiently and without backtracking. Recognizers are applied to primary and secondary matches, and the longest possible match is recorded.

An example pattern and text match is shown in Figure 4. For convenience, copies of expressions can be included by naming them in other expressions. In the figure, the expression NP1 refers to a noun phrase, whose pattern is defined elsewhere.

Regular Expression Operators:

+	One or more instances
?	Zero or one instances
{...}	sequence of instances
(...)	inclusive-or of instances

Lexico-Syntactic pattern:

```
{   NP1 (are were include {such as} )
 +{ NP2 ,}
 ?  NP3 ? { and NP4}}
```

Example match:

"Countries such as Egypt, Sudan, and Israel ..."
NP1 NP2 NP2 NP4

Figure 4: Example Pattern and Document Match

For robustness, phrase matching is layered on top of co-occurrence matching so if the input is not a question (or a question beginning with "how" or "why") the system provides output that is typical of co-occurrence based search methods.

A large corpus mitigates some of the problems inherent in using simple language modelling. In a document match, a relation may not be verified because it requires more sophisticated analysis than is feasible with a finite-state grammar. However, the relation may be expressed in several places in the encyclopedia and thus more simply in some places, improving the chances of verifying it.

Likewise it happens that spurious matches are also made by simple phrase matching. Other things being equal, an answer hypothesis having more instances of the match is preferred.

It is less likely that spurious matches for an answer hypothesis occur for several different phrase relations, so many of these errors don't propagate far enough to cause an erroneous answer.

6.1.1 Verifying Type Phrases

The following relations are used to try to verify answer hypotheses as instances of type phrases:

Apposition

This is exemplified by the match between the type phrase of the following question and the document match below it:

"Who was the last Anglo-Saxon king of England?"

1) "The last Anglo-Saxon king of England, Harold, b. c.1022, was defeated and killed at ..."

The IS-A Relation

This is demonstrated by the following document match:

2) "Saint Edward the Confessor, b. between 1002 and 1005, d. Jan. 5, 1066, was the next to last Anglo-Saxon king of England (1042-66)."

List Inclusion

Lists are often used to enumerate objects of the same type. Examples are shown in Figures 3 and 4.

Noun Phrase Inclusion

Type phrases are often related to answer hypotheses by being included in them. In the question and corresponding document match shown below, the type phrase *river* is in the same noun phrase as the answer hypothesis *Colorado River*:

"What river does the Hoover Dam dam?"

"...the Hoover Dam, on the Colorado River ..."

6.1.2 Predicate/Argument Match

This operation associates answer hypotheses and other noun phrases in a document match that satisfy a verb relation implied in a question. Currently verbs are simply assumed to be monotransitive and patterns accounting for active and passive alternation are applied. This is illustrated by the question and document match shown below:

"Who succeeded Shastri as prime minister?"

"...Shastri was succeeded by Indira Gandhi as Indian prime minister ..."

6.1.3 Minimum Mismatch

For reliable identification, simple noun phrases are extracted from primary document matches. For the question in Figure 1, the phrase "mayor of New York City" is first considered as two simpler and independent noun phrases. Exact matching of the overall noun phrase is done after all document matches are found. When comparing type phrases with answer hypotheses, the mimimum degree of mismatch is considered best. This is illustrated by considering the first question in Section 6.1.1 and the associated document matches (1) and (2). Both "Harold" and "Saint Edward the Confessor" match equally well with the type phrase "last Anglo-Saxon king of England". However, "Harold" is (correctly) preferred because the match is exact, whereas a longer match is involved for "Saint Edward the Confessor" (namely, he was the "next to last Anglo-Saxon king of England").

6.1.4 Person Verification

The confirmation of an answer hypothesis as a person's name is important. In the encyclopedia, a reliable property of peoples' names is that they have word-initial capital letters. This simple consideration significantly reduces the number of answer hypotheses that require further consideration.

Many different multi-national names are present and exhaustive manual enumeration is impractical. However there are indirect clues that can be used. Articles about people generally have their name as the title and in such cases there is often a mention at the beginning of the article of birth and/or death dates which are easily identified. Usually there is also a higher percentage of words that are male or female pronouns than in other articles. Thus to try and confirm an answer hypothesis as a person's name, a secondary query is made to see if it is present as a title, and then it is decided whether the article is about a person. This heuristic is simple, yet robust (and of course is open to improvement by more sophisticated analysis).

6.2 Secondary Queries

Secondary document matches are a supplementary means of confirming phrase relations and are found via secondary queries which are constructed by MURAX and passed to the IR system. Broadening is applied as necessary to secondary queries, but terms are never dropped because they are required in the resulting matches. For person verification, only an answer hypothesis is used in a secondary query, but other relations require other question phrases to be included. These are considered next.

6.2.1 Type Phrase Queries

Answer hypotheses are included verbatim in a query, but when trying to verify a type phrase, only the head word of the phrase is included. This provides the minimal necessary constraint on secondary document matches. The detailed matching of all words in the type phrase is done by considering the degree of mismatch with the type phrase (Section 6.1.3). When the type phrase cannot be matched against an answer hypothesis using any lexico-syntactic pattern, the fact of their co-occurrence in a sentence is still recorded, as it may serve as a means of ranking alternative hypotheses in the absence of any better information (the relation may still be implied by the document match, but cannot be inferred from the simple matching operations that are used).

6.2.2 Co-Occurrence Queries

It is expedient to include other question phrases in secondary queries. As mentioned in Section 4.2, a relevant phrase match may not be found because the primary document match in which it occurs has too low a score in comparison to other primary document matches. Creating secondary queries with individual question phrases allows the relevant phrase match to be found.

Secondary queries are also used to find co-occurrences of answer hypotheses and question phrases that extend beyond the context of a single sentence. This can be useful for ranking alternative answer hypotheses in the absence of other differentiating phrase matches. It is illustrated in the following question and primary document matches:

"What film pits Humphrey Bogart against gangsters in the Florida Keys?"

"...Bacall and Bogart became a famous romantic couple in films such as The Big Sleep (1946) and Key Largo (1948)."

"Some of his most popular films were The Maltese Falcon (1941); Casablanca (1942), with Ingrid Bergman; The Big Sleep (1946) costarring his wife, Lauren Bacall; The Treasure of Sierra Madre (1948); ..."

Secondary co-occurrence queries determine that the answer hypothesis *Key Largo* co-occurs with *Florida Keys*, but the other "film" hypotheses do not; thus in the absence of stronger evidence to the contrary, *Key Largo* receives a preference.

6.3 Equivalent Hypotheses

Answer hypotheses are identified by the article and sentence in which they occur, so it is generally the case that the same answer is expressed by several hypotheses. When hypotheses refer to a person, it is often the case that different word sequences may refer to the same person. For example, *President Kennedy, John F. Kennedy, President John F. Kennedy* all refer to the same person, and in particular articles so does *Kennedy.* In the latter case, reference to the title of the article helps to disambiguate the usage. In the document titled *Kennedy, John F.*, mention of *Kennedy* refers to the title, whereas other members of the family are named explicitly (e.g. *Joseph P. Kennedy*) so as not to confuse the reader.

After answer hypotheses are formed, equivalent ones are linked together, and then all hypotheses are rescored. The most general one in an equivalence class (usually the one that serves as a title) is assigned the cumulative score of all the members and used as the representative of the set. When a document title can be used as the representative, it usually provides the best description for the user. For example, for sentence (1) in Section 6.1.1, "Harold" is a correct hypothesis, which is better represented by the more definitive title "Harold II."

Similarly to surnames, the pronouns *he* and *she* often refer to the article's title (if the title is a person's name), or an immediately preceding name. For the example output shown in Figure 3, in the document displayed for Norman Mailer, the same answer hypothesis *Mailer* is used in both sentences, however if he had been referred to instead as *he*, the result would be the same.

To verify whether an answer hypothesis refers to a person's name, primary documents are first examined. If no verification occurs, a secondary query is used. If a title in the resulting matches is equivalent to the hypothesis and the title can be verified as a person's name, then the hypothesis is linked to the title.

6.4 Combining Phrase Matches

The various phrase matching procedures are used in concert to provide a ranked list of answer hypotheses. The system does this by providing several intermediate layers of scoring and preference criteria. A list of criteria for partially ordering hypotheses is given below, in order of precedence. It is typically the case that not all of these apply for any given question, however the lowest level (item [5]) always applies, and usually several other items also.

1. When type phrases occur, the highest ranking answer hypotheses are those with minimum mismatch.

2. The number of question phrases that co-occur with an answer hypothesis. This match is qualified by the number of different articles needed to match the most question phrases.

3. Predicate/argument matches produce preferences among different answer hypotheses.

4. For *who* questions, an answer hypothesis that is verified as a person takes precedence.

Rank of Correct Hypothesis	Nr. Questions
Top	37
Top 3	49
Top 5	52
Not in Top 5	18

Table 2: Interim Evaluation

5. Answer hypotheses are ranked in terms of of their co-occurrence with question phrases.

7 An Interim Evaluation

The implementation of the MURAX system is not yet complete so it is only possible to provide an interim evaluation. However enough programming has been completed to permit performance estimates for questions beginning with "who" and "what", which have simple noun phrase answers (i.e. excluding conjoined noun phrases such as "Richard Nixon and Nikita Khrushchev"). Seventy "Trivial Pursuit" questions were used for the evaluation, each of which was known to have a simple noun phrase answer. Additionally it was confirmed that the answers were present in the encyclopedia (i.e given a question and its answer, a person could find text in the encyclopedia from which the answer could be inferred using common-sense).

The evaluation is based on an objective, system-dependent criterion: the rank of the correct answer hypothesis. The results are shown in Table 2. It indicates that the best guess is the correct answer for half of the questions (53%), and the correct answer lies in the top five guesses for (74%) of the questions. Using a cutoff of five guesses, answers are thus considered to be found for 74% of the questions and the mean rank of the correct hypothesis is 1.44. The questions shown in Figure 1 were included in the evaluation. Correct answers were in the top five hypotheses for all the questions except (10). (The answer "forget-me-not" was not found because references involving impersonal pronouns are not currently considered. Such a reference is necessary to infer the answer from the sentence "It is the state flower of Alaska.")

8 Discussion

Analysis of the errors made by the current state of the MURAX system both confirms the expectation that a complete implementation will improve performance, and suggests further ways of improving performance. A comprehensive evaluation will take account of not only the rank of correct hypotheses, but also the suitability of the document matches that are presented to the user to verify phrase relations (this would also provide a means of comparing other search techniques that perform sentence-level matching).

The current implementation is not optimized and does not operate on an interactive timescale (processing time varies, but can require ten or more minutes for a question). Currently any article that is accessed is tagged and analyzed in its entirety, when in fact it is only necessary to analyze specific sentences. If interactive operation were a primary goal, it is likely that it could be realized with little loss (if any) in performance.

9 Future Work

The MURAX system is a means for investigating how natural language methods can be used for intelligent information retrieval. Beyond completing the system described here, many possibilities exist for further development. Task evaluation indicates where further effort might be most productive and also indicates how new components contribute to overall performance.

In particular the WordNet thesaurus [Miller *et al.*, 1990] appears well-suited to the task and could provide useful synonym and hyponym information. For example, consider the question "What Pulitzer Prize-winning novelist ran for mayor of New York City ?". WordNet indicates that *novelist* is a hyponym of *writer*, *author* and also *person*. This means that the answer is likely to be a person's name even though the question starts with "what". Furthermore, any type phrase matches involving the words *writer* or *author* are also relevant.

The linguistic analysis is based on an underlying regular grammar formalism, both in the HMM tagger and the phrase recognizers. There may be benefits from the use of stochastic context-free grammars, which can also be trained from unlabelled text [Kupiec, 1992a] and enable ambiguity to be quantified in probabilistic terms.

10 Acknowledgments

I would like to thank colleagues at Xerox PARC, particularly Doug Cutting, Ron Kaplan, Lauri Karttunen, and Jan Pedersen, who have provided support that has made this work possible.

References

[Cutting *et al.*, 1991] D.R. Cutting, J. Pedersen, and P.-K. Halvorsen. An object-oriented architecture for text retrieval. In *Conference Proceedings of RIAO'91, Intelligent Text and Image Handling, Barcelona, Spain*, pages 285–298, April 1991.

[Cutting *et al.*, 1992] D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, Trento, Italy, April 1992. ACL.

[Francis and Kučera, 1982] W. N. Francis and F. Kučera. *Frequency Analysis of English Usage.* Houghton Mifflin, 1982.

[Grolier, 1990] The Academic American Encyclopedia. Grolier Electronic Publishing, Danbury, Connecticut, 1990.

[Hearst, 1992] M. A. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 539–545, Nantes, France, 1992.

[Hopcroft and Ullman, 1979] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley, 1979.

[Jacobs *et al.*, 1991] P. S. Jacobs, G. R. Krupka, and L. F. Rau. Lexico-Semantic pattern matching as a companion to parsing in text understanding. In *Proceedings of the Fourth DARPA Speech and Natural Language Workshop*, pages 337–342, San Mateo, CA, February 1991. Morgan Kaufmann.

[Kupiec, 1992a] J. M. Kupiec. Hidden Markov estimation for unrestricted stochastic context-free grammars. In *Proceedings of the 1992 International Conference on Acoustics, Speech and Signal Processing*, pages I–177–180. IEEE Signal Processing Society, IEEE, March 1992.

[Kupiec, 1992b] J. M. Kupiec. Robust part-of-speech tagging using a hidden Markov model. *Computer Speech and Language*, 6:225–242, 1992.

[Miller *et al.*, 1990] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Five papers on WordNet. Technical report, Princeton University, Computer Science Laboratory, July 1990.

[Salton and Buckley, 1991] G. Salton and C. Buckley. Automatic text structuring and retrieval: Experiments in automatic encyclopedia searching. In *Proceedings of the Fourteenth International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–30, October 1991.

[Salton *et al.*, 1983] G. Salton, C. Buckley, and E. A. Fox. Automatic query formulations in information retrieval. *Journal of the American Society for Information Science*, 34(4):262–280, July 1983.

[Wendlandt and Driscoll, 1991] E. B. Wendlandt and J. R. Driscoll. Incorporating a semantic analysis into a document retrieval strategy. In *Proceedings of the Fourteenth International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 270–279, October 1991.