Sapere: Improving the Precision of Information Retrieval Systems Using Syntactic Relations

Boris Katz & Jimmy Lin

Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

http://www.ai.mit.edu

The Problem: Traditional information retrieval systems based on the "bag-of-words" paradigm cannot capture the semantic content of documents. While these systems are relatively robust and have high recall, they suffer from very poor precision. On the other hand, it is impossible with current technology to build a practical information access system that fully analyzes and understands unrestricted natural language. Existing natural language systems, despite their high precision, have low recall and lack robustness.

Motivation: By simplifying the sophistication of natural language techniques applied to document analysis, a significant portion of semantic content can be captured while many of the intractable complexities of language can be ignored. This can result in a large-scale information access system which is capable of processing unrestricted text, largely understanding it, and answering natural language queries robustly and with high precision.

Previous Work: See [1, 3].

Approach: We wish to bridge the gap between natural language and information retrieval by distilling natural language text into a representational structure that is amenable to fast, large-scale indexing. Rather than representing linguistic relationships as syntactic parse trees or semantic case frames, which are much more difficult to index and retrieve efficiently due to their size and complexity, we represent them as ternary expressions. Ternary expressions are two place predicates of the form {subject relation object}, for example {frog eat insect}; they can easily express many types of relations, e.g., subject-verb-object relations, possession relations, etc. For 20 years, they have been successfully used by the START system [1] to store knowledge and answer queries. Ternary expressions fail to capture some subtleties of language, but they represent the basic sense of language, and because they are so simple in form, they can be indexed and retrieved rapidly.

By indexing relations between entities in natural language as ternary expressions, an information access system is able to distinguish the differences in meaning between pairs of sentences and phrases such as these:

(1) The bird ate the young snake.	< bird eat snake >, < young `mod` snake >
(1′) The snake ate the young bird.	< snake eat bird >, < young `mod` bird >
(2) The meaning of life	< meaning `possessive-relation` life >
(2′) A meaningful life	< meaningful `describes` life >
(3) The bank of the river	< bank `possessive-relation` river >
(3′) The bank near the river	< bank `near-relation` river >

To test our ideas, we have built a system, called Sapere, which indexes information using ternary expressions as described above. Once data is stored in the system via indexed ternary expressions, Sapere can accept queries and analyze them into ternary expressions via the same mechanism. If the representation of the user's query matches representations stored in the system, then Sapere retrieves the associated original data and presents it to the user.

Impact: Although Sapere is slower than the simple keyword indexer, we believe that the potential to dramatically increase precision offsets the longer processing time. By using simplified NLP techniques which are rapid yet retain much of the intelligence of full NLP, and applying them to the domain of IR, we should be able to improve on keyword-based IR algorithms without suffering the drawbacks of full NLP systems. As shown in Figure 1, we submitted the query "What do frogs eat?" to Sapere and a standard keyword-based IR system. The standard IR system retrieved 32 answers from an encyclopedia, two of which were correct, while Sapere returned only the two correct answers.

Question: What do frogs eat?

Answer:

Keyword-based IR system:

(R1) Adult frogs eat mainly insects and other small animals, including earthworms, minnows, and spiders.
(R2) Bowfins eat mainly other fish, frogs, and crayfish.
(R3) Most cobras eat many kinds of animals, such as frogs, fishes, birds, and various small mammals.
(R4) Cranes eat a variety of foods, including frogs, fishes, birds, and various small mammals.
(R5) Frogs eat many other animals, including spiders, flies, and worms.
...
(R32) Sometimes the snake eats insects and frogs.

Sapere:

(R1) Adult frogs eat mainly insects and other small animals, including earthworms, minnows, and spiders.
(R5) Frogs eat many other animals, including spiders, flies, and worms.

Figure 1: Example of keyword-based IR vs. Sapere.

Future Work: The current version of Sapere is a prototype; we need to extend the grammar, index large amounts of text, and test a variety of queries to vet the effectiveness of this approach. We would like to explore combined approaches to resolve some of the speed and low recall issues with Sapere; perhaps we could use keyword-based IR as a first pass and Sapere as a second pass.

When making use of linguistic relations in text, it is necessary to detect when structures have similar meanings but differ in form, e.g., "eat" should match "feed on"; "the frog's diet is X" should match "the frog eats X". Otherwise the system's recall will suffer as it fails to match queries to text that was indexed differently. Sapere is capable of equating minor variations in language, such as synonyms, active / passive, etc., but it cannot yet recognize greater variations. Consider the following sentences that express the same meaning using different constructions:

(4) Whose declaration of guilt shocked the country?
(5) Who shocked the country with his declaration of guilt?

Transformational rules [2, 1, 4] provide a mechanism to explicitly equate alternate realizations of the same meaning at the level of ternary expressions. These rules are generally applied to classes of words which act similarly, rather than to individual words.

In order for a question answering system to be successful and have adequate linguistic coverage, it must have a large number of these rules. A lexicon which classified verbs by argument alternation patterns would be a good start, but this is another resource lacking in the world today. Rules generally may be quite complex, and thus an efficient mechanism by which to create these rules is an issue that requires further research.

Research Support: This research is funded by DARPA under contract number F30602-00-1-0545 and administered by the Air Force Research Laboratory.

References:

[1] B. Katz. Using English for Indexing and Retrieving. In P. H. Winston and S. A. Shellard, editors, *Artificial Intelligence at MIT: Expanding Frontiers*, volume 1, Cambridge, MA, 1990. MIT Press.

[2] B. Katz and B. Levin. Exploiting lexical regularities in designing natural language systems. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING '88)*, 1988.

[3] B. Katz and J. Lin. REXTOR: A System for Generating Relations from Natural Language. In *Proceedings of ACL 2000 Workshop on Recent Advances in Natural Language Processing and Information Retrieval*, 2000.

[4] Boris Katz. Annotating the World Wide Web using natural language. In *Proceedings of the 5th RIAO Conference on Computer Assisted Information Searching on the Internet (RIAO '97)*, 1997.