Natural Language Annotations for Question Answering*

Boris Katz, Gary Borchardt and Sue Felshin

Massachusetts Institute of Technology
Computer Science and Artificial Intelligence Laboratory
32 Vassar Street, Cambridge, MA 02139
{boris, borchardt, sfelshin}@csail.mit.edu

Abstract

This paper presents strategies and lessons learned from the use of natural language annotations to facilitate question answering in the START information access system.

1. Introduction

START [Katz, 1997; Katz, 1990] is a publicly-accessible information access system that has been available for use on the Internet since 1993 (http://start.csail.mit.edu/). START answers natural language questions by presenting components of text and multi-media information drawn from a set of information resources that are hosted locally or accessed remotely through the Internet. These resources contain structured, semi-structured and unstructured information.

START targets high precision in its question answering, and in large part, START's ability to respond to questions derives from its use of natural language annotations as a mechanism by which questions are matched to candidate answers. When new information resources are incorporated for use by START, natural language annotations are often composed manually—usually at an abstract level—then associated with various information components. While the START effort has also explored a range of techniques for automatic generation of annotations, this paper focuses on the use of, and benefits derived from, manually-composed annotations within START and its affiliated systems.

2. Background

The START system analyzes English text and produces a *knowledge base* which incorporates, in the form of nested *ternary expressions* [Katz, 1990], the information found in the text. One can think of the resulting entry in the knowledge base as a "digested summary" of the syntactic structure of an English sentence. A user can query the system in English. The query is analyzed in the same way as assertions used to create the knowledge base. The query's analyzed form is matched against the knowledge base to retrieve stored knowledge. The system will then produce an English response. Because matching occurs at the level of syntactic structures, linguistically sophisticated machinery such as synonymy, hyponymy, ontologies, and structural transformation rules can all be brought to bear on the matching process. Linguistic techniques allow the system to achieve capabilities beyond simple keyword matching, for example, handling complex syntactic alternations involving verb arguments.

A representation mimicking the hierarchical organization of natural language syntax has one undesirable consequence: sentences differing in their surface syntax but close in meaning are not considered similar by the system. For this reason, START deploys its structural transformation rules (in both forward and backward modes) which make explicit the relationship between alternate realizations of the arguments of verbs and other structures [Katz and Levin, 1988].

3. Natural Language Annotations

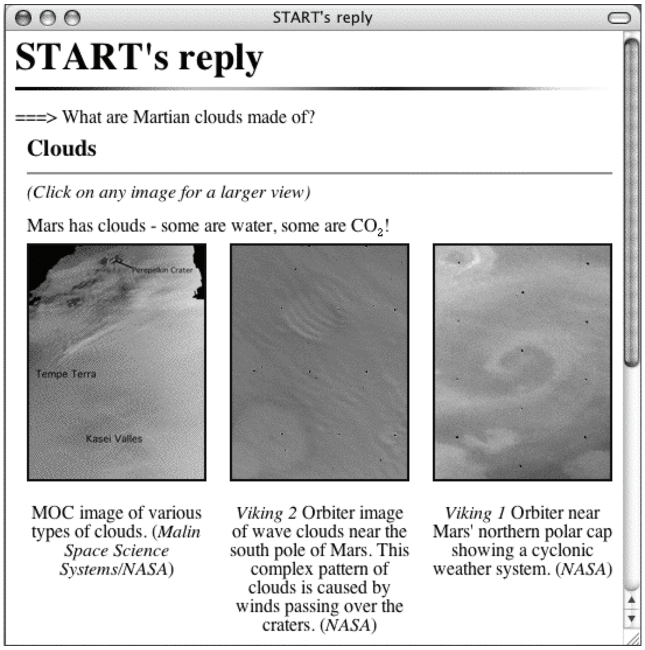
The START system bridges the gap between sentence-level text analysis capabilities and the full complexity of unrestricted natural language (and multimedia information) by employing *natural language annotations* [Katz, 1997]. Annotations are computer-analyzable collections of natural language sentences and phrases that describe the contents of various information segments. START analyzes these annotations in the same fashion as any other sentences, but in addition to creating the required representational structures, the system also produces special pointers from these representational structures to the information segments summarized by the annotations. For example, the HTML fragment about clouds on Mars in Figure 1 may be annotated with the following English sentences and phrases:

```
clouds exist on Mars
Martian clouds are composed of water and carbon dioxide
...
```

*This work is supported in part by the Advanced Research and Development Activity as part of the AQUAINT Phase II research program.

START parses these annotations and stores the parsed structures (nested ternary expressions) with pointers back to the original information segment. To answer a question, the user query is compared against the annotations stored in the knowledge base. If a match is found between ternary expressions derived from annotations and those derived from the query, the segment corresponding to the annotations is returned to the user as the answer. For example, annotations like those above allow START to answer the following questions (see Figure 1 for an example):

Are there clouds on Mars?
What do Martian clouds look like?
What is the chemical composition of Martian clouds?
Do you know what clouds on Mars are made of?



START's reply

START's reply

===> What are Martian clouds made of?

Clouds

(Click on any image for a larger view)

Mars has clouds - some are water, some are CO_2!

MOC image of various types of clouds. (*Malin Space Science Systems/NASA*)

Viking 2 Orbiter image of wave clouds near the south pole of Mars. This complex pattern of clouds is caused by winds passing over the craters. (*NASA*)

Viking 1 Orbiter near Mars' northern polar cap showing a cyclonic weather system. (*NASA*)

Figure 1: START responding to the question "What are Martian clouds made of?" with an information segment containing both text and images.

With large resources, of course, it is impractical to annotate each item of content. However, resources of all types—structured, semi-structured and unstructured—can contain significant amounts of parallel material. *Parameterized annotations* address this situation by combining fixed language elements with "parameters" that specify variable portions of the annotation. As such, they can be used to describe whole classes of content while preserving the indexing power of non-parameterized annotations. As an example, the parameterized annotation (with parameters in italics)

number people live in the metropolitan area of *city*.

can describe, on the data side, a large table of population figures for various cities. On the question side, this annotation, supported by our structural transformation rules, can recognize questions submitted in many forms:

How many people reside in Chicago?
Do many people live in the metropolitan area of Pittsburgh?
What number of people live in Seattle's metropolitan area?
Are there many people in the Boston area?

In combination with other parameterized annotations that describe the population figures in other ways (for example, using the terms "population" or "populous"), and by parameterizing additional elements of annotations, a large number of different questions can be answered using a small number of parameterized annotations.

4. Handling Large Semi-Structured and Structured Resources

Large resources, and particularly those that are semi-structured or structured, typically contain significant amounts of parallel information—similar information about different entities. Parameterized annotations can be used to represent large swaths of content in these resources, enabling a question answering system to easily detect matches between its input questions and information contained in the resources.

As an extensive application of parameterized annotations, our Omnibase system [Katz *et al.*, 2002] supports the START system by retrieving answers from a variety of semi-structured and structured resources using object–property queries generated by START. Omnibase acts as an abstraction layer that provides a uniform interface to disparate Web knowledge sources. Parameterized annotations serve as the interface between START and Omnibase, allowing the combined systems to answer questions about a variety of topics such as almanac information (cities, countries, lakes, etc.; weather, demographics, economics, etc.), facts about people (birth dates, biographies, etc.), and so forth. Some examples of object–property combinations are a country's population, area, GDP or flag; a city's population, location or subway map; a famous individual's place of birth, date of birth, or spouse.

In order to successfully match input questions to parameterized annotations, START must know which terms can be associated with any given parameter. Omnibase supports this need by acting as an external gazetteer for resource-specific terminology, with variants of terms being calculated automatically from objects' names, extracted from semi-structured material in resources, or manually defined.

When a parameter in an annotation matches a term in the user's question, the system "binds" the parameter to the identifier(s) found by Omnibase. These identifiers are then used by Omnibase in its dealings with the associated information resources.

Our IMPACT system (based on [Borchardt, 1992; Borchardt, 1994]) provides an additional example of the use of parameterized annotations to support question answering over semi-structured and structured resources. IMPACT provides access to information in relational databases by associating parameterized annotations with selections of columns from database tables. By matching input questions to the parameterized annotations created for this purpose, START can perform a range of database selection operations through IMPACT. Separately, IMPACT provides a layer of inference rules that can fuse information from multiple sources and perform calculations concerning time and events, and the heads of these inference rules are also associated with parameterized annotations. IMPACT's use of parameterized annotations is described more fully in [Katz et al., 2005].

5. Decomposing Complex Questions

Recently, we have been exploring the answering of complex questions such as "When was the president of France born?" Such questions are interesting because answering them typically involves information from different sources, and indeed, in answering one part of such a question—e.g., "Who is the president of France?", a system needs to identify an answer before proceeding to use that value—in this case, Jacques Chirac—within another subquestion to be answered—e.g., "When was Jacques Chirac born?". Parameterized annotations can help, in that they can be used to describe sets of simple questions that can be answered independently. In addition, the mechanism of parameter matching—via synonyms, hyponyms, etc.—plus the underlying mechanisms that supply answers to the simple questions, can be used to bridge terminology differences between resources, permitting a range of complex questions to be answered.

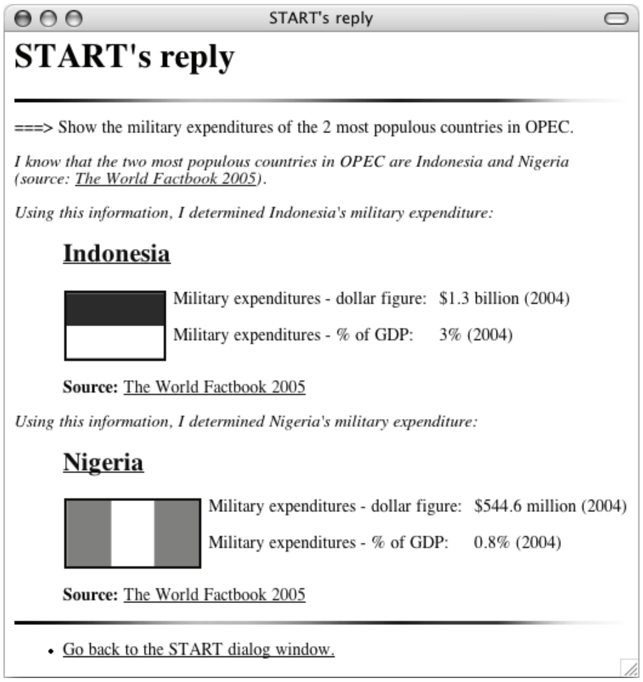
We have been exploring an approach whereby START analyzes complex questions linguistically in order to isolate candidate subquestions, then checks, via its base of annotated resource materials, to see if particular subquestions can be answered. This approach is described more fully in [Katz et al., 2005]. Figure 2 provides an example of START answering a complex question.

6. Conclusions

Natural language annotations, in combination with sentence-level NLP, enable very high precision question answering:

- Ternary expressions can serve as a compact, yet expressive, representational foundation for natural language based matching.
- Natural language annotations make it possible to index text and non-text resources which cannot be analyzed by current systems.
- Parameterized annotations make it practical to index large resources which contain significant amounts of parallel information, using only a small number of annotations.
- Natural language annotations can provide assistance toward the answering of complex questions.

Our ongoing research in the area of annotations focuses on producing annotations semi-automatically, with minimal, data-driven manual guidance based in the lexicon and on information resource categorization.



START's reply

===> Show the military expenditures of the 2 most populous countries in OPEC.

I know that the two most populous countries in OPEC are Indonesia and Nigeria (source: The World Factbook 2005).

Using this information, I determined Indonesia's military expenditure:

Indonesia

Military expenditures - dollar figure: $1.3 billion (2004)

Military expenditures - % of GDP: 3% (2004)

Source: The World Factbook 2005

Using this information, I determined Nigeria's military expenditure:

Nigeria

Military expenditures - dollar figure: $544.6 million (2004)

Military expenditures - % of GDP: 0.8% (2004)

Source: The World Factbook 2005

- Go back to the START dialog window.

Figure 2: START answers a complex question using its syntactic decomposition strategy. START uses its ability to generate language to explain how it derived its answer.

References

Borchardt, G. C., 1992. "Understanding Causal Descriptions of Physical Systems," in *Proceedings of the AAAI Tenth National Conference on Artificial Intelligence*, 2-8.

Borchardt, G. C., 1994. *Thinking between the Lines: Computers and the Comprehension of Causal Descriptions*, MIT Press.

Katz, B. and Levin, B., 1988. "Exploiting Lexical Regularities in Designing Natural Language Systems," in *Proceedings of the 12th International Conference on Computational Linguistics (COLING '88)*

Katz, B., 1990. "Using English for Indexing and Retrieving," in P. H. Winston and S. A. Shellard (eds.), *Artificial Intelligence at MIT: Expanding Frontiers*, volume 1, MIT Press, Cambridge, MA.

Katz, B., 1997. “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).*

Katz, B., Felshin, S., Yuret, D., Ibrahim, A., Lin, J., Marton, G., McFarland, A. J. and Temelkuran, B., 2002. “Omnibase: Uniform Access to Heterogeneous Data as a Component of a Natural Language Question Answering System,” in *Proceedings of the 7th International Workshop on Applications of Natural Language to Information Systems (NLDB 02)*.

Katz, B., Borchardt, G. and Felshin, S., 2005. “Syntactic and Semantic Decomposition Strategies for Question Answering from Multiple Resources,” in *Proceedings of the AAAI Workshop on Inference for Textual Question Answering.* Pittsburgh, PA, 35-41.