Q

Question Answering from Procedural Semantics to Model Discovery

John Kontos
National and Kapodistrian University of Athens, Greece

Ioanna Malagardi
National and Kapodistrian University of Athens, Greece

INTRODUCTION

Question Answering (QA) is one of the branches of Artificial Intelligence (AI) that involves the processing of human language by computer. QA systems accept questions in natural language and generate answers often in natural language. The answers are derived from databases, text collections, and knowledge bases. The main aim of QA systems is to generate a short answer to a question rather than a list of possibly relevant documents. As it becomes more and more difficult to find answers on the World Wide Web (WWW) using standard search engines, the technology of QA systems will become increasingly important. A series of systems that can answer questions from various data or knowledge sources are briefly described. These systems provide a friendly interface to the user of information systems that is particularly important for users who are not computer experts. The line of development of ideas starts with procedural semantics and leads to interfaces that support researchers for the discovery of parameter values of causal models of systems under scientific study. QA systems historically developed roughly during the 1960-1970 decade (Simmons, 1970). A few of the QA systems that were implemented during this decade are:

- The BASEBALL system (Green et al., 1961)
- The FACT RETRIEVAL System (Cooper, 1964)
- The DELFI systems (Kontos & Kossidas, 1971; Kontos & Papakontantinou, 1970)

The BASEBALL System

This system was implemented in the Lincoln Laboratory and was the first QA system reported in the literature according to the references cited in the first book with a collection of AI papers (Feigenbaum & Feldman, 1963). The inputs were questions in English about games played by baseball teams. The system transformed the sentences to a form that permitted search of a systematically organized memory store for the answers. Both the data and the dictionary were list structures, and questions were limited to a single clause.

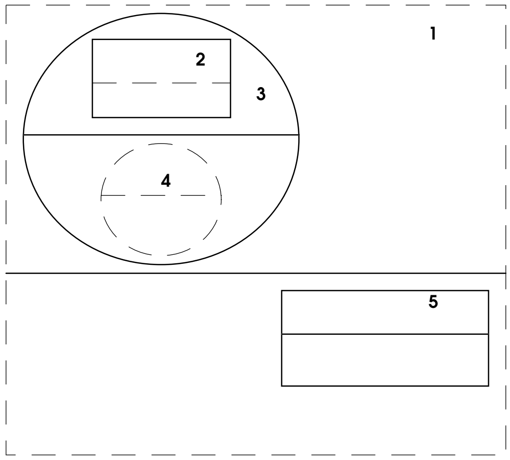
The FACT RETRIEVAL System

The system was implemented using the COMIT compiler-interpreter system as programming language. A translation algorithm was incorporated into the input routines. This algorithm generated the translation of all information sentences and all question sentences into their logical equivalents.

The DELFI System

The DELFI system answers natural language questions about the space relations between a set of objects. These are questions with unlimited nesting of relative clauses that were automatically translated into retrieval procedures consisting of general-purpose procedural components that retrieved information from the database that contained data about the properties of the objects and their space relations. The system was a QA system based on procedural semantics. The following is an example of a question put to the DELFI system:

Figure 1. The objects of the DELFI I example application database



"Is an object that has dotted contour below the object number 2?"

The answer is, "Yes, object no. 4," given the objects with numbers 1, 2, 3, 4, and 5 as shown in Figure 1 (Kontos, 2004; Kontos & Papakonstantinou, 1970).

The DELFI II System

The DELFI II system (Kontos & Kossidas, 1971) was an implementation of the second edition of the system DELFI augmented by deductive capabilities. In this system, the procedural semantics of the questions are expressed using macro-instructions that are submitted to a macro-processor that expands them with a set of macro-definitions into full programs. Every macro-instruction corresponded to a procedural semantic component. In this way, a program was generated that corresponded to the question and could be compiled and executed in order to generate the answer. DELFI II was used in two new applications. These applications concerned the processing of the database of the personnel of an organization and the answering of questions by deduction from a database with airline flight schedules using the following rules:

- If flight F1 flies to city C1, and flight F2 departs from city C1, then F2 follows F1.
- If flight F1 follows flight F2, and the time of departure of F1 is at least two hours later than the time of arrival of F2, then F1 connects with F2.
- If flight F1 connects with flight F2, and F2 departs from city C1, and F1 flies to city C2, then C2 is reachable from C1.

Given a database that contains the following data:

- F1 departs from Athens at 9 and arrives at Rome at 11
- F2 departs from Rome at 14 and arrives at Paris at 15
- F3 departs from Rome at 10 and arrives at London at 12

If the question "Is Paris reachable from Athens?" is submitted to the system, then the answer it gives is *yes*, because F2 follows F1, and the time of departure of F2 is three hours later than the time of arrival of F1. It should be noted also that F1 departs from Athens, and F2 flies to Paris.

If the question "Is London reachable from Athens?" is submitted to the system, then the answer it gives is *no*, because F3 follows F1, but the time of departure of F3 is one hour earlier than the time of arrival of F1. It should be noted here that F1 departs from Athens, and F3 flies to London.

In Figure 2, the relations between the flights and the cities are shown diagrammatically.

Figure 2. The relations between the flights and cities of the DELFI II example application (Kontos, 2003)