

A TECHNIQUE FOR MANAGING THE LEXICON
IN A NATURAL LANGUAGE INTERFACE TO
A CHANGING DATA BASE*

S. Jerrold Kaplan
Computer Science Department
Stanford University
Stanford, California 94305

Eric Mays
Aravind K. Joshi
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, Pennsylvania 19104

A difficulty in designing a Natural Language (NL) interface to a Data Base (DB) management system is insuring that DB updates do not obsolete the NL components. Of particular concern is the lexicon (the list of words that the system can process), mainly because NL queries can contain terms that appear as values in the DB, and hence are subject to change as the contents of the DB changes. For example, to process the question "Does John Jones still work for the company?", it is necessary to identify the string "John Jones" as a (potential) value in the "EMPLOYEE-NAME" field (assuming a suitable DB). If such names must appear in the lexicon for the system to process the query, then the lexicon will go out of date as the company's personnel shifts. Using the DB itself as an extension of the lexicon is equally problematic: the system will be unable to parse and provide a negative answer to such a question if the name does not appear at all in the DB.

The approach suggested here is to infer a plausible field from the context of the query and semantic information about the domain that is implicitly encoded in the structure of the DB. This technique has been implemented in CO-OP, a NL DB query system that provides cooperative responses and operates with a typical CODASYL DB system. CO-OP treats the problem of selecting a plausible field as a special case of resolving semantic ambiguities. Examples drawn from the implementation are presented.

A. Introduction

When designing a Natural Language (NL) interface to a Data Base (DB) that is subject to updates (a "live" DB), it is useful to insure that changes in the contents of the DB do not require separate adjustments in the NL components. This eliminates the need for periodic intervention in the interface to keep it consistent with the DB. The property of remaining up to date on a live DB could be called transparency of DB update.

One component of NL DB query systems that is particularly prone to obsolescence is the lexicon - the vocabulary required by the system to parse and interpret the NL queries. The main reason for this is that the queries can contain terms that appear as values in the DB, and hence are subject to change as the contents of the DB shifts. Consider the question "Has William Smith checked out of room 423?"** To process this question, it is necessary to identify William Smith as a hotel guest (perhaps a potential value of the "GUEST-NAME" field of a record). If such names are explicitly coded in

* This work partially supported by NSF Grant MCS78-08401 and ARPA Contract MDA 903-77-C-0322.

** The examples given in the text are drawn from an imaginary DB about the operation of a hotel. Examples drawn from two real DBs are presented in the conclusion.

in the lexicon, the query system will go rapidly out of date as the clientele of the hotel shifts. Clearly, providing lexical entries for DB values is a poor strategy: not only do DB updates require a corresponding change to the lexicon, but the information that William Smith is a guest is represented twice in the system, raising the possibility of anomalous states (considered as a whole, the system can be said to be improperly normalized).

B. The concept of DB image

An alternative strategy is to use the DB itself as an extension of the lexicon, as in the ROBOT system of (1). Thus, updates will automatically be reflected in the lexicon. A problem with this approach is that extensive searches of the DB may be required to locate the terms not explicitly present in the lexicon* (these will be called unknown terms). A more serious problem is that the values that appear in the DB are only a subset of the unknown terms that may be appropriately incorporated into a query. In the example above, if William Smith has already checked out and paid his bill, his name may no longer appear as a value anywhere in the DB, preventing the system from processing the query. In general, if the DB

*

The ROBOT system circumvents this difficulty by utilizing a fully inverted DB.

is used as an extension of the lexicon, the system will be incapable of responding negatively to queries containing unknown terms. The source of the difficulty is that unknown terms are words or phrases that could appear in the DB, not only words or phrases that do appear. Note that simply keeping a lexicon of "old" (deleted) terms in the DB does not solve this problem - it is possible to pose reasonable queries that contain terms that were never in the DB to begin with. The problem, then, is not to locate them in the DB, but rather to identify the field that could appropriately contain the unknown term, regardless of whether it actually occurs there or not. This field could be called the DB image of the unknown term.

C. Inferring the DB image of unknown terms

One technique for determining the DB image of an unknown term is to rely on the user to provide this information. Although effective, this strategy has several drawbacks. First, the user may not be sufficiently familiar with the logical structure of the DB to select the correct DB image. After all, one of the justifications for using a NL DB query system is that it frees a user from having to know precisely this type of information. Second, the correct DB image may seem obvious to the user, and being asked to provide an unnecessary clarification should be an annoyance ("Is 'William Smith' a GUEST-NAME?").

An alternative approach is to infer a plausible DB image from the context of the unknown term in the query and semantic information about the domain that is implicitly encoded in the structure of the DB (schema). This technique, described below, has been implemented with considerable success in CO-OP, a NL DB query system that provides cooperative responses to NL questions that request the retrieval of data, and operates with a typical CODASYL DB system.(2)

1. Treating unknown terms as semantically ambiguous

In CO-OP, selecting an appropriate DB image for unknown terms is reduced to the problem of selecting a sense (meaning) for each word in the input query. The various senses that a word can take on may be regarded as semantic ambiguities. Associated with each entry in the lexicon for terms that make reference is a set of definitions of the possible senses that term can take on in the domain of the DB. For example, a term like "orders" might have a list of senses designating room service orders, purchase orders, reception desk assignments, etc. Temporary lexical entries are created for unknown terms that designate as their potential senses each field in the DB that could reasonably occur as a DB image in a query. Thus, an unknown term like "William Smith" might

be treated as a lexical item that could have the DB image of either an employee name, a supplier name, a guest name, etc. The word sense disambiguation routines then proceed to select a sense for unknown terms as they do for other terms, as detailed below.

2. Predictive value of selectional restrictions

A particular word sense is selected using two heuristics. The first exploits the predictive value of the words in the immediate syntactic context of the ambiguous term to constrain the set of potential senses. Specifically, verbs and prepositions are marked in the lexicon (when appropriate) to indicate the more likely subjects and objects they can take. If a subject or object has one or more senses that are consistent with these predictions, the alternative senses are eliminated from further consideration. In effect, simple selectional restrictions are used to reduce the semantic ambiguities.

For example, a word like "who" may refer to employees, suppliers, or guests; in the context of the query "Who works in the maintenance department?", the verb "work" may predict employees as a subject, and the possibility of "who" referring to suppliers or guests can be eliminated. Similarly, "William Smith" can be resolved to be a hotel guest, by the predictive value of the verb "checked out". Another potentially useful technique (not implemented) would be to take advantage of verb case roles, or semantic memory structures to help predict categories for the unknown term. As none of these richer knowledge structures were present in CO-OP, no such approach was adopted.

3. Schema distance and semantic relatedness

The above heuristic will not always result in an unambiguous selection. Indeed, a verb like "is" may have no predictive value at all. Remaining ambiguities are resolved by choosing the sense whose DB image results in the shortest distance through the DB schema to the DB image of the surrounding referring terms in the query*. This distance metric is a rough measure of semantic relatedness - semantically related terms tend to have DB images that appear close to each other in the DB schema. Thus "Who is the maintenance department?" may still select employees as the sense of 'who' because the DB image of employees may be closer in the schema to departments than suppliers (perhaps linked through inventory) or guests (perhaps linked through orders or complaints). Similarly, the unknown term "maintenance" (assuming it appears solely as a DB value, if at all) can be correctly identified as a

* -----
A network like structure (such as CODASYL) is required for the operation of this heuristic.

department name, through the close association in the schema of the "DEPT-NAME" field with the DB image of the more specific term "department".

These heuristics, while admittedly rather crude, tend to operate acceptably when the DB schema accurately models the structure of the real world. When CO-OP makes an incorrect interpretation, its mistake is made clear to the user via a paraphrase of the query (3), presented to the user for approval. Experience indicates that it is easy to correct these misinterpretations by rephrasing the query in more specific terms, for example, "Which employees are in the maintenance department?"

D. Examples

Some examples of interpretations of unknown terms actually produced by CO-OP are reproduced below "Q:" (below stands for query, "P:" for paraphrase (responses are deleted in the interest of brevity), commentary is in square brackets []). The first examples are taken from a DB at the National Center for Atmospheric Research in Boulder, Colorado. The DB contains information on their users, divisions, projects, and computer resource utilization.

Q: Which projects in oceanography does NASA Headquarters sponsor?

P: (I am assuming that 'NASA Headquarters' is a SPONSOR NAME.) (I am assuming that 'oceanography' is an AREA OF INTEREST.) Which projects does NASA Headquarters sponsor? Look for projects that are in oceanography.

[This query contained the unknown terms 'oceanography' and 'NASA Headquarters'. The system has correctly identified the first because the preposition 'in' predicts a division, super-division, or area-of-interest as an object in this domain, and area-of-interest is the nearest to projects in the DB schema. 'NASA Headquarters' was identified as a SPONSOR NAME because it is the subject of the verb 'sponsor'.]

Q: Which projects are advised by Thomas Harris?

P: (I am assuming that 'Thomas Harris' is an ADVISOR NAME.) Which projects are advised by Thomas Harris?

[CO-OP is capable of recognizing that 'Thomas Harris' is the logical subject of advise, and hence is probably an advisor, although the passive form is maintained throughout the balance of the processing for other purposes.]

[The following examples are drawn from the ONRODA DB, a DB about military equipment and logistics created as part of the Operational Decision Aids Program of the Office of Naval Research at the University of Pennsylvania*.]

Q: What is the fire rate of the M-61?

P: (I am assuming that 'M-61' is a GUN NAME.) Display the fire rate of the M-61.

[Here, 'M-61' has been identified as a GUN NAME because it is the nearest field in the DB schema to the 'FIRE RATE' field that could reasonably contain an unknown term. The semantic relatedness (schema distance) heuristic is the only one invoked in this case.]

Q: Which F-5s carry strut curve radar?

P: (I am assuming that 'F-5s' is a SHIP NAME.) (I am assuming that 'strut curve' is a RADAR NAME.) Which F-5s carry strut curve radar?

[Because of insufficient contextual cues, CO-OP incorrectly identifies 'F-5s' as a SHIP NAME (it is actually a type of aircraft) - radar can appear on ships, subs, or aircraft in the DB, and the selection defaults to ships. By contrast, 'strut curve' is correctly identified as a type of radar, because it is parsed as modifying 'radar'. The proper interpretation can be induced by re-phrasing the question in a more specific form, as follows.]

Q: Which F-5 aircraft carry strut curve radar?

References

- (1) Harris, L.R., Natural Language Data Base Query Using the Data Base Itself as the Definition of World Knowledge and as an Extension of the Dictionary, Technical Report TR 77-2 Mathematics Dept., Dartmouth College, Hanover, N.H., 1977.
- (2) Kaplan, S. Jerrold, Cooperative Responses from a Portable Natural Language Data Base Query System, Ph.D. dissertation, Department of Computer and Information Science, University of Pennsylvania, Phila., Pa., 1979.
- (3) McKeown, K., "Paraphrasing Using Given and New Information in a Question-Answering System", Association for Computational Linguistics Meeting, San Diego, CA., August 1979.

* Contract N00014-75-C-0440 with the assistance of CTEC, Inc. of McLean, VA.