THE LEXICON IN FCIDB: A FRIENDLY CHINESE INTERFACE FOR DBMS

Da-Jinn Wang, Tsong-Yi Chen and Martha W. Evens

Department of Computer Science and Applied Mathematics
10 West 31st Street, Room 236
Illinois Institute of Technology, Chicago, IL 60616, U.S.A.
EMAIL: {wangdaj, chentso}@harpo.cns.iit.edu, mwe@math.nwu.edu

ABSTRACT

FCIDB (Friendly Chinese Interface for DataBase management systems) can understand users' queries in the Chinese language. It works like a translator that translates Chinese queries into SQL commands. In the translation process, the lexicon of FCIDB plays a key role in both parsing and word segmentation. We designed some questionnaires to collect the frequently occurring words and add them to the public lexicon in FCIDB. FCIDB will produce a private lexicon for every new connected database. This paper will focus on the words included in the public lexicon and in the private lexicon. We also discuss the function, the structure, and the contents of the lexicon in FCIDB.

1. INTRODUCTION

A natural language interface helps users communicate with computers in their own natural language. Natural language interfaces are an application of natural language processing. The goal of a natural language interface to database management systems (DBMSs) is to help users get information from a DBMS without knowing about databases or their contents.

Some natural language interface systems have actually appeared, e.g., LADDER [3], DATALOG [2], and TELI [1]. Today, some domain-independent English interface systems for database are available as commercial products, e.g., English Wizard. English Wizard was developed by the Linguistic Technology Corp. [4]. It allows you to talk to several database management systems, which support Open DataBase Connectivity (ODBC), in the English language. FMDSIC (Friendly Medical Database System Interface in Chinese) is a domain-dependent interface [6]. A domain-independent Chinese interface system, as far as we know, has not appeared before.

FCIDB (Friendly Chinese Interface for DataBase management systems) is a domain-independent interface system. It allows users to communicate with database management systems in Chinese. FCIDB translates Chinese questions into SQL, so that your DBMS receives the SQL it expects.

The architecture of FCIDB includes the processing components: Look-Up & Word Segmentation, Parsing, Translating to SQL Commands, and Dialogue; and the information components: Lexicon,

Chinese grammar rules, SQL grammar rules, and Mapping Dictionary. Figure 1 shows the architecture of FCIDB. The function of the Look-Up and Word Segmentation component is to look up Chinese character strings in a lexicon and determine the word boundary. At the same time, this component obtains information about words from the lexicon, and sends them to the Parser. The Parser applies a set of Chinese grammar rules and a bottom-up parsing algorithm to parse user queries. The Parser must also choose the proper meaning for a word with multiple meanings. During translation the Parser output is reorganized into a legal SQL command. The legal SQL command will be sent to the Dialogue component. Dialogue sends the SQL command to the DBMS and gets the results back. Dialogue must then decide how to phrase the response to the users. If the result is only one value, then the value is included in a sentence. Otherwise, the result is given to the user in table form.

We will focus on the lexicon of FCIDB in this paper. In the following sections, we will talk about the function of the lexicon, the structure of the lexicon, and the contents of the lexicon.

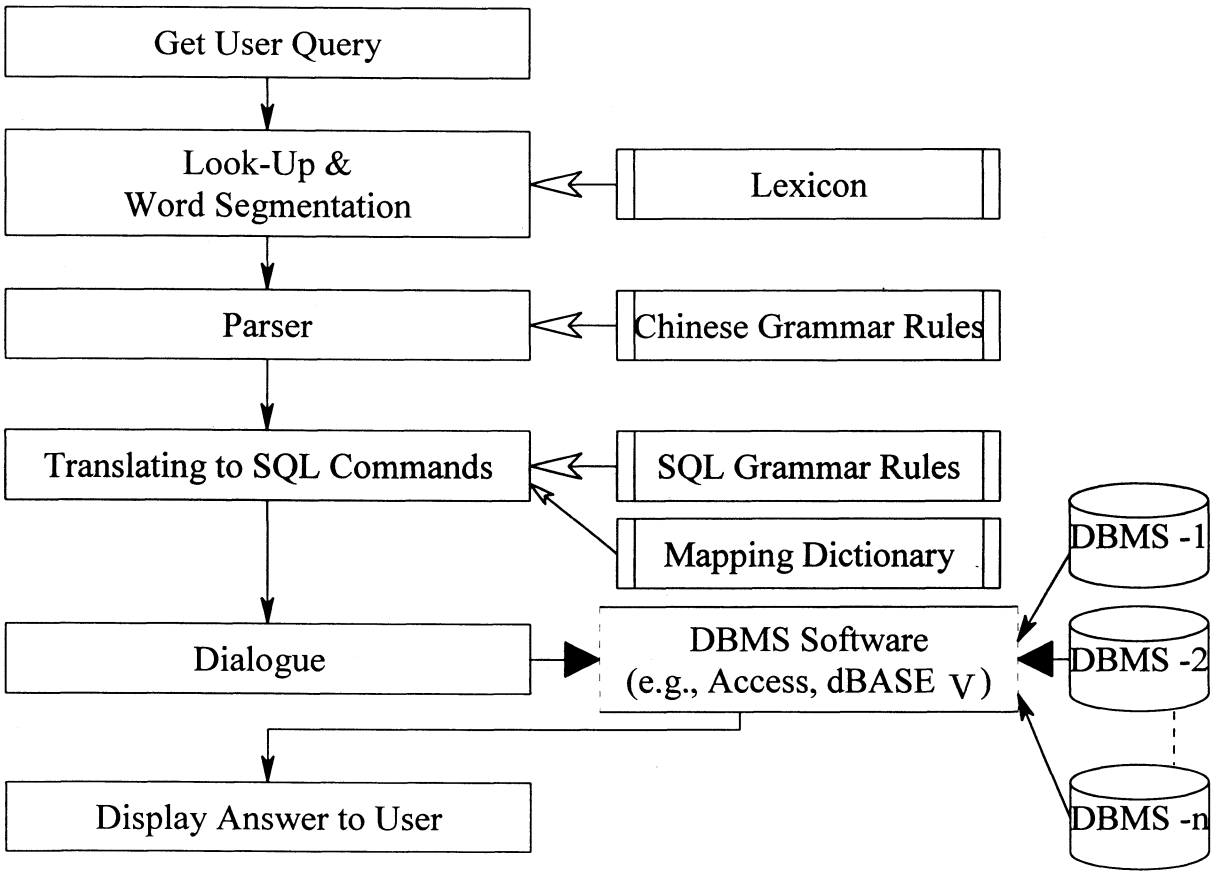

Fig. 1: The Architecture of FCIDB.
(Double bars identify Knowledge Stores)

2. FUNCTION OF THE LEXICON

FCIDB has to recognize every word in the input queries. Our lexicon is a computational dictionary that stores information to help FCIDB understand the meaning of words. FCIDB looks up input query words in the lexicon to retrieve knowledge about those words. The main function of lexicon is to provide this knowledge of words.

The lexicon plays another role in FCIDB to help to implement word segmentation. In the Chinese language, there are no spaces between the words, so we must determine the word boundaries before other natural language processing steps can be taken. The Chinese word segmentation process has been the subject of much intensive research for the past decade. These approaches can be classified into two main groups: 1) the dictionary-based approaches, 2) the statistical approaches [5]. The dictionary-based approaches require a dictionary. They look up character strings in a dictionary and find all words that have the same character sequence as the input sentence. Generally, several different word segmentation strings may be obtained from a Chinese sentence. The maximum-matching (or longest matching) algorithm is often used to select the word sequence that contains the longest words. The dictionary-based approach is applicable to make use for a natural language interface. To implement the longest matching algorithm, words in the lexicon of FCIDB are arranged in descending order.

3. STRUCTURE OF THE LEXICON

The lexicon in FCIDB includes the following information: the word to match, the word category, the equivalent SQL command word, the equivalent SQL command word type, the synonym, and the related field. Why do we need the equivalent SQL command words? The lexicon of FCIDB represents word meanings as SQL command words, as some other systems use a logical form language as a semantic representation.

a. **The Word to Match.** FCIDB matches each word in the input queries against the words in the lexicon.
b. **The Word Category.** Word categories are nouns, verbs, adjectives, adverbs, pronouns, conjunctions, proper nouns, and so on.
c. **The Equivalent SQL Command Words.** Equivalent SQL commands words are SELECT, COUNT, AND, and so on.
d. **The Equivalent SQL Command Word Types.** Equivalent SQL commands word types are <value>, <table_name>, <conditions>, <operator>, and so on.
e. **The Synonym.** For example, "查出" (discover) and "查一下" (check) have the synonym "查詢" (look into).
f. **The Related Field.** The related field is to record the situation where the word may be used. For example, the word "壽星" (a reference to a person on his birthday) may appear in any query to a database that includes the "birthday" column.

4. CONTENTS OF THE LEXICON

To get high-quality results, FCIDB requires a complete dictionary. It is impossible to suppose that a complete Chinese dictionary will be available, because of the enormous size, new words constantly being produced, too many proper nouns, and so on. And we know "the larger the lexicon, the slower the search." Users can not stand waiting for a result for a long time. To minimize the size of the lexicon, we put only the necessary words that may appear in users' queries into the lexicon. Which words should be put into the lexicon? To find out, we have designed some questionnaires to obtain potential queries in the Chinese language.

Using questionnaires to collect queries allowed us to collect a large number of queries in a short time. The disadvantage of questionnaires is that the collected queries do not include all the information available from real conversations. For example, you will not find elliptical queries in the collected queries. In fact, ellipsis often occurs in human dialogue. To make up for this shortcoming, we asked users to work with FCIDB directly once a prototype system was ready.

These questionnaires can be divided into two different fields, one field is about school registrar's affairs and the other one is about warehouse affairs. We collected 1405 queries for the school registrar's affairs, and 945 queries for the warehouse affairs. Then, we calculated the frequency of every word in the collected queries. From this experiment, we found some words appearing in the queries to both fields. Actually, these domain-independent words may appear in the queries no matter what the field is. On the other side, some words, which are domain-dependent, appear in the queries to the special fields.

FCIDB is a domain-independent interface. It means that FCIDB can work with many different databases. To achieve the goal, the lexicon in FCIDB should include all domain-independent words and all domain-dependent words. We call this lexicon as the public lexicon. When FCIDB is connected to a database, it will automatically produce a private lexicon for the specific database. The private lexicon only includes all domain-independent words and related domain-dependent words. The initial private lexicon is a subset of the public lexicon.

4.1 Domain-Independent Words

No matter what kind of databases we have, domain-independent words are likely to appear in queries. The domain-independent words include:

- some imperative verbs: e.g., 查詢 (look for), 列出 (list)
- pronouns: e.g., 我 (I), 你 (you)
- quantifiers: e.g., 全部 (all), 整 (all)
- some common verbs: e.g., 請 (please), 麻煩 (trouble you), 是 (is)
- cardinals and ordinals: e.g., 一 (one), 二 (two)
- prepositions: e.g., 在 (at)
- interrogative pronouns, adverbs: e.g., 什麼 (what), 誰 (who)

· some most frequently occurring vocabulary: e.g., 大於 (greater than), 的 (of)

These domain-independent words are predefined by the system. And every time a new database is built; these words will be added to its private lexicon. Table 1 shows the definition of some domain-independent words in the FCIDB lexicon. The Related Field is not used for domain-independent words.

Table 1: Each Row is a Lexical Entry for a Domain-Independent Word in the FCIDB Lexicon.

Word	Category	SQL Word	SQL Category	Synonym	Related Field
查一查 (look for)	VERB	SELECT	<verb>	查詢	
請 (please)	VERB				
大於 (greater than)	ADJ	>	<operator>		
你 (you)	PRONOUN				

Table 2: Each Row is a Lexical Entry for a Domain-Dependent Word in the Private Lexicon for a School Database

Word	Category	SQL Word	SQL Category	Synonym	Related Field
學生 (student)	NOUN	STUDENT	<table_name>		
老師 (teacher)	NOUN	TEACHER	<table_name>		
課程 (course)	NOUN	COURSE	<table_name>		
修課 (take)	VERB	TAKE	<table_name>		
出生日期 (day of birth)	NOUN	STUDENT.S_BIRTHDAY	<column_names>		
中文系 (Dept. of Literature)	PROPER NOUN	STUDENT.S_DEPT='中文系'	<conditions>		
張小風 (person's name)	PROPER NOUN	STUDENT.S_NAME='張小風'	<conditions>		
B87160045	PROPER NOUN	STUDENT.S_ID='B87160045'	<conditions>		
同學 (classmate)	NOUN			學生 (student)	學生 (student)
生日 (birthday)	NOUN	EXTRACT(MONTH FROM {出生日期}), EXTRACT(DAY FROM {出生日期})	<column_names>		出生日期 (the day of birth)

4.2 Domain-Dependent Words

When FCIDB is connected with a database management system, the words related to the database should be added to the private lexicon. We classified the domain-dependent words as data dictionary words, proper nouns, special field words, and user defined words. In the public lexicon, FCIDB has predefined the category and synonym for every frequently occurring domain-dependent word. The equivalent SQL command words for every Chinese word needs to ask database administrator to help to define them. Table 2 shows the definition of some domain-dependent words in the private lexicon for a school database. The following subsections will talk about every kind of domain-dependent word more detail.

4.2.1 Data Dictionary

FCIDB is a domain-independent interface. It can work with many databases. FCIDB must ask the database administrator to provide Chinese words for every type of information in the data dictionary. These Chinese words represent names of columns and the relationships between tables and between tables and columns in the database. Of course, these Chinese words should be included in the private lexicon. For example, in Table 2, the words 學生 (student), 老師 (teacher), 課程 (course), 修課 (take), and 出生日期 (the day of birth) are data dictionary words.

4.2.2 Proper Nouns in Database

Proper nouns in databases include personal names, company names, id numbers, and so on. Usually, lexicons do not include proper nouns, because there are too many combinations to include all of them. Some lexicons use the pattern to represent some of proper nouns, for example, Ballard [1] represents the office number with "(office ((digit 1) (letter 1) (hyphen 1) (digit 3)) ((digit 1) (letter 1) (hyphen 1) (digit 3) (letter 1)))."

From our collected queries, we find the proper nouns appearing in queries that also appear in the database. We append those proper nouns appearing in database to the private lexicon. This makes word segmentation easier. In Table 2, the words 中文系 (Dept. of Literature), 張小風 (person's name), and B87160045 are proper nouns.

4.2.3 Special Field Words

In addition to data dictionary words and proper nouns, FCIDB also predefines some frequently occurring domain-dependent words. We call these words "special field words." From our experiment, 老師 (teacher) can be a title or a person. It will appear in the queries to the databases about school affairs. The Chinese word 貨物 (goods) will appear in the queries to the business databases. If FCIDB works with a school database as an interface, then the word 貨物 (goods) does not need to be added to the private lexicon.

FCIDB has predefined the category, synonym, and related field for every special field word. It is

very difficult to predefine the equivalent SQL command words for every special field word, because FCIDB can not anticipate the names of columns in databases. We have to ask the database administrator to provide Chinese synonyms for all attributes in a new database. Sometimes synonyms for SQL command words are database dependent as well. FCIDB has defined the equivalent SQL commands for some special field words in a dynamic form. For example, in Table 2, we find the SQL command word of 生日 (birthday) is "EXTRACT (MONTH, {出生日期}), EXTRACT (DAY, {出生日期})". Here {出生日期} (day of birth). Here, { } is a dynamic representation. We know that 生日 (birthday) is related with 出生日期 (day of birth). If 出生日期 (day of birth) is used for defining a column of table in a database, then 生日 (birthday) will be added to the private lexicon for the database. At the same time, the dynamic representation, {出生日期}, will be replaced with the name of the column. For example, 出生日期 (day of birth) is defined as a column name "birth", then the definition of 生日 (birthday) will be modified as "EXTRACT (MONTH, birth), EXTRACT(DAY, birth)". But anyway, the best solution is to ask database administrator to help to define the equivalent Chinese words in the special field for every related SQL command.

4.2.4 User Defined Words

As we mentioned earlier in this paper, it is impossible for a lexicon to include all words. Users can define missing words themselves. If the predefined word in the FCIDB lexicon does not tally with yours, you can also redefine the words by yourself. FCIDB will replace its old definition with the user's new definition.

Table 3: Each Row is a Lexical Entry for a User Defined Word in the Private Lexicon for a Warehouse Database.

Word	Category	SQL Word	SQL Category	Synonym	Related Field
規格 (specification of a manufactured item)	NOUN	GOODS.G_SIZE, GOODS.G_COLOR, GOODS.G_WEIGHT	<column_names>		
庫存清單 (a list of items which are stored in ware-house)	NOUN	GOODS.G_ID, GOODS.G_DES, GOODS.G_QTY	<column_names>		
存貨清單 (a list of items which are stored in ware-house)	NOUN			庫存清單	

New words are added to the private lexicon, but not to the public lexicon. So, users can give words (even words predefined by FCIDB) different definitions in different databases. For example, "退休" (retired) can be defined by its equivalent SQL command words "DateDiff("yyyy", [Birthday], SYSDATE)>65" in the private lexicon for database A; the same word "退休" (retired) can be defined "DateDiff("yyyy", [BIRTHDAY], SYSDATE)>60" in the private lexicon for database B. The advantage is that users can define every word.

5. CONCLUSION

A lexicon is the foundation of a natural language interface. The lexicon is essential to word segmentation and query translation.

We carried out an experiment to explore the lexicon. We constructed two different databases and designed questionnaires to collect queries. The results helped us to identify which words we needed in the public and private lexicon.

We still need to simplify the word definition process to make it easier for users to add terminology and to move from one database to another. Now, the system can be an interface with ACCESS and Visual dBASE. In the future, we hope to port it to other systems.

6. REFERENCES

[1] Bruce W. Ballard, "A Lexical, Syntactic, and Semantic Framework for TELI: a User Customized Natural Language Processor," In: Relational Models of the Lexicon, Edited by M. W. Evens, Cambridge University Press, Cambridge, UK, 211236, 1988.

[2] Carole D. Hafner and Kurt S. Godden, Design of Natural Language Interfaces: A Case Study; Research Publication GMR-4567, Computer Science Department, General Motors Research Laboratories, 1984.

[3] Gary G. Hendrix, Earl D. Sacerdoti, Deniel Sagalowicz, and Jonathan Slocum, "Developing a Natural Language Interface to Complex Data"; ACM Transactions on Database Systems, Vol. 3, 105-147, 1978.

[4] Linguistic Technology Corp., English Wizard Version 1.0; Acton, MA, in disk, 1995.

[5] Jian-Yun Nie, Xiaobo Ren, and Martin Brisebois, "A Unifying Approach to Segmentation of Chinese and its Application to Text Retrieval"; Proceedings of R.O.C. Computational Linguistics Conference Ⅷ, 175-190, 1995.

[6] Da-Jinn Wang, Tsong-Yi Chen, and Martha Evens, "Friendly Medical Database System Interface in Chinese"; Proceedings of MAICS96, Bloomington, IN. http://www.cs.indiana.edu/event/maics96/Proceedings/Wang/wang.ps, 1996.