Prepositions and Conjunctions in a Natural Language Interfaces to Databases*

J. Javier González B.[1], Rodolfo A. Pazos R.[2], Alexander Gelbukh[3], Grigori Sidorov[3], Hector Fraire H.[1], and I. Cristina Cruz C.[1]

[1] Instituto Tecnológico de Ciudad Madero, México
jjgonzalezbarbosa@hotmail.com,
hfraire@prodigy.net.mx, ircriscc@hotmail.com
[2] Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET), México
pazos@sd-cenidet.com.mx
[3] Centro de Investigación en Computación
{gelbukh,sidorov}@cic.ipn.mx

Abstract. This paper present the treatment of prepositions and conjunctions in natural language interfaces to databases (NLIDB) that allows better translation of queries expressed in natural language into formal languages. Prepositions and conjunctions weren't sufficiently studied for their usage in NLIDBs, because most of the NLIDBs just look for keywords in the sentences and focus their analysis on nouns and verbs getting rid of auxiliary words in the query. This paper shows that prepositions and conjunctions can be represented as operations using formal set theory. Additionally, since prepositions and conjunctions keep their meaning in any context, their treatment is domain independent. In our experiments we used Spanish language. We validate our approach using two databases; Northwind and Pubs of SQL Server, with a corpus of 198 different queries for the first one and 70 queries for the second one. The 84% of queries were translated correctly for the database Northwind and 80% for Pubs.

1 Introduction

The present situation with natural language interfaces to databases (NLIDBs) is such that they do not guarantee in general satisfactory translation of natural language queries into formal language representation [1]. Additionally, most NLIDBs are limited to certain database domains and they are usually configured manually by the DB administrators.

A survey of existing NLIDB architectures shows that most of them have the following characteristics [2]: a) Each NLIDB carries out a transformation from natural language into an intermediate representation language, from which a query is generated for obtaining results from the repository, b) they have an inherent dependency on the domain of the database information, c) NLIDBs implementations are modular and robust, but the queries that they can answer are

* This research was supported in part by CONACYT and DGEST.

P. Thulasiraman et al. (Eds.): ISPA 2007 Workshops, LNCS 4743, pp. 173–182, 2007.

limited by the vocabulary and grammar defined for the NLIDB and d) NLIDBs were designed for obtaining answers for specific domains instead of trying to make them domain independent.

Additionally, most NLIDBs focus on the analysis of the sentence structure and only few concentrate on the meaning of the constituent elements or use discourse handling techniques. Still, practically all of them use some limited grammar for detection of syntactic relations between words.

Each phase of analysis supplies important information for the query translation process, but the largest effort has been focused on the morphological and syntactic analyses; however, exact understanding of users queries is far from being achieved by NLIDBs.

2 Related Work

In some NLIDBs, the semantic analyzer extracts the meaning of the sentence and generates a logical structure from the syntactic structure obtained using a syntactic parser [3]. One of the key tasks of the semantic processing consists in determining, which combinations of individual word meanings are possible while generating a coherent meaning of the sentence. This approach can be used for reducing the number of possible meanings for each word of the given sentence [4].

The semantic analysis of sentences is still a very complex task [5], just the determination of the meaning of words is a difficult task due to their polysemy; for example, file is a tool and also a place for keeping documents, etc.

In many NLIDBs, the semantic analysis of a natural language query involves searching of keywords in the input sentence, which are evaluated according to a predefined pattern through multiple database mappings. In some research works, the semantic analysis uses probabilistic models [6, 7], which need a corpus labeled with semantic information. This is a subjective approach and it requires a considerable manual effort, besides, these models were applied to specific tasks within a restricted semantic domain and use a semantic representation that usually consists of case-frames [8].

For carrying out the semantic analysis of a sentence, the computer needs mainly a structure for storing the meanings and relationships of each word, a dictionary that holds the general language knowledge (words, meanings, relationships, synonyms, antonyms, etc.), and algorithms and techniques for obtaining the relevant information for carrying out the translation into a formal language regardless of domain.

Prepositions and conjunctions were not sufficiently studied for tasks of processing natural language queries, because most of the NLIDBs just look for keywords in the sentence [13] and focus their analysis on nouns and verbs getting rid of auxiliary words in the query [14, 15].

3 Proposed Approach

The solution of the problem of obtaining an exact understanding of a users query can benefit from information regarding invariant parts of the sentence,

like prepositions and conjunctions, which can be exploited for facilitating the query translation process.

In this project will be utilized the preposition de and the conjunction y, by having a very high frequency of use in the Spanish language[17].

The purpose of the proposed approach is designing a technique that permits better translation of a natural language query into Structured Query Language (SQL) and that requires minimum configuration effort for operating with different domains.

The proposed general architecture of the system is shown in Figure 1, and a short description of the constituent modules and their contribution to the translation module is given below.

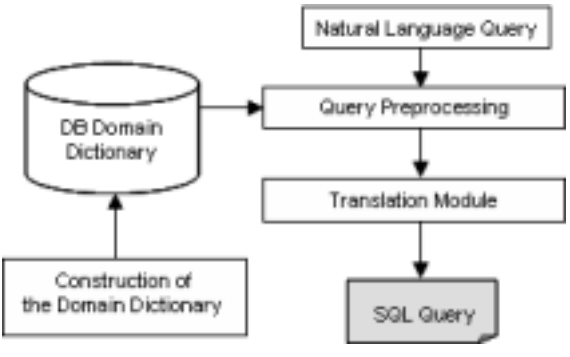

Fig. 1. General architecture of the system

Query Preprocessing: The preprocessor analyzes each word of the sentence in order to obtain its lexical, syntactic, and semantic information. The built-in parser extracts the lexical and syntactic information, whereas the semantic information can be extracted only by interacting with the domain dictionary.

The output of this module consists of the query labeled as shown in Table 1. The query is divided into words that are the minimal meaningful units of the sentence, and for each word information of the following types is included: **lexical** (word stems, synonyms and antonyms), **morphosyntactic**(grammatical category according to its function in the sentence) and **semantic** (meaning of the word with respect to the database). Table 1 shows an example of this information for a query.

Translation Module: This module receives the labeled sentence and processes it in three phases.

Phase 1: Identification of the select and where phrases. The query phrases that define the SQL select and where clauses are identified in order to pinpoint the columns (and tables) referred to by these phrases. Since these clauses always involve table columns, then, we assume that the phrases are query subphrases that include at least one noun (and possibly prepositions, conjunction, articles, adjectives, etc.) and that the phrase that defines the *select* clause always precedes

Table 1. Query Information

QUERY: Muestra los nombres de los empleados. (Show the names of the employees)				
Word	Stem	Morphosyntactic information	Columns	Table
muestra (show)	mostrar (show)	verb, imperative		
los (the)	el (the)	plural, male, determinative		
nombres (names)	nombre (name)	plural, male, noun	cat.catN, cust.comN, emp.fNom, ord.shNom	
de (of)	de (of)	preposition		
los (the)	el (the)	plural, male, determinative		
empleados (employees)	empleado (employee)	plural, male, noun	emp.empID ord.empID	Emp

the phrase that defines the *where* clause. In Spanish, the words that separate these phrases are: verbs, *cuyo* (whose), *que* (that), *con* (with) *de* (from, with), *donde* (where), *en* (in, on, at), *dentro de* (inside), *tal que* (such that), etc.

Phase 2: Identification of tables and columns. Usually each noun in the *select/where* phrases refers to several database columns or tables (see Table 1), which would yield several translations of the query. Therefore, in order to pinpoint the columns and tables referred to, it is usually necessary to analyze the preposition *de* (of) and conjunction *y* (and), since they almost always appear in *select/where* phrases expressed in Spanish [17]. Examination of prepositions and conjunctions permits, besides considering the meaning of individual nouns, to determine the precise meaning of a *select/where* phrase that involves nouns related by prepositions and conjunctions. For this, preposition *de* (of) and conjunction *y* (and) are represented by operations using set theory, because of the role they play in queries.

Preposition de (of) establishes a close relationship between a word and its complement, such that, if there exists a select/where phrase that includes two nouns p and q related by preposition de (of), then the phrase refers to the common elements (columns or tables) referred to by p and q. Formally, S(p prep_de q) = S(p) ∩ S(q), where S(x) is the set of columns or tables referred to by phrase x. Conjunction y (and) expresses the notion of addition or accumulation, such that if there is a select phrase that involves two nouns p and q related by conjunction y (and), then the phrase refers to all the elements referred to by p and$q$. Formally, S(p conj_y q) = S(p) ∪ (q). Conjunction y (and) in a where phrase is treated as a Boolean operation.

For example, consider the query: *cuáles son los nombres y direcciones de los empleados* (which are the names and addresses of the employees). Consider the *select* phrase *nombres y direcciones de los empleados* (names and addresses of the employees). According to the above explanation, to extract the meaning of the

select phrase it is necessary to apply two set operations: a union, corresponding to the conjunction *y* (and), and an intersection, corresponding to the preposition *de* (of). A heuristics is applied to determine the order of the two operations. In this case the preposition *de* (of) applies to the two nouns (*names and addresses of the employees* = *names of the employees and addresses of the employees*), therefore, the intersection operation has precedence above the union. The output of Phase 2 is the semantic interpretation of the *select* and *where* phrases (i.e. the columns and tables referred to by these phrases), which will be used in Phase 3 to translate them into the *select* and *where* clauses of the SQL statement.

The translation module has a tree structure that represents the database information, and additionally the relationships among database columns are included. The columns, tables and search conditions obtained in this phase are marked in the tree, and from this structure a graph is constructed that represents the user's query.

Phase 3: Construction of the relational graph. Once the graph has been constructed, an equivalent SQL expression is generated.

4 Invariant Parts of Sentences: Prepositions and Conjunctions

A sentence is a word or set of words that bears a complete meaning. Some of these words may remain unchanged in all situations; i.e., they do not change, for example, according to gender or number independently of the rest of the sentence (invariant sentence parts); while other words may change according to, say, gender and number (variable sentence parts) [9].

Usually, the invariant sentence parts are auxiliary words like prepositions and conjunctions. These categories are known also as relational elements because they are used for relating some elements to the other. A preposition links a main word with its complement, it links and subordinates simultaneously, while a conjunction links words or syntagms of the same category.

A careful examination of the labeled query reveals that each noun is related to one or more columns or tables (Table 1), to which it may refer, while prepositions and conjunctions express relationships among them. The rest of this section is devoted to demonstrating how prepositions and conjunctions can be represented as operations using set theory. This can be used for facilitating the query translation process.

4.1 Prepositions

Preposition, as mentioned previously, is an invariant that is used for linking a main word (syntactic core) with its complement (glass *of* wine, I am going *to* Rome). This complement is called *preposition term* because the relationship established by the preposition stops and is consummated at this point [10, 11].

Preposition *de* (*of*) establishes a close link between a word and its complement in such a way that the expression constituted by any word or noun *p*

and its complement q linked by preposition *de* refers to the common elements represented by p and q.

$$p \ prep\_de \ q = R \ R = p \cap q.$$

For example, let us consider the following expression: *fecha de nacimiento* (*date of birth*).

Let p and q denote the set of columns and tables that are referred to by the nouns *fecha* (*date*) and *nacimiento* (*birth*). If we want to obtain the set represented by *fecha de nacimiento (date of birth)*, or equivalently, p *prep_de* q, then the common elements of p and q must be obtained.

fecha={c_{fecha}, t_{fecha}}
c_{fecha}={employees.birthDate, employees.hireDate,
orders.orderDate, orders.requiredDate, orders.shipDate}
t_{fecha}={∅}
nacimiento={$c_{nacimiento}$, $t_{nacimiento}$}
$c_{nacimiento}$={employees.birthDate}
$t_{nacimiento}$={∅}

where the t's stay for database tables and the c's denote table columns. Therefore, the information referred to by *fecha de nacimiento* is represented by

$$\text{fecha} \cap \text{nacimiento} = \{\text{employees.birthDate}\}$$

In the previous example none of the nouns (*fecha*, *nacimiento*) refers to a table, and consequently the intersection operates only on the columns. Now let us consider an example that illustrates a different situation: *direcciones de empleados* (*addresses of employees*)

$$\begin{aligned}
&direcciones = \{c_{direcciones}, t_{direcciones}\}\\
&c_{direcciones} = \{customers.address, employees.address, orders.shipAddress,\\
&\qquad suppliers.address\}\\
&t_{direcciones} = \{\emptyset\}\\
&empleados = \{c_{empleados}, t_{empleados}\}\\
&c_{empleados} = \{employees.employeeID, employeesTerritories.employeeID,\\
&\qquad orders.empID\}\\
&t_{empleados} = \{employees\}
\end{aligned}$$

then

$$\begin{aligned}
direcciones \cap empleados &= \{c_{direcciones} \cap c_{empleados}\} \cup \{c_{direcciones} \cap t_{empleados}\}\\
&\quad \cup \{t_{direcciones} \cap c_{empleados}\}\\
&= \{\emptyset\} \cup \{employees.address\} \cup \{\emptyset\}\\
&= \{employees.address\}
\end{aligned}$$

Preposition *de* generally implies the intersection between the columns referred to by the two words, the columns referred to by one word and the tables referred to by the other word, and vice versa.

$$p \cap q = \{\{c_p \cap c_q\} \cup \{c_p \cap t_q\} \cup \{t_p \cap c_q\} \cup \{t_p \cap t_q\}\}$$

This type of operation is valid only for nouns that represent DB columns or tables. When the preposition establishes a link between a set of columns and a value, this case is treated differently. If preposition *de* operates on a column and a value, it will be considered as a comparison operator; for example: *salario de 20,000* (salary equal to 20,000).

Prepositions *con* (*with*) and *sin* (*without*) establish search conditions, and preposition *entre* (between) permits to establish a value range for a column; for example: *salario entre 15,000 y 25,000* (*salary between 15,000 and 25,000*).

4.2 Conjunctions

Conjunction has been traditionally defined as the sentence part that is used for linking two or more elements in an equality relationship [12].

Copulative conjunctions express the notion of addition or accumulation, such that the set of any two nouns p_1 and p_2 that are linked through a conjunction y (*and*) represents the set of all the elements represented by p_1 and p_2

$$p_1 \ conj\_y \ p_2 = R \quad R = p_1 \cup p_2.$$

When a query requires two or more operations of union or intersection (for example, *Muéstrame la fecha de nacimiento y el nombre del empleado*), it is nedeed to determinate which would be the priority of the operations. Thus, in order to evaluate prepositions and conjunctions, numeric values are asigned as follows:

Nomenclature: [n1 prep/conj n2]

- n1: a numeric value of the noun before to the preposition or conjunction (1 for a column, 2 for tables)
- prep/conj: a preposition de, conjunction y.
- n2: a numeric value of the noun after the preposition or conjunction (1 for a column, 2 for tables)

Cases

1. If a pattern [2 prep 2] is identified, it is changed to [1 prep 2] in the case that the first noun is a column that belongs to the table of the second noun.
2. A numeric value of 1 is given to the prepositions in the patterns [1 prep 1] and [2 prep 2].
3. A numeric value of 2 is given to the conjunctions in the patterns [1 conj 1] and [2 conj 2].
4. A numeric value of 3 is given to the prepositions in the pattern [1 prep 2].
5. A numeric value of 4 is in other case.

Now, the prepositions and conjunctions are processed in the order of their numeric value.Those prepositions or conjunctions with a numeric value of 1 are processed first, then those with a numeric value of 2 and so on.

In the example *Muéstrame la fecha de nacimiento y el nombre del empleado (Show me the birthday and name of the employees)*, a set of columns and tables are determined whit the treatment of prepositions and conjunctions. This query includes the cases 2 [1 prep 1], 3 [1 conj 1] y 4 [1 prep 2].

Query	fecha	de	nacimiento	y	nombre	de	empleado
Values	1	1	1	2	1	3	2
Priority		1		2		3	

The union between the nouns *fecha* and *nacimiento* are performed first. The last preposition *de* affects the elements: *fecha de nacimiento y nombre de empleados* = *fecha de nacimiento de empleados y nombre de empleados* (*birth day and name of employees* = *birthday of employees and names of employees*), and therefore the union must be performed before this intersection.

5 Experimental Results

For the experiment, the Northwind and Pubs databases of SQL Server 7.0 were used, and a group of 50 users was gathered in order to formulate queries in Spanish, collecting a corpus consisting of 198 different queries for the Northwind database and 70 queries for the Pubs database. For formulating their queries the users only were allowed to see the databases schemas (definitions).

The queries were classified in the following types according to the kind of information the users express: a) Queries with explicit attributes and relationships, b) Queries with implicit attributes and explicit relationships, c) Queries with explicit attributes and implicit relationships, d) Queries with implicit attributes and relationships, e) Queries that require special functions (average, sum, etc.) and f) Queries that need to be reformulated due to insufficient information to answer them. As far as translated queries are concerned, the following results were obtained: 84% of queries were translated correctly to Northwind Database (See Table 2) and 80% of queries were translated correctly to Pubs database (Table 3). The other percent had errors in translation. There exist two basic reasons that caused these errors: special functions or deduction processes are needed, and lack of explicit information for processing.

Additional experiments were conducted in order to assess the impact of the analysis of prepositions and conjunctions on the translation success. When this analysis was excluded from the translation process, most of the queries were answered with correct information; however, extra columns were obtained. This

Table 2. Results obtained for the Northwind database

Query Results	**Query Type**						**Total**	**%**	**%**
	1	**2**	**3**	**4**	**5**	**6**			
Answered correctly	31	57	19	49	0	0	156	79	**84**
Answered with additional information	0	0	5	5	0	0	10	5	
Incorrect answer	0	0	0	1	23	5	29	15	16
Unanswered	0	0	0	3	0	0	3	1	
Total	31	57	24	58	23	5	198	100	100

Table 3. Results obtained for the Pubs database

Query Results	Query Type						Total	%	%
	1	2	3	4	5	6			
Answered correctly	7	29	8	12	0	0	56	80	**80**
Answered with additional information	0	0	0	0	0	0	0	0	
Incorrect answer	0	0	0	1	10	1	12	17	20
Unanswered	0	0	0	0	1	1	2	3	
Total	7	29	8	13	11	2	70	100	100

show that the treatment of prepositions and conjunctions helps to the semantic analysis of the translation process to get correct results.

6 Conclusions

The elements that express syntactic relations like prepositions and conjunctions are invariant parts of sentences and provide information that is important for the query translation process. These relational elements link or subordinate word categories refining their meaning, thus enhancing the information discrimination. Additionally, since prepositions and conjunctions keep their meaning in all situations, their treatment favors domain independence.

Preposition *de* (*of*) and conjunction *y* (*and*) were studied, as well as their relationship to nouns. We are currently working on the analysis of the meaning of other invariant sentence parts and their relationship to the different syntactic categories aiming at using them in the semantic analysis during query processing.

The experiments conducted include simple and compound sentences that have a similar sentence structure. The experimental results show that it is possible to treat invariant parts of a user's query through the usage of set theory.

References

1. Popescu, A.M., Etzioni, O., Kautz, H.: Towards a Theory of Natural Language Interfaces to Databases. In: Proceedings of the 2003 International Conference on Intelligent User Interfaces, ACM Press, New York (2003)
2. Orea, M.Q.: Interfaz en Espanol para Recuperacion de Informacion en una Base de Datos Geografica. B.S. thesis. Universidad de las Americas. Puebla (2001)
3. AVENTINUS - Advanced Information System for Multinational Drug Enforcement, `http://www.dcs.shef.ac.uk/nlp/funded/aventinus.html`
4. Areas de Investigacion; Procesamiento de Lenguaje Natural. (1998), `http://gplsi.dlsi.ua.es/gplsi/areas.htm`
5. Sidorov, G.: Problemas actuales de Lingüistica Computacional. Revista Digital Universitaria 2(1) (2001), `http://www.revista.unam.mx/vol.2/num1/art1/`
6. Stallard, M.S., Bobrow, D., Schwartz, R.: A Fully Statistical Approach to Natural Language Interfaces. In: Proc. 34^{th} Annual Meeting of the Association for Computational Linguistics (1996)

7. Minker, W.: Stochastically-Based Natural Language Understanding Across Task and Languages. In: Proc. of EuroSpeech97, Rodas, Greece (1997), http://citeseer.nj.nec.com/
8. Moreno, L., Molina, A.: Preliminares y Tendencias en el Procesamiento del Lenguaje Natural, Departamento de Sistemas Informaticos y Computacion. Universidad Politecnica de Valencia.
9. Profesor en Linea. Gramatica y Ortografia: http://www.profesorenlinea.cl/castellano/oracionpartesdela.htm
10. Enciclopedia Libre. Enciclopedia Libre Universal en Espanol (2004), http://enciclopedia.us.es/wiki.phtml
11. Prytz, O.: Notas sobre las Preposiciones Simples en Espanol Moderno (1994), http://www.digbib.uio.no/roman/Art/Rf1-94-1/Prytzb.pdf
12. Martin, V.G.: Curso de Redaccion. In: Martin, V. (ed.) Teoria y Practica de la Composicion del Estilo. Editorial Thomson. 33^a edicion, Madrid, España (2002)
13. Martin, A.: Una Propuesta de Codificacion Morfosintactica para Corpus de Referencia en Lengua Espanola. In: Martin, A. (ed.) Estudios de Lingüistica Espanola (ELiEs). Publicacion periodica de monografias sobre lingüistica espanola (1999), http://elies.rediris.es/elies3/cap31a.htm
14. Bridge, G., Harlow, S.: An Introduction to Computational Linguistics, Intelligent System Group. Department of Computer Science. University of York.
15. Meng, F., Chu, W.W.: Database Query Formation from Natural Language Using Semantic Modeling and Statistical Keyword Meaning Disambiguation. Computer Science Department. University of California
16. InBase-Online. English queries to personnel DB. Russian Research Institute of Artificial Intelligence (2001), http://www.inbase.artint.ru/nl/kadry-eng.asp
17. Montero, J.M.: Sistemas de conversion texto voz. B.S.thesis. Universidad Politecnica de Madrid. http://lorien.die.upm.es/~juancho