Corpus Based Context Free Grammar Design for Natural Language Interface to Database

Avinash J. Agrawal and Dr. O. G. Kakde

**Shri Ramdeobaba Kamla Nehru Engineering College,
Katol Road, Nagpur, India**

**Vishvesvarya National Institute of Technology
Bajaj Nagar, Nagpur, India**

Abstract

For practical implementation of Natural Language Interface to Database, deep semantic analysis needs to be used as it increases success rate and portability. Deep analysis uses more language knowledge rather than the domain knowledge. A primary step in deep semantic analysis is formalizing syntax of possible input questions. This paper describes a context free grammar designed for a domain specific natural language interface to database. The grammar is designed for the domain Railway Inquiry for which corpus of question is collected and analyzed.

Keywords: *Natural Language Interface to Database, Corpus of Questions, Context Free Grammar, Deep Semantic Analysis.*

1. Introduction

Natural language Interface to Database (NLIDB) can acts as an alternative interface for finding structured information from database particularly on a small handheld devices because writing questions in natural language is much easier for a casual user than the complicated and time consuming navigation required in the traditional database interfaces. NLIDB shifts a user's burden of learning use of interface to describe his or her need for information to the system. NLIDB thus demands less input-output and processing facilities which make it more useful for mobile devices [1].

Natural language interface to database is not a new area, a lot of research is going on since a long time [4]. Most of the NLIDB systems developed so far used shallow analysis [11],[12,[13]. Shallow analysis uses domain knowledge rather than linguistic knowledge to interpret the meaning of input question, which results in a low success rate. This is a main difficulty in implementing NLIDB for any practical use. To increase the success rate deep analysis of input question can be used. Deep analysis uses linguistic knowledge for detail understanding. Deep analysis has an advantage of portability that is not possible in shallow analysis due to too much dependency on database.

To explore deep analysis for NLIDB a railway inquiry is selected as a domain. For railway inquiry domain a corpus of question is collected by conducting a survey with different group of railway inquiry users. This corpus is then analyzed to find the pattern of questions used in the domain. Based on these patterns a context free grammar is designed to represent syntax of input questions. Representing syntax is an important step as in deep analysis method the subsequent steps are dependent on it.

2. Corpus Design

2.1 Domain Selection

Different restricted domains benefit from different Question Answering techniques. Some domains are particularly appropriate for the development of question answering systems. Not all domains are appropriate for natural language interface to database. For a domain specific question answering restricted domain must be circumscribed, complex and practical [12].

Circumscribed means the domain where user knows what to expect from the system and knows what questions are appropriate to the domain. A more important motivation for a circumscribed domain is the need for clearly defined knowledge sources.

A domain should be complex enough to warrant the use of a QA system. There is no need for a QA system in a

domain where a simple list of facts or a FAQ would be sufficient to satisfy the user's need for information.

Practicality is an important condition to consider when developing a QA system. The domain should be of use to a relatively large group of people. Otherwise one risks wasting effort on a system that nobody would use.

The selected domain railway inquiry system satisfies all the three requirements. It is circumscribed in a sense that user very well know what types of questions are to be asked in railway inquiry system. Authoritative and comprehensive resources containing required information are already maintained by the railway in the form of database. Thus in railway domain knowledge sources is clearly defined. To answer the question of typical railways inquiry use of extensive knowledge from outside the domain is not required.

Although the railways inquiry domain is circumscribed a QA system will prove to be very useful for the users. It is complex enough domain as simply list of facts or frequently asked questions could not satisfy the wide variety questions with different argument for different groups of people. Especially in countries like India where the network of railways is huge, a railways inquiry may include question related to availability of seats, trains, information regarding stoppage, fare and many information related to thousands of routes and trains. This domain is certainly very interesting for researchers and helpful for users.

It is very much practical domain as there is no risk of wasting of efforts. This type of system is very much required and in demand by people due to difficulty in getting information from the conventional ways. Specially in countries like India, China where railways is a part of life and is a very important medium of transportation and millions of people travel daily by rails and needs some or other kind of information related to the this domain.

2.2 Corpus Collection

More than 150 questions of railway inquiry domain are collected. This question set contains queries related to only general inquiry. Query related to reservation is not included in it. Discussing various people of different age, gender, profession and background does this question collection. The questions set contains queries related to different attribute of railways like arrival departure time, availability of trains between stations, fare, status, concessions etc. Same information may be asked in more than one way in any natural language and example of this is available in the question set. Duplicate questions are removed from the corpus.

From the collected corpus a set of questions is separated for testing purpose. The test corpus contains 24 questions. The test corpus is generated on the basis of the structure of parse tree. Each question of the test corpus is having a distinguished parse tree structure. Thus the test corpus represents the different syntactic structure of questions related to the domain.

Sample questions from the question set are listed below:
What is the position of the gitanjali Express.
What is the fare from Nagpur to gondia.
When howrah mail is coming.
How much late mangla express is running.
Whether gitanjali express having stoppage at Akola.
By what time gitanjali express reaches wardha.
In which platform Vidarbha Express will arrive.
How many trains are available from nagpur to raipur.
What is the fair for A/C two tier for the train vidarbha Express.
What is the difference in fair for ac class and sleeper class.
How much amount will be deducted if we cancel the ticket before 24 hours.
How many trains are available from nagpur to delhi.
Is any direct train to goa from nagpur.
What is the route to jaipur from nagpur.
Is any concession in the fare for student's educational tour.
What is the fare for the sport team from akola- nagpur.
What percent of concession is given to the handicappedperson.
How many stoppage does rajdhani express has.
How many trains are available for Mumbai from nagpur on Wednesday.
How many superfast trains are available for Mumbai from nagpur on Wednesday.
Whether charges of Doronto is more than superfast.
List all trains from nagpur to raipur.

The corpus is then analyzed thoroughly on different aspects that will be useful in designing the context free grammar for syntax analysis purpose and the NLIDB as a whole. Some graphs are also plotted based on the observations like length of questions in words, starting word of question and addition in question corpus with respect to number of person surveyed. These graphs are shown in (Fig.1, 2 & 3).

The graph plotted for percentage addition in corpus with respect to number of person surveyed is decreasing about exponentially with increase in number of person surveyed as shown in (Fig.1). This graph shows that for collecting sufficient number of questions in corpus it is not necessary

to survey large number of people. For a well-defined and useful domain a sufficiently large corpus can be obtained by surveying a small number of people. As we increase the number of people for surveying chances of repetitive questions increases which do not contribute in addition in corpus.

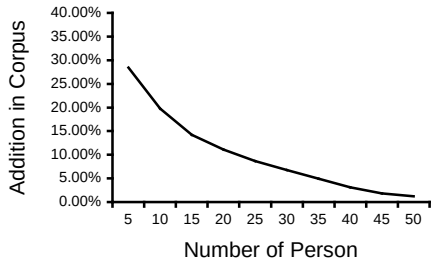

Fig.1 Corpus Collection: Addition in corpus with respect to number of person surveyed

The graph for question length in the collected corpus shows that minimum length is five words and maximum is eighteen as shown in (Fig.2). Maximum number of questions (more than three forth) in the corpus is having length between seven and eleven. It is an important observation that questions in such domains are small in length. This observation affects semantic analysis as many times question may not contains sufficient information for its interpretation. In such a situation discourse and pragmatics plays role to decide meaning of question. This observation also favors deep semantic analysis against shallow one that normally requires complete and semantically tractable questions [9].

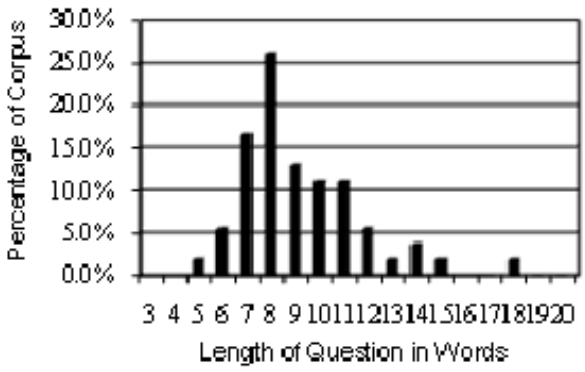

Fig.2 Question Length in Corpus

Graph showing possible starting word of question shown in (Fig.3) is simple one and is used to design context free grammar rules for representing syntax of questions. In our domain maximum questions (more than 50%) are starting from what and how but actually it depends on domain and people surveyed.

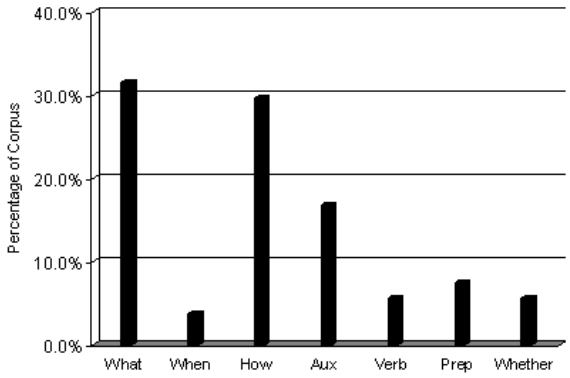

Fig.3 Starting word percentage in corpus

3. DESIGN OF GRAMMAR BASED ON THE CORPUS

The question set is analyzed thoroughly to understand the patterns of question asked for the domain. Each question is a unique in nature but still there are many common things that can be extracted like start of question. Question may start with Wh-word, Auxiliary verb, main verb or prepositions [6]. Depending on type of question information to be extracted found in different constituents of the input question. Based on such observations a grammar for the corpus needs to be designed. While designing a grammar it is very important to note that what kind of information would be required to extract by parsing the given question. After observing the questions it is found that every question contains three elements, which will be very useful in semantic analysis [7]. These elements are:

- The standard question items (“What is the ….?”, “How many …?”, “Is there any …?”)
- The goal of the query (i.e. which values have to be retrieved and reported to the user?)
- The restrictions, enabling the system to extract the relevant items

Based on the presence of these three elements a grammar is to be designed for the natural language questions. For example if question is starting from “how many” then the noun phrase immediately following it will be the goal of query and the phrases following to verb describes the restriction part. However all questions are not that much simple to interpret, it requires a detail analysis to find relationship among constituents of a sentence [10]. So basically in syntax analysis different constituents of input question needs to be identified and later on these can be related to each other to interpret meaning. Here complete grammar of English is not required as in domain specific

questions only limited variations in syntax is used while grammar of complete language is very complicated and large in size. Moreover users of such system may not be caring exact and all rules of language. So in such a case simple and tolerate grammar is required. These simple syntactic rules can be modeled by context free grammar (CFG). CFG are powerful enough to express sophisticated relations among the constituents in a sentence, yet computationally tractable enough that efficient algorithms exist for parsing sentences.

3.1 Context Free Grammar

Context free grammar is defined by four tuples [5],
G = (V, T, P, S) where
V is set of Non terminal Symbols (It can be replaced by any other string of symbol)
T is set of Terminal Symbols (It cannot be replaced by any other string)
P is set of Production and (Defines replacement rules)
S is starting symbol of the grammar. (Every valid string can be generated from it)

In the designed grammar terminal symbols are the part of speech assigned to individual words of the question like noun, preposition, verb, adjectives etc. Morphological analyzer does assignment of this part of speech. Non-terminal symbols are the syntactic categories of English language grammar, which represents a set of strings. Each string of the set is a meaningful constituent of an english sentence. Set of Non-terminal symbol includes noun phrase, verb phrase, adjectival phrase, prepositional phrase etc. The non-terminal S that also is the starting symbol of the grammar represents an input question. Production Set defines the rules for replacement used for replacing a non-terminal by any string of terminal or non-terminal symbols. The production rules with example are given in (Fig.4).

First production rule is for an input question represented by symbol S. A question can start with a verb followed by a noun phrase and then verb phrase or only noun phrase. It includes all imperative and yes/no question structure. Most commonly used questions starts with Wh-words. To represent such questions a non-terminal Wh-NP is included which can be replaced by simply a Wh-word or Wh-word followed by Nominal. In question this Wh-NP can be followed by either verb pharse only or Auxiliry verb followed by NP and then followed by VP. In last production of S question starts with a preposition followed by Wh-NP then NP and VP. Similarly productions of other non-terminals like Noun Phrase, Verb Phrase, Prepositional Phrase, Adjectival Phrase and Nominal defines rules by which that can be generated. Each rule followed by an example for easy understanding.

The grammar described in previous section is successfully tested on test corpus by using YACC [8]. YACC generates code for LALR parser [2] which uses bottom up technique. For using YACC, grammar was provided in a file called .y file [8]. It uses lexical specification defined in .l file. For testing purpose, only simple morphological rules are used in .l file [8] to define valid tokens and also it assigns part of speech to each token. The parser generated through YACC takes a question from test corpus and makes parse tree for it. An example parse tree is given in (Fig. 5).

Terminals:

N – Noun
P – Preposition
Aux – Auxiliary Verb
Verb – General Verb
Adj – Adjective
Wh-Words – Wh-Words (what, when,..)
Det – Determiner (a, an, the)
Card – cardinal number (one, two, three)
Ord – Ordinal Numbers (first, second)
Quant – Quantifiers (many, more, any)
Conj – Conjunctions (and, or)

Non Terminals

S → Query
NP → Noun Phrase
VP → Verb Phrase
NOM → Nominal
V → Verb
PP→ Preposional Phrase
Wh-NP → Wh Noun Phrase
AP → Adjectival Phrase

4. CONCLUSIONS AND FUTURE SCOPE

For the practical use of Natural language interface to database, deep analysis is to be implemented. To implement deep analysis the detail analysis of the input question is required. This method makes use of more linguistic knowledge than the domain knowledge. The first step for the detail analysis is grammar design. This paper described the context free grammar design for the railway inquiry domain. The grammar was based on the corpus of questions collected for the domain. The paper also describes results of different analysis done on the corpus that are used to design the grammar. The grammar is successfully tested on test corpus using YACC generated parser.

Context Free Grammar for the Question Set

Production Rules	Example
S ? V NP VP	Does + garibrath + has a stoppage at Mumbai
| V NP	Is + any direct train to goa from nagpur
| Wh-NP VP	What + is the departure time of Mangla express
| Wh-NP Aux NP VP	How many stoppage + does + rajdhani express + has from Mumbai to Delhi
| P Wh-NP NP VP	By + what time + vidarbha Express + reaches Madras
NP ? NP Conj NP	departure + and + arrival
| (Det) (Card) (Ord) (Quant) (AP) NOM	the + next + more + fast + train
	(Det) (Ord) (Quant) (AP) NOM
NOM ? N NOM	departure + time
| N	concession
| NOM (PP)*	departure time + of the train
VP ? V	is coming
| V NP	reaches + wardha
| V NP PP	having + stoppage + at akola
| V PP	going + to mumbai
PP ? P NP	to + Mumbai
V ? Verb	coming
| Aux	is
| Aux V	is + coming
Wh-NP ? Wh-Word	what
| Wh-Word NOM	when + howrah mail
| Wh-Word Quant	how + much
AP ? Adj AP	summer + special
| Adj	special

Fig.4 Context free grammar with example

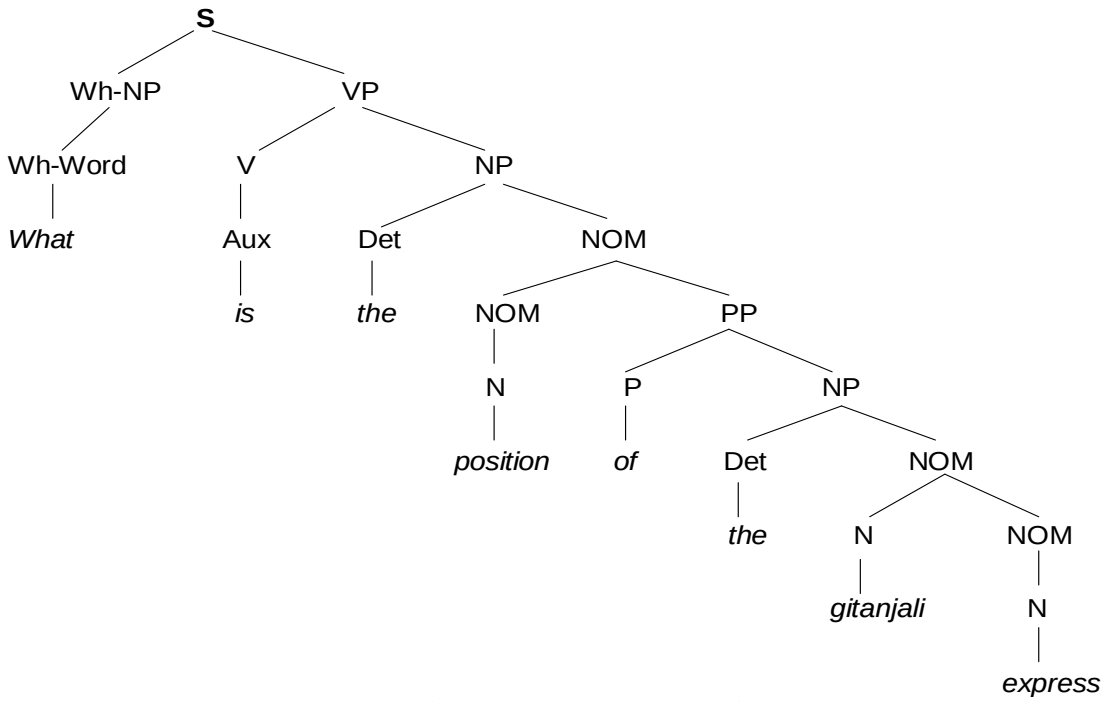

Fig.5 A sample Parse tree for question "What is the position of the gitanjali express

After deciding the syntax of input questions some semantic structure can be used to represent the meaning of the given natural language query. Now a days, NLP research has progressed tremendously and many techniques are available at lexical and syntax level. The basic problem of NLIDB lies in semantic analysis. Deep semantic analysis increases the success rate and portability by using semantic structures defined previously. This semantic structure may be either First order predicate calculus (FOPC), Head driven phrase structured grammar (HPSG), Frame structure, Semantic network etc [3]. An appropriate semantic structure needs to be selected and designed for the domain to represent meaning of given question.

References

[1] Agrawal Avinash J., Using Domain Specific Question Answering Techniques for Automatic Railways Inquiry on Mobile Phone, 5th International Conference on Information Technology: New Generations (ITNG 2008), Las Vegas, Nevada, USA, April 7-9, 2008,1111-1116.

[2] Aho, Ullman, Principle of Compiler Design, A book Published by Narosa Publications, 2002.

[3] Allen James, Natural Language Understanding, A Book Published by Addison Wesley Publications, 1994.

[4] Androutsopoulos I., Ritchie G.D., and Thanisch P., Natural Language Interfaces to Databases – An Introduction, Journal of Natural Language Engineering Part 1, 1995, 29–81.

[5] Chomsky, Noam, Three models for the description of language, Information Theory, IEEE Transaction 2, Sept. 1956,56.

[6] Jurafsky D., Martin J., Speech and Language Processing, A Book Published by Prentice Hall Publications, 2008.

[7] Lesmo Leonardo, Natural Language Query interpretation in restricted domains, 6th International Conference on NLP ICON-08, Pune, India, 2008.

[8] Levine John, Lex & Yacc, A book published by O'Reilly Publication, 1992.

[9] Minock, Michel, Where are the killer application of restricted domain question answering, In proceeding of the IJCAI Workshop on Knowledge Reasoning in Question Answering, Edinbergh, Scotland, 2005,page 4.

[10] Minock Michael, Olofsson Peter, Naslund Alexander, Towards Building Robust Natural Language Interface to Database, 13th International Conference on Applications of Natural Language to Information Systems, NLDB 2008 London, UK, 2008,187-198.

[11] Popescu Ana-Maria, Etzioni Oren, and Kautz Henry, Towards a Theory of Natural Language Interfaces to Databases, IUI, 2003,149–157.

[12] Popescu Ana-Maria, Armanasu Alex, Etzioni Oren, Ko David, and Yates Alexander, Modern Natural Language Interfaces to Databases : Composing Statistical Parsing with Semantic Tractability, 20th international conference on Computational Linguistics COLING-2004,Geneva, Switzerland, 2004,141.

[13] Wong Yuk Wah, Learning for Semantic Parsing Using Statistical Machine Translation Techniques, Technical Report UT-AI-05-323, University of Texas at Austin, Artificial Intelligence Lab, 2005.

Avinash J. Agrawal received Bachelor of Engineering Degree in Computer Technology from Nagpur University, India and Master of Technology degree in Computer Technology from National Institute of Technology, Raipur, India in 1998 and 2005 respectively. He is currently pursuing Ph.D. from Visvesvaraya National Institute of Technology, Nagpur. His research area is Natural Language Processing and Databases. He is having 12 years of teaching experience. Presently he is Assistant Professor in Shri Ramdeobaba Kamla Nehru Engineering College, Nagpur. He is the author of seven research papers in International and National Journal, Conferences.

Dr. O. G. Kakde received Bachelor of Engineering degree in Electronics and Power Engineering from Visvesvaraya National Institute of Technology (formerly Visvesvaraya Regional College of Engineering), Nagpur, India and Master of Technology degree in Computer Science and Engineering from Indian Institute of Technology, Mumbai, India in 1986 and 1989 respectively. He received Ph.D. from Visvesvaraya National Institute of Technology, Nagpur, India in 2004. His research interest includes theory of computer science, language processor, image processing, and genetic algorithms. He is having over 22 years of teaching and research experience. Presently he is Professor and Dean, Research and Development at Visvesvaraya National Institute of Technology, Nagpur, India. He is the author or co-author of more than thirty scientific publications in international journals, international conferences, and national conferences. He also authored five books on data structures, theory of computer science, and compilers. He is the life member of Institution of Engineers, India. He also worked as the reviewer for international and national journals, international conferences, and national conferences and seminars.