# Query Optimisation using Natural Language Processing

[1]Ashish Tamrakar, [2]Deepty Dubey
[1,2]Dept. of CSE, Chhatrapati Sivaji Institute of Technology, CG, India

**Abstract**
Interesting change has come in Natural Language Processing (NLP) has seen in both research direction and methodology in the past several years. Increased work in computational linguistics tended to focus on purely symbolic methods. Now various work is shifting toward hybrid methods that combine new empirical corpus-based methods, including the use of probabilistic and information theoretic techniques, with traditional symbolic methods. Need of Natural Language Query Processing is for an English sentence to be interpreted by the computer and appropriate action taken. Question answering to any databases in natural language is a very convenient and easy method of data access, especially for casual users who do not understand complicated database query languages such as SQL So this paper proposes the architecture for translating English Query into SQL using Semantic Grammar

**Keywords**
NLP, SQL, Query Optimisation, AI

## I Introduction

Natural Language is the language used for communication amongst human beings in the real world. The term real world makes the problem much more difficult. While the term Natural Language (NL) refers to the language spoken by human beings, Natural Language Processing (NLP) refers to an area of Artificial Intelligence (AI) that deals with systems and programs that can accept comprehend and communicate in natural language. Systems that are capable of processing and understanding natural language bridge the man-machine communication barriers to a great extent. Today one of the most targeted problems in the field of artificial intelligence (Computer Science) is to make machine this much intelligent so then it can almost behave like a human being. Some of the behaviors of human beings have been accomplished during machine implementation e.g. now days machines can hear with the use of microphone, speak by producing sound, see with the use of cameras, smell with sensors but still there are some areas where this machine development is not completely successful and some of them are to understand natural language, learning from experience and making autonomous decisions in real time environment etc.

## II Related Work of NLP

Phonology: This level deals with the interpretation of speech sounds within and across words. There are, in fact, three types of rules used in phonological analysis:

### A. Phonetic rules

for sounds within words.

### B. Phonemic rules

for variations of pronunciation when words are spoken together, and.

### C. Prosodic Rules

for fluctuation in stress and intonation across a sentence. In an NLP system that accepts spoken input, the sound waves are analyzed and encoded into a digitized signal for interpretation by various rules or by comparison to the particular language model being utilized.

### 1. Morphology

This level deals with the componential nature of words, which are composed of morphemes – the smallest units of meaning. For example, the word preregistration can be morphologically analyzed into three separate morphemes: the prefix pre, the root registration, and the suffix. Since the meaning of each morpheme remains the same across words, humans can break down an unknown word into its constituent morphemes in order to understand its meaning. Similarly, an NLP system can recognize the meaning conveyed by each morpheme in order to gain and represent meaning. For example: Adding the suffix –ed to a verb, conveys that the action of the verb took place in the past. This is a key piece of meaning, and in fact, is frequently only evidenced in a text by the use of the -ed morpheme.

### 2. Lexical

At this level, humans, as well as NLP systems, interpret the meaning of individual words. Several types of processing contribute to word-level understanding – the first of these being assignment of a single part-of-speech tag to each word. In this processing, words that can function as more than one part-of-speech are assigned the most probable part-of speech tag based on the context in which they occur. Additionally at the lexical level, those words that have only one possible sense or meaning can be replaced by a semantic representation of that meaning. The nature of the representation varies according to the semantic theory utilized in the NLP system. The following representation of the meaning of the word launch is in the form of logical predicates. As can be observed, a single lexical unit is decomposed into its more basic properties. Given that there is a set of semantic primitives used across all words, these simplified lexical representations make it possible to unify meaning across words and to produce complex interpretations, much the same as humans do.
Ex. "Large boat used for carrying people on rivers, lakes harbors, etc.) ((CLASS BOAT) (PROPERTIES (LARGE) (PURPOSE (PREDICATION (CLASS CARRY) (OBJECT PEOPLE))))"

### D. Syntactic

This level focuses on analyzing the words in a sentence so as to uncover the grammatical structure of the sentence. This requires both a grammar and a parser. The output of this level of processing is a (possibly delinearized) representation of the sentence that reveals the structural dependency relationships between the words. There are various grammars that can be utilized, and which will, in turn, impact the choice of a parser. Not all NLP applications require a full parse of sentences, therefore the remaining challenges in parsing of prepositional phrase attachment and conjunction scoping no longer stymie those applications for which phrasal and clausal dependencies are sufficient. Syntax conveys meaning in most languages because order and dependency contribute to meaning. For example the two sentences: 'The dog chased the cat.' and 'The cat chased the dog.' differ only in terms of syntax, yet convey quite different meanings.

### E. Semantic

This is the level at which most people think meaning is determined, however, as we can see in the above defining of the levels, it is all the levels that contribute to meaning. Semantic processing determines the possible meanings of a sentence by focusing on the interactions among word-level meanings in the sentence. This level of processing can include the semantic disambiguation of words with multiple senses; in an analogous way to how syntactic disambiguation of words that can function as multiple parts-of-speech is accomplished at the syntactic level. Semantic disambiguation permits one and only one sense of polysemous words to be selected and included in the semantic representation of the sentence. For example, amongst other meanings, 'file' as a noun can mean either a folder for storing papers, or a tool to shape one's fingernails, or a line of individuals in a queue. If information from the rest of the sentence were required for the disambiguation, the semantic, not the lexical level, would do the disambiguation. A wide range of methods can be implemented to accomplish the disambiguation, some which require information as to the frequency with which each sense occurs in a particular corpus of interest, or in general usage, some which require consideration of the local context, and others which utilize pragmatic knowledge of the domain of the document [6].

### F. Discourse

While syntax and semantics work with sentence-length units, the discourse level of NLP works with units of text longer than a sentence. That is, it does not interpret multisentence texts as just concatenated sentences, each of which can be interpreted singly [13]. Rather, discourse focuses on the properties of the text as a whole that convey meaning by making connections between component sentences. Several types of discourse processing can occur at this level, two of the most common being anaphora resolution and discourse/text structure recognition. Anaphora resolution is the replacing of words such as pronouns, which are semantically vacant, with the appropriate entity to which they refer [7]. Discourse/text structure recognition determines the functions of sentences in the text, which, in turn, adds to the meaningful representation of the text. For example, newspaper articles can be deconstructed into discourse components such as: Lead, Main Story, Previous Events, Evaluation, Attributed Quotes, and Expectation [17] .

### G. Pragmatic

This level is concerned with the purposeful use of language in situations and utilizes context over and above the contents of the text for understanding The goal is to explain how extra meaning is read into texts without actually being encoded in them [15]. This requires much world knowledge, including the understanding of intentions, plans, and goals. Some NLP applications may utilize knowledge bases and inferencing modules. e.g., the following two sentences require resolution of the anaphoric term 'they', but this resolution requires pragmatic or world knowledge [16].

## III. Previous Work

The very first attempts at NLP database interfaces are just as old as any other NLP research. In fact database NLP may be one of the most important successes in NLP since it began. Asking questions to databases in natural language is a very convenient and easy method of data access, especially for casual users who do not understand complicated database query languages such as SQL [14]. The success in this area is partly because of the real-world benefits that can come from database NLP systems, and partly because NLP works very well in a single-database domain. Databases usually provide small enough domains that ambiguity problems in natural language can be resolved successfully. Here are some examples of database NLP systems. LUNAR (Woods, 1973) involved a system that answered questions about rock samples brought back from the moon. Two databases were used, the chemical analyses and the literature references. The program used an Augmented Transition Network (ATN) parser and Woods' Procedural Semantics. The system was informally demonstrated at the Second Annual Lunar Science Conference in 1971 [1]. LIFER/LADDER was one of the first good database NLP systems. It was designed as a natural language interface to a database of information about US Navy ships. This system, as described in a paper by Hendrix, used a semantic grammar to parse questions and query a distributed database. The LIFER/LADDER system could only support simple one-table queries or multiple table queries with easy join conditions [4].

## IV. Proposed Approach

To process a query, the first step is speech tagging; followed by word tagging. The second step is parsing the tagged sentence by a grammar. The grammar parser analyzes the query sentence according to the tag of each word and generates the grammar tree/s. Finally, the SQL translator processes the grammar tree to obtain the SQL query [8].

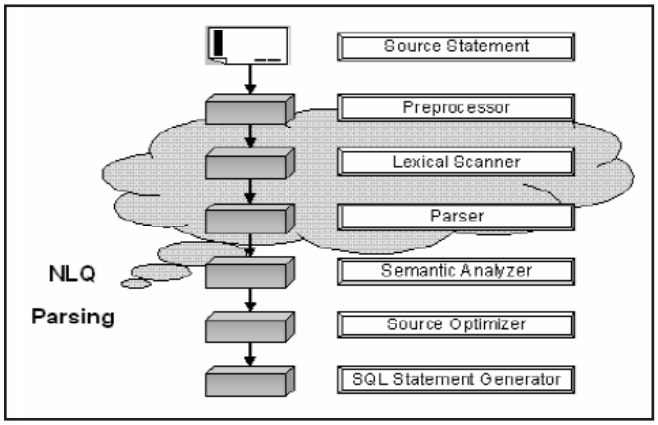


Fig. 1: Architecture of NLDBI System

Consider a sentence w1m which is a sequence of words w1 w2 w3… wm (ignoring punctuations), and each string wi in the sequence stands for a word in the sentence. The grammar tree of w1m can be generated by a set of predefined grammar rules; usually more than one grammar tree may be generated. The formalizing capability of grammar help in describing most sentence structures and built efficient sentence parsers. A parser is one of the components in an interpreter or compiler, which checks for correct syntax and builds a data structure (often some kind of parse tree, abstract syntax tree or other hierarchical structure) implicit in the input tokens [11]. The parser often uses a separate lexical analysis to create tokens from the sequence of input characters. Parsers may be programmed by hand or may be semi automatically generated (in some programming language) by a tool (such as Yacc) from a grammar written in Backus-Naur form. The SQL translator generates query in SQL [9]. Using grammar the parse tree is obtained from the input statement. The leaves of the parse tree are translated to corresponding SQL. Fig. 2, depicts the processing of English input statement to generate SQL query. The entire process involves tagging of input statement, apply grammar and

semantic representation to generate parse tree, analyze the parse tree using grammar and translating the leaves of the tree to generate corresponding SQL query [10].

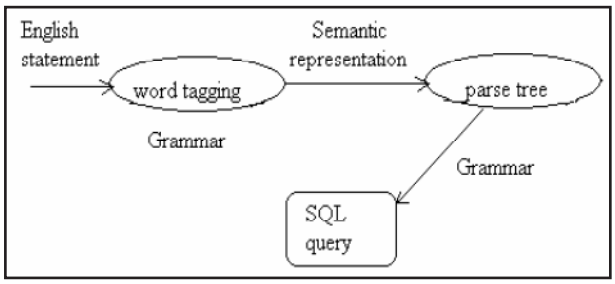


Fig. 2: Generation of SQL Query from English Statement

The database tables considered are EMP (empid, empname, salary, edepid, address, post, mobileno), DEPT (deptid, deptname, deptloc, dcapacity) and PROJECT (pid, pname, epid). From the input NL statement, to generate parse tree the grammar written based on database tables is:

WhatKeyBank → for | of | with | is | where | whose | having | in | on
AAnTheBank → a | an | the
empid → integer | id | number
empname → string | name
salary → integer | salary | income | earning
mgrid → integer | manager | boss | superior
edeptid → integer | id | number
deptid → integer | id | number
deptname → string | name
deptloc → string | location
dcapacity → integer | capacity
EmpTable → employee | worker | person | emp | employees | emps | workers |persons
ProjectTable → project | projects
DeptTable → department | dept | dpt | departments | depts. | dpts

The experimental work is to design an interface for generating queries from natural language statements/questions. It also consists of designing a parser for the natural language statements, which will parse the input statement, generate the query and fire it on the database. The experimental work will understand the exact meaning the end user wants to go for, generate a what-type sentence and then convert it into a query and handover it to the interface. The interface further processes the query and searches for the database. The database gives the result to the system which is displayed to the user. The following modules were developed [12].

### A. An Interface

It allows the user to enter the query in NL, interact with the system during ambiguities and display the query results.

### B. Parsing

Derives the Semantics of the statement given by the user and parses it into its internal representation, to convert NL input statement into what- type question for selection of data.

### C. Query Generation

It generates a query against the user statement in SQL and passes on to the database.

The structure of an algorithm is given below

**Step 1:**

Take i/p of natural language sentence
Call NLP ENGINE
Receive array from NLP engine
If array not null
Display
Else
no record is found

**Step 2:**

NLP ENGINE
Parse the sentence
Call name entity reorganization
Extract noun form the sentence
Stored in the array of variable

**Step 3:**

Call semantic processing
Extract the possible meaning from the sentence
Make syntax from the array of noun and semantic mining
Convert into sql query
Execute the query in sql engine
Store in array
Return array

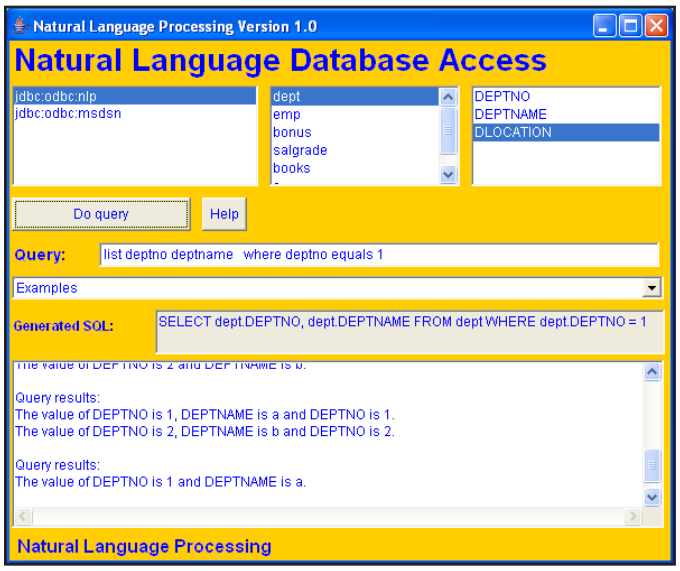


Fig. 3:

## V. Conclusion

Natural Language Processing can bring powerful enhancements to virtually any computer program interface. This system is currently capable of handling simple queries with standard join conditions. Because not all forms of SQL queries are supported, further development would be required before the system can be used within NLDBI. Alternatives for integrating a database NLP Component into the NLDBI were considered and assessed.

## References

[1] Huangi,Guiang Zangi, Phillip C-Y Sheu,"A Natural Language database Interface based on probabilistic context free grammar", IEEE International workshop on Semantic Computing and Systems 2008.
[2] Akama, S. Logic,"language and computation", Kulwer Academic publishers, pp. 7-11, 1997.

[3] ELF Software CO.,"Natural-Language Database Interfaces", from ELF Software Co., cited November 1999, [Online] Available: http://hometown.aol.com/elfsoft/
[4] Hendrix, G.G., Sacerdoti, E.D., Sagalowicz, D., Slocum, J.,"Developing a natural language interface to complex data", in ACM Transactions on database systems, 3(2), pp. 105- 147, 1978.
[5] Joseph, S.W., Aleliunas, R.,"A knowledge-based subsystem for a natural language interface to a database that predicts and explains query failures", in IEEE CH, pp. 80-87, 1991.
[6] Mitrovic, A.,"A knowledge-based teaching system for SQL", University of Canterbury, 1998.
[7] Moore, J.D.,"Discourse generation for instructional applications: making computer tutors more like humans", in Proceedings AI-ED, pp. 36-42, 1995.
[8] Suh, K.S., Perkins, W.C.,"The effects of a system echo in a restricted natural language database interface for novice users", in IEEE System sciences, 4, pp. 594-599, 1994.
[9] Whenhua, W., Dilts, D.M.,"Integrating diverse CIM data bases: the role of natural language interface", in IEEE Transactions on systems, man, and cybernetics, 22(6), pp. 1331-1347, 1992.
[10] Dan Klein, Christopher D. Manning: Corpus-Based Induction of Syntactic Structure: Models of Dependency and Constituency. ACL 2004, pp. 478-485.
[11] In-Su Kang, Jae-Hak J. Bae, Jong-Hyeok Lee, "Database Semantics Representation for Natural Language Access", Department of Computer Science and Engineering, Electrical and Computer Engineering Division Pohang University of Science and Technology (POSTECH) and Advanced Information Technology Research Center (AITrc), 2002.
[12] Woods, W., Kaplan, R.,"Lunar rocks in natural English: Explorations in natural language question answering", Linguistic Structures Processing. In Fundamental Studies in Computer Science, 5, pp. 521-569, 1977.
[13] Androutsopoulos, I., Richie, G.D., Thanisch, P.,"Natural Language Interface to Database–An Introduction", Journal of Natural Language Engineering, Cambridge University Press. 1(1), pp. 29-81, 1995.
[14] "Linguistic Technology", English Wizard – Dictionary Administrator's Guide. Linguistic Technology Corp., Littleton, MA, USA, 1997.
[15] Dan Klein, Christopher D. Manning,"Corpus-Based Induction of Syntactic Structure", Models of Dependency and Constituency. ACL 2004, pp. 478-485.
[16] M-C.de Marneffe, B. MacCartney, C. D. Manning, "Generating Typed Dependency Parses from Phrase Structure Parses", In Proceedings of the IEEE /ACL 2006 Workshop on Spoken Language Technology. The Stanford Natural Language Processing Group. 2006.
[17] Dan Klein, Christopher D. Manning,"Fast Exact Inference with a Factored Model for Natural Language Parsing", In Advances in Neural Information Processing Systems 15 (NIPS 2002), Cambridge, MA: MIT Press, pp. 3-10, 2003
[18] Marie-Catherine de Marneffe, Bill MacCartney, Christopher D. Manning,"Generating Typed Dependency Parses from Phrase Structure Parses", In LREC 2006.

Ashish Tamrakar received his B.Sc. degree in Computer Science from Kalyan College, Bhilai in 2007, the M.Sc. degree in Computer Science from Kalyan College Pt. Ravishankar University, Raipur(C.G.) in 2009, and the M.Tech. degree pursuing in Computer Science and Engineering from Chhattisgarh Swami Vivekananda Technical University (Bhilai) Chhattisgarh, in 2010-2012. I am currently working as an assistant professor, with Department of Computer Science from Bhilai School of Engineering (C.S.V.T.U.).