ISSN (Online) : 0974-5645
ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, Vol 8(1), 01–10, January 2015
DOI: 10.17485/ijst/2015/v8i1/54123

Natural language to SQL Generation for Semantic Knowledge Extraction in Social Web Sources

K. Javubar Sathick*[*] and A. Jaya

Department of Computer Applications, B.S. Abdur Rahman University, Chennai, India;
javubar@bsauniv.ac.in, jaya@bsauniv.ac.in

Abstract

Enormous evolution of web data creates a peculiar myth in the field of computer and information technology for extracting the meaningful content from the web. Many organizations and social networks use databases for storing information and the data will be fetched from the specified data store. Data can be retrieved or accessed by SQL queries whereas the query is in the form of natural lingual statement which has to be processed. So, the primary objective of this research article is to find the suitable way to convert natural language query to SQL and make the data apt for semantic extraction. This Research paper also aims to derive an automatic query translator for Natural Language based questions into their associated SQL queries and provides an user friendly interface between end user and the database for easy access of social web data from different web sources such as facebook, twitter and linkedIn etc.,. This paper is implemented using java as the front end, SQL server as the back end and R-tool is used to collect the data from social web sources. This research article provides an optimized SQL query generation for the Natural Language question provided by the end user.

Keywords: Natural Language Interface for Databases (NLIDB), Natural Language Processing (NLP), R-tool, Semantic Knowledge Extraction (SKE), Structured Query Language (SQL), Social Web Data

1. Introduction

Data Storage plays an important role in today's commercial system especially with progression of social media, the size of the data in database and data accessing pace become more crucial part in the recent research world. Plenty of new database tools and emerging technologies are growing in a wide spectrum, therefore the provision for storing the huge set of data is available, but the fact that the technology or an interface which can process the user request and pulls the exact data as per the request from the huge database is not familiarized as that of storage provisions. Most of the businesses and social sites need these types of applications by using the SQL language. Natural Language Processing (NLP) is becoming one of the most active techniques used in Human-computer Interaction which plays a vital role since the social media started playing its part to a large extent in the current trend. In the context of social media the query conversion is quite important in terms of bringing out the exact data which is requested by the web users who surf on the net. The query /request will be of natural statement such as blog, comment, tweets etc., these statement must be converted into a most reliable and acceptable form of typical query which system can understand and crack the exact data from the database. So these factors are acting as a precious evidence for implementing the proposed work through this article. The objective of NLP is to facilitate communication between human and computers without memorization of multifaceted instructions and procedures. In other words, NLP is the technique that can make the workstation to understand the natural languages used by users. In the current world the basic requirement of commercial and social based system is to extract the data from a database such as MS Access, Oracle and Hadoop where the large collection of data is stored. An end user or layman without

[*] *Author for correspondence*

knowledge of SQL may find quite difficult to extract the data with the complicated query in corresponding with the database. Therefore in this work the development of system for people to interact with the database in simple English language is implemented and analyzed for the accuracy. This enables a user to input their queries in simple English and get the answer in same language which is referred as Natural Language Interface to a Database (NLIDB).The knowledge extraction is enabled with the successful implementation of SQL generation from the natural language statement. Once the query is processed, then extracting the semantic knowledge from the social web sources done by analyzing the data collected in the R-tool interface.

The paper is organized as follows. Section 2 provides background and related work.Section 3 describes the architectural layout of the proposed work which projects the actual flow of NLP.Section 4 discusses the Implementation and results obtained out of sample query processing experiment conducted. Section 5 briefly analyse the generated the SQL with Precision and Recall Threshold measure.Section 6 summarizes the most important conclusions of research work and draws the future lines of research

2. Background and Related Works

The various attempts have been made so far for conversion of natural language to dedicate the SQL query for easy accessing of database. NLP database interfaces are just as old as any other NLP research. Posting query to databases in natural language is a exceptionally fitting and uncomplicated method of data access, especially for unfussy users who do not recognize the complex database query languages such as SQL. The accomplishment in this area is partly because of the real-world remuneration that can come from database NLP systems, and to a certain extent because NLP works very well in a single-database domain, but the fact that the social web sources is the combination of various aspects, therefore the flexible interface is need. Some of the contributions made by several authors were highlighted in this section. Databases usually provide small enough domains that ambiguity problems in natural language can be resolved successfully. Akshay G. Satav et al[1] proposes a system that will provide search interface/ NLP System for users without knowing any specific syntax or knowledge of a database language. Hence the author present a system that will provide the search interface for users especially for online applications, search engines and many other different databases, where accuracy and efficiency are most important terms required. Various analyses shows user is not restricted to formulate any kind of query so this system provides result to users any type of query he fires to the system accurately and efficiently. The semantic search is not enabled in the system which is enabled in this research work.

Gaganpreet Kaur in[2] emphasize about a usage of regular expressions in NLP to search text is well known and understood as a useful technique. Regular Expressions are generic representations for a string or a collection of strings. Regular expressions (regexps) are one of the most useful tools in computer science. NLP, as an area of computer science, has greatly benefitted from regexps: they are used in phonology, morphology, text analysis, information extraction, & speech recognition. It helps a reader to give a general review on usage of regular expressions from natural language processing, whereas the clear collection of expression in sentence is not clearly specified which is acting as a issue which will be addressed in this research article.

Mahesh P. Gaikwad in[3] discussed about the need for natural language interfaces to database has become increasingly acute as more and more people access information from web browsers. Yet NLI (Natural Language Interface) is only usable if they map natural language questions to SQL queries correctly. Natural Language processing is becoming one of the most active areas in Human-Computer Interaction. The goal of NLP (Natural Language Processing) is to enable communication between people and computers without requiring to memorization of complex commands and procedures. The main purpose of natural Language Query Processing is for an English sentence to be interpreted by the computer and appropriate action taken. The use NLI in NLP is elaborated in this article to a large extent to attain the actual purpose of query processing system.

Jasmeen Kaur et al in[4] describes about the purpose of Natural Language Query Processing which is used to interpret an English sentence and hence a complementary action is taken. Querying to databases in natural language is a convenient method for data access, especially for newbie's who have less knowledge about complicated database query languages such as SQL. The author emphasize on the structural designing methods for translating English Query into SQL using automata. A system that is capable of handling simple queries with standard join conditions is introduced here but because not all forms of SQL queries are supported, further development would be required which is carried out in the proposed research through this article.

As database plays a vital role, here are some examples of database NLP systems proposed by Anil M. Bhadgale et al[5] 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. 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 (1978), 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.

Avinash J. Agrawal, O. G. Kakde[6] describes a method for semantic analysis of natural language queries for Natural Language Interface to Database (NLIDB) using domain ontology. Implementation of NLIDB for serious applications like railway inquiry, airway inquiry, corporate or government call centers requires higher precision. This can be achieved by increasing role of language knowledge and domain knowledge at semantic level. Also design of semantic analyzer should be such that it can easily be ported for other domains as well. Intermediate result of the system is evaluated for a corpus of natural language queries collected from casual users who were not involved in the system design. The domain ontology has the minimum level of data extraction tendency which is enhanced in this research work.

In[7] Saravjeet Kaur et al discussed about an interface module that converts user's query given in natural language into a corresponding SQL command. Asking questions to databases in natural language like English is a very convenient and easy method of data access from database system, especially for normal users who do not understand complicated database query languages such as SQL. Syntactic analysis and semantic analysis of natural language query takes place for relevant and exact conversion of structured query. The complete semantic conversion is not attained due to complex sentences as a query statement.

Arati K. Deshpande and Prakash. R. Devale in[8], proposed "Natural Language Processing using probabilistic context free grammar", the author discussed a method to create new NLDBI system using Probabilistic Context Free Grammar (PCFG). This paper highlights, Natural language statement is converted into internal representation based on the syntactic and semantic knowledge of the natural language. This representation is then converted into queries using a representation converter, but the optimization factor is missing in finding the right grammar which is handled in this article.

Dshish Tamrakar Dshish Tamrakar, published a article[9] titled "Query Optimisation using Natural Language Processing", the author proposed the architecture for translating English Query into SQL using semantic Grammar. LIFER/LADDER method used in the syntax analysis. The LIFER/LADDER system could only support simple one table Queries or multiple table queries with easy join conditions which restricts the system to a large extent.

An important issue proposed in[10] by Michael Gage is re-emerging in the field of relational database management systems is the ability for non-expert users to access stored data using the more powerful aspects of the Structured Query Language (SQL). The widespread use of relational database management systems in industry as well in scientific research has increased the need for a solution to this issue. The method used will allow non-expert users more successfully obtain data is the use of an artificial intelligence application to process natural language from the user in the form of a question or sentence into a SQL statement. Author explores the foundations of this field as well as the branches of the more recent approaches including multi-lingual solutions, phrase recognition and substitution, SQL keyword mapping, and fuzzy logic applications. Some of the aspects were re-analyzed in the current work.

Neelu Nihalani et al in[11] describes and proposed about the Structured Query Language (SQL) norms which are been pursued in almost all languages for relational database systems. However, not everybody is able to write SQL queries as they may not be aware of the structure of the database. So this has led to the development of Intelligent Database System (IDBS). There is an overwhelming need for non-expert users to query relational databases in their natural language instead of working with the values of the attributes. As a result many intelligent natural language interfaces to databases have been developed, which provides flexible options for manipulating queries. The idea of using Natural Language instead of SQL has prompted the development of new type of processing called Natural language Interface to Database. NLIDB is a step towards the development of intelligent database systems (IDBS) to enhance the users in performing flexible querying in databases. The author emphasizes on the overview of NLIDB which can be enhanced.

Alessandra Giordani and Alessandro Moschitti, proposed in[12] "Semantic Mapping Between Natural Language Questions and SQL Queries via Syntactic Pairing", the author proposed an automatic translation of natural language query into SQL query using support vector machine algorithm and kernel functions. In this algorithm to design a dataset of relational pairs containing syntax trees of questions and queries and encode them using kernel functions. Some of the functionalities used need the up gradation which is carried out on the current work.

Gauri Rao, Snehal Chaudhry, Nikita KulKarni, Dr. S. H. Patil, proposed in[13] "Natural language processing using semantic grammar", the author proposed the architecture for translating Natural language Query into SQL using semantic Grammar. Lexicon and post preprocessor are used in the semantic analysis. Lexicon that stores all possible words that grammar is aware of. Post preprocessor transforms the semantic representations of the sentence into a SQL query. This system capable of handling simple queries with standard joins conditions but not flexible.

N. D. Karande, and G. A. Patil describes about[14] "Natural Language Database Interface for Selection of Data Using Grammar and Parsing", the authors proposed the NLDBI system considers selection of data and performing primitive queries onto database and join operation with some constraints. ATN (Augmented Transition Network) parser is used for generating a parse tree.

Thus, the existing work and strategies were discussed in detail in this section which provides the sufficient reason to expand and enable the current work. On the whole all the existing work provides the complicated and different type of framework which is not suitable for many cases which holds up the reason for the extension. Therefore, the proposed work is discussed with the help of simple architectural layout and implementation phase in the following sections.

3. A Simple Architectural Layout

Figure 1 illustrates the simple architectural layout of the proposed framework. In this layout the Preprocessing phase consists of four modules such as Morphological analysis, Semantic analysis, Mapping table and Retrieval of reports. Here the Natural Language sentence is given as an input by the user. Morphological analysis involves the following steps.

In Morphological analysis the user gives an NLP sentence as input and it is sent to Tokenizer. The Tokenizer split the sentences into Word based on whitespace character. The tokenized words is taken to extractor for stemming process. In stemming process, the extractor maintains the collection of predefined words which is used for comparison with the incoming new words. Predefined words are most used words in the document for querying. It compares the tokenized words with the predefined and extracts the main keywords. i.e., the keywords are words that are present in the predefined list of words. Then from the extracted words, the root words are identified. In the semantic analysis, the identified set of words will be given as input. The parse tree is generated through parser and subject, object & verb present in the set of words is identified. The output of this analysis will be the collection of identified words. In Mapping, the mapping table consists of predefined set of SQL queries along with the maximum possibility of NLP words. Map the collection of identified words with the mapping table and find the

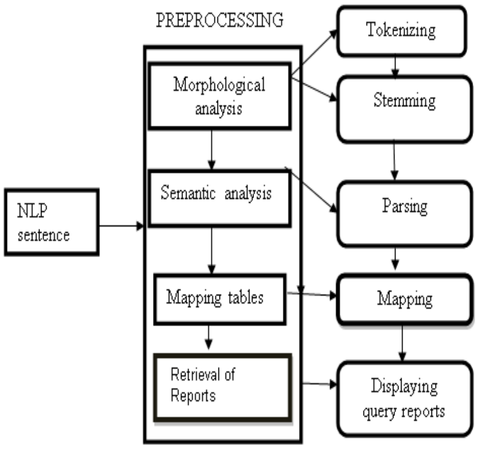


Figure 1. A Simple Architectural Layout.

best suitable query. The sql query is generated at the end as a report from which the query is picked and answered semantically.

4. Implementation and Experimental Results

The proposed framework is implemented by elaborating the actual working aspect of natural language processing analysis which is explained in few steps. Generally NLP has following steps (courtesy[7])

4.1 Morphological Analysis

Individual words are analyzed into their components and non word tokens such as punctuation are separated from the words.

4.2 Syntactic Analysis

Linear sequences of words are transformed into structures that show how the words relate to each other. Some word sequences may be rejected if they violate the languages rules for how words may be combined.

For example An English semantic analyzer would discard the sentence "guy the go the candy shop".

4.3 Semantic Analysis

The structures created by the syntactic analyzer are assigned meanings. In other word mapping is made between syntactic structure and objects in the task domain. Structures for which no such mapping is possible may be rejected.

For example the sentence "monochrome red dreams sleep frantically" would be discarded as semantically anomalous.

4.4 Discourse Integration

The meaning of an individual sentence may depend on the sentences that precede it and may influence the meanings of the sentences that follow it.

For example the word "it" in the sentence "Tom wanted it" depends on the prior dissertation context, while the word "Tom" may manipulate the meaning of later sentence (such as "he always had").

4.5 Pragmatic Analysis

The structure representing what was said is reinterpreted to determine what was actually meant.

For example the sentence "Do you know what day it is?" should be interpreted as request to be told the day. The boundaries between these five phases are often very fuzzy. The phases are sometimes performed in sequence. One may need to appeal for assistance to another. For example part of process of performing the syntactic analysis of the sentence "Is the goblet pot peanut cooking oil?" is deciding how to form two noun phrases out of four nouns at the end of the sentence.

We consider a database SQL Server 2005. We have placed 3 tables in this SQL Server database. Novice users can not access contents of databases as they don't have knowledge of SQL language. That's why we proposed system which will enable user to access contents of databases using simple English language. Suppose we want comment of a facebook whose likes "Flipkart" then we have to form a SQL query: Select comments from facebook where flipkart="like"; For a novice user it is not possible to form SQL query so using our system he/she can simply asks a question like "**What is name and comments of facebook who likes flipkart?"** In our daily life we always use a WH question that's why proposed system easily interprets WH questions and generates its relevant intermediate query. In addition with WH questions our system works with In Which, Total Number Of, On which type questions. Figure 2[7] represents the structure of the system in which the NLP sentence gets processed at the respective levels and refined as a typical sql query. The refined sql query will fetch the desired data from the database which finally mapped to the user.

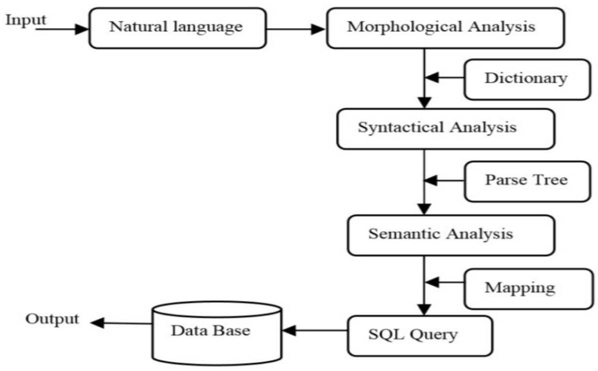


Figure 2. Structure of the System.

Then the Morphological analysis are identified each world like what, is, the, Customer's, Facebook Id. Individual words are analyzed into their components, and it separated noun and adjective in the sentences. Morphological analysis must pull apart the word "Customer's" into the proper noun "Customer" and the possessive suffix' "s". A limited data dictionary is also used to store all related words about the system. After this, Syntactic rules checks the grammatical mistakes of a sentence and Semantic analysis must map individual words into appropriate objects in the knowledge base or database and the meanings of the individual words combine with each other and find out the meaning of simple English query. Example: Meaning of query: Facebook Id of Customer with name Customer. Then the translator will change the above sentence with SQL query and with the help of SQL query, we will able to find out the results. SQL Query: Select custname, comments from facebook where jabong="like";

When user opens system he/she has to establish connection to database and then he/she can fire queries to database. User can ask queries to database in how many, Total number of, in which format in addition with WH formats. Our system also provides facility to update tables in database. User can insert values into tables and can also delete values from table. Our system generates number of intermediate queries depending on semantics of user entered English statement. Users have to select one of intermediate query which is more relevant to users intended query. Then system will generate its appropriate SQL query. Our system also works fine with JOIN. User can retrieve data from two or more columns also.

Firstly system accepts English statement from user then system tokenized that statement and removes unwanted words. After that it identifies synonyms of column names and table names then replace synonyms with actual names. System places tokens in 4 parts depending on criteria words and then properly placing that part generates one or more intermediate statements. This is one part of the system which only generates intermediate query. System simplify decision making task by relying on user for selection of intermediate query. This also helps to system to give proper output to the user and user can also easily recover from mistakes. After user selects intermediate query system's Generate SQL module takes it as input and finds out 3 main keywords i.e. Select keyword, From keyword, Where keyword. Select keyword contains attributes that user wants to retrieve. From keyword contains table name from which user wants to retrieve attributes. From keyword can also contain more than one table then system has to generate query using JOIN as there is relationship between tables. Where keywords contains criteria which helps to retrieve specific contents by placing condition. Then Generate SQL formats all these keywords in specific format and using different conditions that means formatting From keyword is different when there is only one table and different in case of two or more tables where we have to use JOIN. Then it places these keywords in standard SQL query and generates SQL.

"What is name and comments of facebook who likes flipkart" is processed by the system as given below.

Figure 3 illustrates the typical workflow diagram[5] which explains the actual flow of process which is carried out to bring the well refined and appropriate query for fetching from the database. Now we consider some examples and see how system handles them. Assume our database contains two tables Twitter and Facebook in normalized form. Firstly we will take following example:-

What is the comment from Twitter who likes jabong?

Then by processing above English query system generates intermediate query i.e.

What is comment from twitter who likes jabong

Generate SQL modules takes above query as an input and firstly finds out all attributes and table names then by interpreting meaning identify relation between tables and

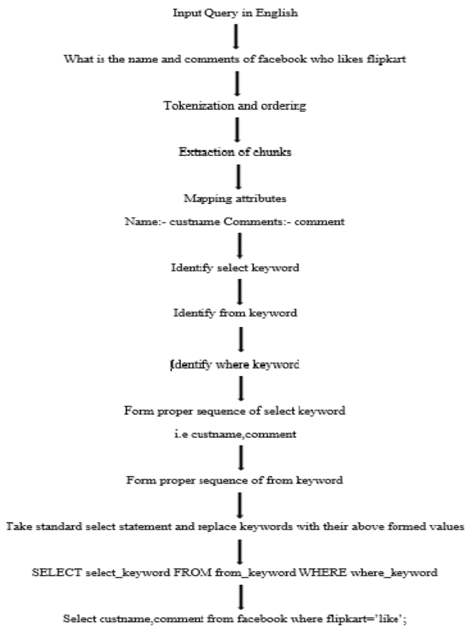

Input Query in English
What is the name and comments of facebook who likes flipkart
Tokenization and ordering
Extraction of chunks
Mapping attributes
Name:- custname Comments:- comment
Identify select keyword
Identify from keyword
Identify where keyword
Form proper sequence of select keyword
i.e custname,comment
Form proper sequence of from keyword
Take standard select statement and replace keywords with their above formed values
SELECT select_keyword FROM from_keyword WHERE where_keyword
Select custname,comment from facebook where flipkart='like';

Figure 3. Workflow diagram.

form query using JOIN condition. Output of Generate SQL module for above query is as follows:-

Select count (comment) from facebook JOIN twitter fbid=tid where flipkart="like";

Analytical sequence of steps followed in the proposed framework:

➢ Process Query

- Divide Query in tokens.
- Remove punctuation marks.
- Do initializations.

➢ Generate Intermediate Analysis for query formation

- Divide Query into parts using criteria words.
- Identify column attributes and table names from user Query and remove unwanted words.
- Replace synonyms of column attributes and table names in Query with its actual names.
- Arrange parts in proper sequence.

➢ Formation of SQL Query

- Take intermediate Query as input

- Identify/ derive 3 things from Query

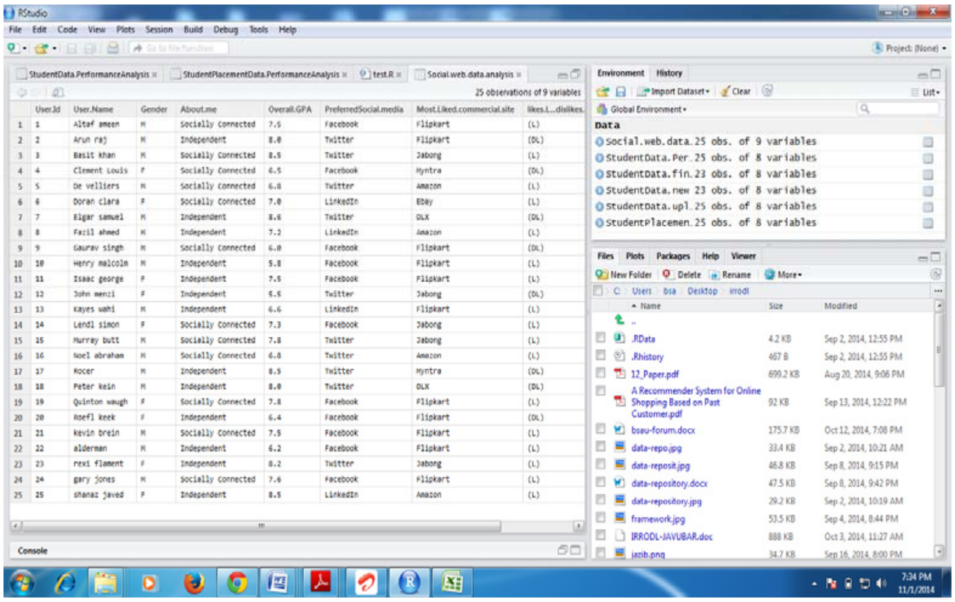
  ▪ Select keyword: - These are attributes which user wants to retrieve.

  ▪ From keyword: - This is the table name from which user want to retrieve data.

  ▪ Where keyword: - This is condition specified in query.

- Replace select keyword with actual table attributes.
- If there is only one from keyword then replace it with actual table name. else form following sequence table name1 JOIN table name2 ON attribute1(primary key of table1) = attribute2(attribute in table2 which is foreign key of table1)
- Replace where keywords attribute with actual table attribute and concatenate„ =" following with value specified by user.
- Form standard template of SELECT Query and substitute above keywords i.e. select keyword, from keyword, and where keyword in their appropriate place.

Figure 4 illustrates the primary data source of social web data which composed of content from various social web sources such as facebook, twitter and linkedIn etc. The users, who surf on the net for most suitable and likely online shopping sites such as flipkart, amazon, jabong etc., were extracted and kept apt for query processing and analysis. The web data is pulled with the help of R-tool which acts as a data extractor. The extracted data is treated with the NLP interface which is developed and respective the content is collected, analyzed and verified for the semantic nature of the absolute social web data.


RStudio
File Edit Code View Plots Session Build Debug Tools Help
Go to file/function
Project: (None)
StudentData.PerformanceAnalysis
StudentPlacementData.PerformanceAnalysis
test.R
Social.web.data.analysis
Environment History
Import Dataset
Clear
List
Global Environment
Files Plots Packages Help Viewer
New Folder Delete Rename More
C Users bsa Desktop irrodl
Console
7:34 PM
11/1/2014

25 observations of 9 variables

	User.Id	User.Name	Gender	About.me	Overall.GPA	PreferredSocial.media	Most.Liked.commercial.site	likes.L...dislikes.
1	1	Altaf ameen	M	Socially Connected	7.5	Facebook	Flipkart	(L)
2	2	Arun raj	M	Independent	8.0	Twitter	Flipkart	(DL)
3	3	Basit khan	M	Socially Connected	8.5	Twitter	Jabong	(L)
4	4	Clement Louis	F	Socially Connected	6.5	Facebook	Myntra	(DL)
5	5	De velliers	M	Socially Connected	6.8	Twitter	Amazon	(L)
6	6	Doran clara	F	Socially Connected	7.0	LinkedIn	Ebay	(L)
7	7	Elgar samuel	M	Independent	8.6	Twitter	OLX	(DL)
8	8	Fazil ahmed	M	Independent	7.2	LinkedIn	Amazon	(L)
9	9	Gaurav singh	M	Socially Connected	6.0	Facebook	Flipkart	(DL)
10	10	Henry malcolm	M	Independent	5.8	Facebook	Flipkart	(L)
11	11	Isaac george	F	Independent	7.5	Facebook	Flipkart	(L)
12	12	John menzi	F	Independent	5.5	Twitter	Jabong	(DL)
13	13	Kayes wahi	M	Independent	6.6	LinkedIn	Flipkart	(L)
14	14	Lendl simon	F	Socially Connected	7.3	Facebook	Jabong	(L)
15	15	Murray butt	M	Socially Connected	7.8	Twitter	Jabong	(L)
16	16	Noel abraham	M	Socially Connected	6.8	Twitter	Amazon	(L)
17	17	Rocer	M	Independent	8.5	Twitter	Myntra	(DL)
18	18	Peter kein	M	Independent	8.0	Twitter	OLX	(DL)
19	19	Quinton waugh	F	Socially Connected	7.8	Facebook	Flipkart	(L)
20	20	Roefl keek	F	Independent	6.4	Facebook	Flipkart	(DL)
21	21	kevin brein	M	Socially Connected	7.5	Facebook	Flipkart	(L)
22	22	alderman	M	Independent	6.2	Facebook	Flipkart	(L)
23	23	rexi flament	F	Independent	8.2	Twitter	Jabong	(L)
24	24	gary jones	M	Socially Connected	7.6	Facebook	Flipkart	(L)
25	25	shanaz javed	F	Independent	8.5	LinkedIn	Amazon	(L)

```
Data
Social.web.data 25 obs. of 9 variables
StudentData.Per 25 obs. of 8 variables
StudentData.fin 23 obs. of 8 variables
StudentData.new 23 obs. of 8 variables
StudentData.upl 25 obs. of 8 variables
StudentPlacemen 25 obs. of 8 variables
```

Name	Size	Modified
..		
.RData	4.2 KB	Sep 2, 2014, 12:55 PM
.Rhistory	467 B	Sep 2, 2014, 12:55 PM
12_Paper.pdf	699.2 KB	Aug 20, 2014, 9:06 PM
A Recommender System for Online Shopping Based on Past Customer.pdf	92 KB	Sep 13, 2014, 12:22 PM
bsau-forum.docx	175.7 KB	Oct 12, 2014, 7:08 PM
data-repo.jpg	33.4 KB	Sep 2, 2014, 10:21 AM
data-reposit.jpg	46.8 KB	Sep 8, 2014, 9:15 PM
data-repository.docx	47.5 KB	Sep 8, 2014, 9:42 PM
data-repository.jpg	29.2 KB	Sep 2, 2014, 10:19 AM
framework.jpg	53.5 KB	Sep 4, 2014, 8:44 PM
IRRODL-JAVUBAR.doc	888 KB	Oct 3, 2014, 11:27 AM
jazib.png	34.7 KB	Sep 16, 2014, 8:00 PM

Figure 4. Screenshot of Social web data for knowledge extraction & analysis.

Figure 5 describes the design of the translator. The translator will accept NL sentence as input. When submit button pressed, it performs the translation of Natural language sentence into SQL. The reset button resets the window for next query. The interface processes the given lingual statement and prepares the given statement to process the given sentence.

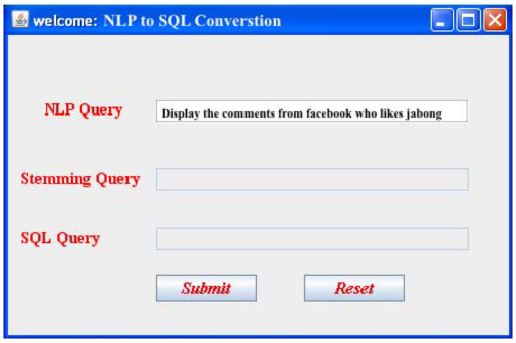
Figure 6 describes the stemming process. This is done by using Porter algorithm. The extractor maintains the collection of predefined words which is used for comparison with the incoming new words. Predefined words are most used words in the document for querying. It compares the tokenized words with the predefined and extracts the main keywords. i.e., the keywords are words that are present in the predefined list of words. Stemming process used to identify the root word and remove the unwanted words in the given input. It will give the important keyword of the NLP sentence.

Figure 7 displays the suitable SQL query for the Natural language sentence. It also displays the important keywords in the given input NLP sentence. The unwanted prepositions are removed the most suitable and apt word are processed as the primary query to trigger the data from the database.

Figure 8 illustrates the second type of input, here the user is giving NLP query for removing records from the table. When user clicks the submit button Then it will delete the appropriate records from the table and it will display the entire details except the removing details.

5. Analysis of Generated SQL

The Generated the SQL statement from the natural lingual statement is analyzed to the large extent to measure the semantic levels of the knowledge extracted, gained and attained. The sample query used in the above experimental study is analyzed with the help of common evaluation measures such as Recall and Precision.

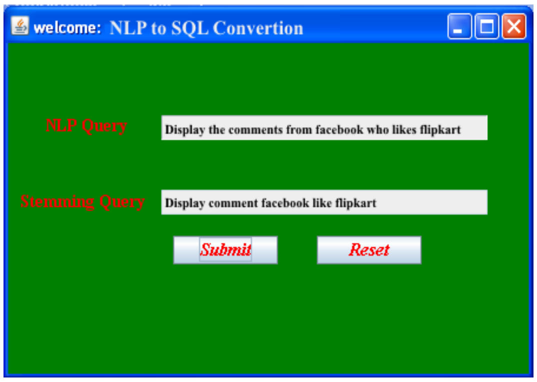


NLP Query: Display the comments from facebook who likes jabong

Stemming Query:

SQL Query:

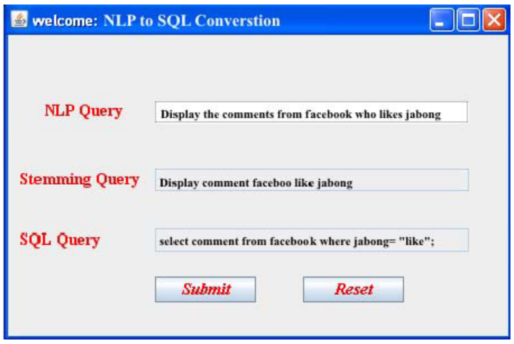
Figure 5. Screenshot For translator.



NLP Query: Display the comments from facebook who likes flipkart

Stemming Query: Display comment facebook like flipkart

Figure 6. Screenshot for stemming process.

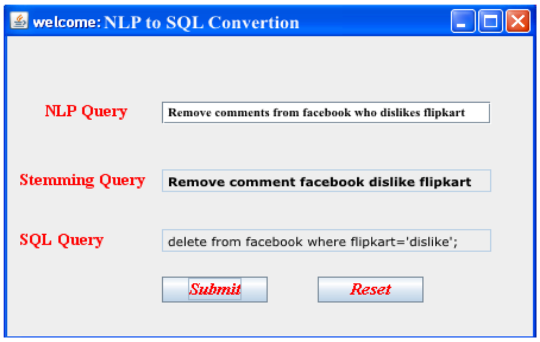


NLP Query: Display the comments from facebook who likes jabong

Stemming Query: Display comment faceboo like jabong

SQL Query: select comment from facebook where jabong= "like";

Figure 7. Screenshot for displaying SQL query.



NLP Query: Remove comments from facebook who dislikes flipkart

Stemming Query: Remove comment facebook dislike flipkart

SQL Query: delete from facebook where flipkart='dislike';

Figure 8. Screenshot for removing records from the table.

5.1 Recall

A measure of the ability of a system to present all r
words.

$$\text{recall} = \frac{\text{number of relevant words retrieved}}{\text{number of relevant words in sentenc}}$$

5.2 Precision

A measure of the ability of a system to present o
evant words.

$$\text{precision} = \frac{\text{number of relevant words retrieve}}{\text{total number of word retrieved}}$$

Precision and recall are set-based measures. T
they evaluate the quality of an unordered set of retrieved query. To evaluate ranked lists, precision can be plotted against recall after each retrieved semantic query. To assist computing average presentation over a set of selected domain each with a different number of relevant documents individual topic precision values are interpolated to a set of standard recall levels (0 to 1 in increments of .1). The particular rule used to interpolate precision at standard recall level i is to use the maximum precision obtained for the topic for any actual recall level greater than or equal to i. Note that while precision is not defined at a recall of 0.0, this interpolation rule does define an interpolated value for recall level 0.0. The example takes up the query used in the above experiment i.e., the query based on social media enquiry "Display the comments from facebook who likes jabong" these query is measured with the recall and precision and the graph is plotted for the total. The actual threshold range of precision and recall is plotted and illustrated in the Figure 9 which displays the suitable range of word occurrence in the generated query. The values are measured precisely and generated as report in the form of graphical chart which is illustrated in the Figure 10. The Recall and Precision level table is generated which provides the exact evidence for evaluating the semantic measure of the collected the social web data which is gathered out of a user query in terms of natural lingual statement.

6. Conclusion and Future Enhancement

These Experimental studies emphasize the need of knowledge extraction technique in the new and trending

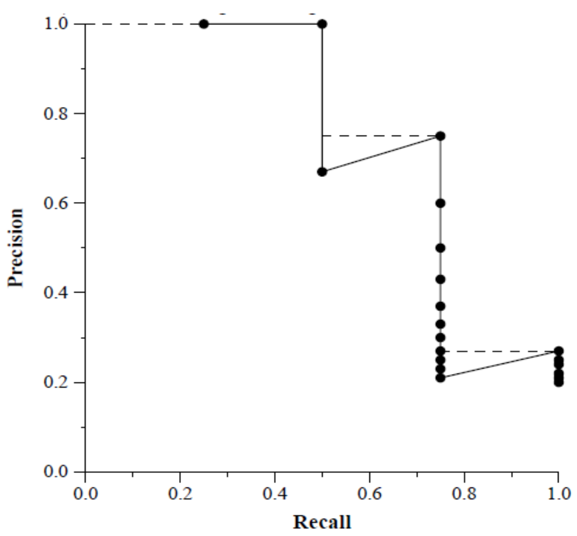


Figure 9. Interpolation of precision and recall threshold range.

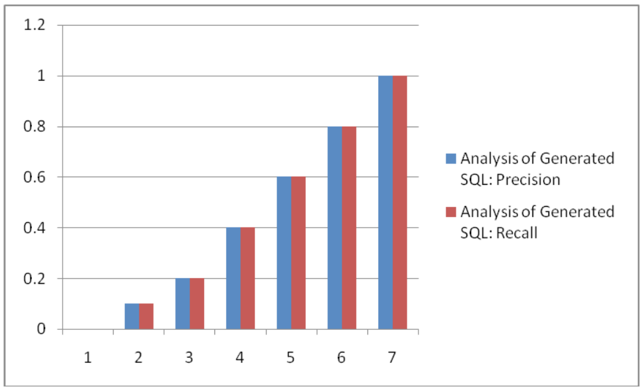


Figure 10. Analysis of generated SQL with precision & recall threshold range.

Table 1. Recall & precision table for generated SQL

Recall Level Precision Averages	
Recall	Precision
0.00	0.5349
0.10	0.4789
0.20	0.4345
0.30	0.3790
0.40	0.2491
0.60	0.1284
0.80	0.0023
1.00	0.0014

domain of social media. In this Research work a system is developed that is able to execute both DDL and DML queries, input by the user in his/her natural language (English). The conversion from natural language to SQL is done precisely with the help of trending technology which is used in this paper. The experimental study and analysis conducted in this research work proves to be the essential factor in identifying the knowledge extraction technique in the field of social media. The system is developed in java programming language and various tools of java are used to build the system. An oracle database is used to store the information's. Input given by the user is not required in the form of questions (who-form like what, who, where, etc.). A limited Data Dictionary is used where all possible words related to a particular system are included. The Data Dictionary of the system must be regularly updated with words that are specific to the particular system. Ambiguity among the words will be taken care of while processing the natural language. The results show that our software is correct and handle the SQL Queries without any problem. The analysis shows that, there is much more room is available for improvement in finding the exact and semantic content from the social web sources which can effectively used by the large group of heterogeneous users. Future researches should consider factors that lead users to reformulate their NLP sentence and should try with different collection web sources with various nature of dataset. Also new research should be done to gather more information in various levels of understanding, effectiveness and situations. Method of gathering information from multiple tables should be carrying forward in the future.

7. References

1. Satav AG, Ausekar AB, Bihani RM, Shaikh A. A Proposed Natural Language Query Processing System. International Journal of Science and Applied Information Technology. 2014; 3(2).
2. Kaur G. Usage of Regular Expressions in NLP. IJRET. 2014; 3(1).
3. Gaikwad MP. Natural Language Interface to Database. IJEIT. 2013; 2(8).
4. Kaur J, Chauhan B, Korepal JK. Implementation of Query Processor Using Automata and Natural Language Processing. International Journal of Scientific and Research Publications. 2013; 3(5).
5. Bhadgale AM, Gavas SR, Patil MM, Pinki R. Natural Language To Sql Conversion System. IJCSEITR. 2013 Jun; 3(2):161–6. ISSN 2249-6831.
6. Agrawal AJ, Kakde OG. Semantic Analysis of Natural Language Queries Using Domain Ontology for Information Access from Database. IJISA. 12:81–90.
7. Kaur S, Bali RS. Sql Generation and Execution from Natural Language Processing. International Journal of Computing & Business Research ISSN (Online): 2012; 2229–6166.
8. Deshpandel AK, Prakash R. Natural Language Processing using probabilistic context free grammar, International Journal of Advances in Engineering & Technology, Devale, Department of Information Technology, Bharati Vidyapeeth Deemed University, Pune, India. 2012; 3(2):568–73.
9. Tamrakar D, Dubey D. Query Optimization using Natural Language Processing, Dept. of CSE, Chhatrapati Sivaji Institute of Technology, IJCST, CG, India 2012; 3(1).
10. Gage M. A Survey of Natural Language Processing Techniques for the Simplification of User Interaction with Relational Database Management Systems, California Polytechnic State University, San Luis Obispo. 2012.
11. Nihalani N, Silakari S, Motwani M. Natural language Interface for Database: A Brief review. IJCSI. 2011; 8(2).
12. Giordani A, Moschitti A. Semantic Mapping between Natural Language Questions and SQL Queries via Syntactic Pairing, Department of Computer Science and Engineering University of Trento via Sommarive 14, 38100 POVO (TN) – Italy. 2010
13. Chaudhry GRS, KulKarni N. Natural language processing using semantic grammar. IJCSE. 2010; 2(2):219–23.
14. Karande ND, Patil GA. Natural Language Database Interface for Selection of Data Using Grammar and Parsing. World Acad Sci Eng Tech. 2009; 3:11–26.