Intelligent Decision Technologies -1 (2019) 1–16 1
DOI 10.3233/IDT-180074
IOS Press

A novel Bengali Language Query Processing System (BLQPS) in medical domain

Kailash Pati Mandal[a,*], Prasenjit Mukherjee[a], Baisakhi Chakraborty[a] and Atanu Chattopadhyay[b]
[a]*Department of Computer Science and Engineering, National Institute of Technology, Durgapur, India*
[b]*Department of BBA (H) and BCA (H), Deshabandhu Mahavidyalaya, Chittaranjan, India*

Abstract. Bengali is the seventh most widely spoken language in the world. Many researchers are working on developing Bengali language based information retrieval, question-answering, query-response systems. The proposed Bengali Language Query Processing System (BLQPS) is based on natural language query-response model. Bengali language has been used in the model to extract knowledge data from a default database. The system is based on scoring and pattern generation algorithm that is able to generate structure query language (SQL) from natural language query in Bengali with the help of a synonym database. The proposed system is domain based and a large number of words have been initialized in the synonym database. The SQL is formulated from semantic analysis. Further, the generated SQL has been used to extract knowledge data in Bengali language from the default database.

Keywords: Query-response, scoring and pattern generation based algorithm, Structure Query Language (SQL), Semantic analysis, Bengali Language Query Processing System (BLQPS), Natural Language Processing (NLP)

1. Introduction

The 21st century is the current century of human-computer interaction era. The main aim of this discipline has to interact with computerized system using less effort. The Natural language processing (NLP) plays a very important role in human-computer interaction. The humans can understand natural language whereas computerized system can understand machine understandable language. As a result the naive user is not able to access the computerized system by their native language. The NLP is a technique which converts the human understandable language to a machine understandable language form. As a result the naive user can be able to access the computerized system by their native language without knowing the details of conversion technique. There is a substantial amount of work that has already been done on natural language interface to database. Different researchers have applied different techniques. The conversion of natural language to SQL can be done through Morphological Analysis, Syntactic Analysis, Semantic Analysis, Discourse integration and Pragmatic Analysis [3]. Some researchers have proposed the Natural Language Query Processing (NLQP) system as an interface to database system using semantic grammar [7]. A Hindi language based graphical user interface for transport system is developed using a set of predefined rules [8]. Another Hindi language interface for database based on karaka theory generates SQL by comparing each token with knowledge base [9]. There are many government awareness campaigns (Health, Education etc.) undertaken by government in various portals or e-platforms in English which majority of the citizens may not be able to access or understand due to language barriers as mostly Indians speaking vernacular languages are not comfortable in English. Bengali is one of the vernacular languages. Bengali language has been used in important states of India such as West Bengal, Tripura, Assam and Andaman Nicobar Islands. The national language of Bangladesh is also Bengali. The proposed system

*Corresponding author: Kailash Pati Mandal, Department of Computer Science and Engineering, National Institute of Technology, Durgapur, West Bengal, India. Tel.: +91 7407367323; E-mail: biltu.cse@gmail.com.

is a query response model in the medical domain that shall able to process medical related query where the query is accepted in Bengali. The system is aimed at overcoming language barrier. The system's synonym database consists of two tables, one is entities another is attributes table. The default database consists of three tables. These tables are hospital, doctor and department table. The user can post a query in Bengali language. The parts of Speech (POS) tagging is done by scoring method. Then the system generates all possible patterns of unknown words. This generated pattern is compared with synonym database and populated in the semantic table for semantic analysis. This semantic analysis helps to construct the SQL of default database. Finally formal SQL is executed by system and fetch the desired result. No Adjective is used in the proposed system architecture pertaining to medical domain because no qualitative query shall be posted in the system. This is the limitation or constraint of the system.

2. Related works

Substantial amount of work has been done since last few decades on natural language processing. The Review on Natural Language Processing research papers addresses the challenges between natural language and computing device. The natural language processing applications are based on Phonology, Morphology, Semantics and Pragmatics. Phonology depends on sound of speech of speaker. Morphology is the structural study of word that locates the root word. The Semantic analysis expresses the textual meaning of the sentence without context. The Pragmatic analysis expresses the meaning of the sentence with context. Natural language processing (NLP) is a field of study of interaction with computer by using human language. A wide range of NLP based system has been developed by using mathematical and computational modeling of various aspects of language. NLP is a technique through which the computing device can understand natural language text or speech. Some NLP based applications are machine translation, natural language text processing and summarization, user interfaces, multilingual and cross language information retrieval (CLIR), speech recognition, artificial intelligence (AI) and expert systems are discussed in [1]. Interface between natural language and database is also a hot topic of research. This intelligent interface is designed for naive user who does not have any knowledge of database. An intelligent interface for relational database system converts the English language query to SQL using semantic matching, data dictionary and a set of production rules together has been defined in [2]. Conversion of human language to a formal language like SQL through different phases of analysis like Morphological Analysis, Syntactic Analysis, Semantic Analysis, Discourse integration and Pragmatic Analysis has been done in paper [3]. The Prolog programming language based question answering system named Chat-80 internally represents the meaning of English questions by a set of Prolog programming logic. Finally the answer is fetched by executing the Prolog logic as discussed in [4]. The EFLEX system is an efficient database interface system that consists of analyzer, mapper and translator. The analyzer interprets the given natural language query for the mapper. The mapper maps the natural language to its corresponding SQL. Finally the translator forms the query correctly. The efficiency of the EFLEX system has been improved by using Knuth-Morris-Pratt algorithm that is explained in [5]. The Knowledge Management System (KMS) is query response tool where the user can post the query in English language into the KMS. Then the KMS retrieves the data from default database using a set pre-defined grammar rules and semantic analysis as discussed in [6]. The NLQP as in [7] reduces extra overhead of complex SQL. This NLQP consists of four modules. These modules are Analyzer Module, Parser Module, Query Builder Module and code optimizer Module. The Analyzer Module tokenizes the English language query into keys after which these tokenized keys are sent to the Parser Module and then the Parser Module combines these tokens and performs syntactic analysis. The Query Builder Module forms the SQL query using parsing information. Finally the Code Optimizer Module fetches the data in efficient way is implemented in [7]. A Hindi language based interface for transport system is developed for native Hindi speaker where the data is retrieved from Hindi database using Hindi language query as in [8]. In this system, SQL statements like insert, update, delete, MIN(), MAX(), SUM() and AVG() are implemented in [8]. In [9], a Hindi language interface for database using Karaka theory has been developed which is very useful for native Hindi users. The proposed system divides the Hindi Language query into number of tokens. The shallow parser removes useless tokens. The Case Solver forms a new Hindi language query which is consists of base words. The shallow parser uses POS type verb to determine the SQL command. The token which is present before the case symbol is treated as a table

name and the token which present after the case symbol is treated as attribute name. Some condition start tokens or words as mentioned in Section 3(d) of [9] are present in the proposed system which has helped to construct the condition part of the SQL query. Then the graph generator represents the relationship among command, table name, attribute name and conditional part. The query translator converts Hindi token to its corresponding English token using knowledge base. Finally Query Executor executes the SQL query and retrieves desired data from database that has been discussed in [9]. The Natural Language Interface to the Database (NLIDB) based on ontology as in [10] has produced better result than any other existing NLIDB. In this system, the semantic representation is done by using Ontology Web Language (OWL) for knowledge modeling which increases the correct responses of user's query that is explained in [10]. In [11] deals with structural ambiguity of a Bengali sentence. The given sentence is tokenized by the Tokenizer. Then the validator checks grammatical mistake in the sentence. Whereas the En-converter converts the given sentence into a Universal Networking Language expression using Dictionary Entry-lookup, rules of morphological analysis and semantic analysis. This technique has been discussed in [11]. The proposed system identifies the Bangla grammar using predictive parser. In this system the parser uses the top down technique. The proposed system uses the pre-defined XML dictionary for parts of speech tagging. The access time of XML file is much lesser than other file format. The parse table is generated using context free grammar in this system. The proposed parser has been discussed in [12]. The Syntax Analysis and Machine Translation of Bangla Sentences system in [13] can able to convert all types of Bangla sentence to English sentence using pre-defined grammar rules. The parsing is done in this proposed system through different steps. The system tokenizes the user given Bengali sentence. Then the system counts the number of tokens. After that the proposed system checks given sentence's length and pre-defined rules length. If both the length is matched then corresponding phrases is retrieved and parse tree is generated. The Bangla to English converter converts the Bangla sentence to English sentence by using training corpus which selects the word to form the sentence which probability is to maximum. The proposed system is discussed in [13]. The Rule Based Bengali Stemmer is a proposed system which derives the Bengali root word by removing the affix from a given word. The proposed system categorizes all words in two parts either verbal affix or nominal affix. The Rule Based Bengali Stemmer checks every letter of a word. If affix is present then remove the affix and find the stem word. The above mentioned process uses for both verbal inflection and nominal inflection. The proposed system is implemented in [14]. The proposed system tokenizes the given English language's query. Then all tokens are passes through automata. Automata remove articles, connectors and extract correct pattern. Automata substitute the keyword with proper attribute. Finally automata map the value which corresponds to a particular attribute as well as a table also. If the two or more tables are associated with the query then the automata joins up the tables. In these way automata build up the SQL from the English language query is discussed in [15]. In [16], tense based English to Bangla translation system has been implemented which convert the English sentence into corresponding Bangla sentence. The proposed system verifies the syntactical correctness using context-free grammar whereas bottom up approach is used to generate parse tree for the given sentence. The proposed system consists of Tokenizer, Syntax Analyzer, Grammatical Rule Generator, Lexicon, Parse Tree and Conversion Unit. Tokenizer tokenizes the given sentence and sends to the Syntax Analyzer. Then the Syntax Analyzer compares each token with lexicon. If the token does not match with lexicon then token is invalid otherwise find out POS type and Bengali meaning of the token. The Grammatical Rule Generator contains a set of pre-defined production which useful to form a correct parse tree. The Conversion Unit converts the English parse tree into corresponding Bengali parse tree. Finally Bengali sentence prints. This system is discussed in [16]. Link data, SPARQL language and interface in natural language may be an interesting solution for accumulate and disseminate biomedical knowledge in biomedical area. Researchers may have difficulty to handle SPARQL language where natural language interface helps to life science researchers for extraction biomedical knowledge from biomedical knowledge base as in [21]. Non-expert users don't have ability to access huge data repository. Natural language interface to web data services are working as an immerging technology to non-expert users for accessing huge data repository has been discussed in [22]. Natural language interface are crucial part of semantic knowledge representation system where understanding of formal representation language to model in a particular domain is a difficult task for users. Authors have introduced a semantic wiki system that is based on controlled natural language interface that uses attempto controlled English in

grammatical frame work. The grammatical frame work helps to manage other natural language (multi lingual) queries as in [23]. Lot of research work has already been done in parsing, parts of speech tagging, stemming, sentiment analysis in vernacular languages like Hindi, Bengali Assamese. But very little amount research work has been done on query processing in Bengali natural language. So, there is a need to develop query systems in local or vernacular languages. Sine majority of users in India are living in rural or sub-rural areas are not very comfortable with English language, such systems may help the rural or backward areas to use such systems with vernacular language interface to handle queries. It has been found that rural areas mostly require information in domains like medical fields, education, agricultural information etc. This work discusses on a Bengali Language Query Processing System on medical domain as to aid or facilitate the rural people or people in backward areas to access medical information pertaining to their village/district/state. The rest of the research paper is organized as follows. The Section 2 discusses the literature reviews on related works. The Section 3 gives the architecture of the BLQPS. The Section 4 explains methodology and tools used. Then the Section 5 discusses general features based comparative study of the proposed system with similar type system. The Section 6 discusses conclusion and future works of the proposed system.

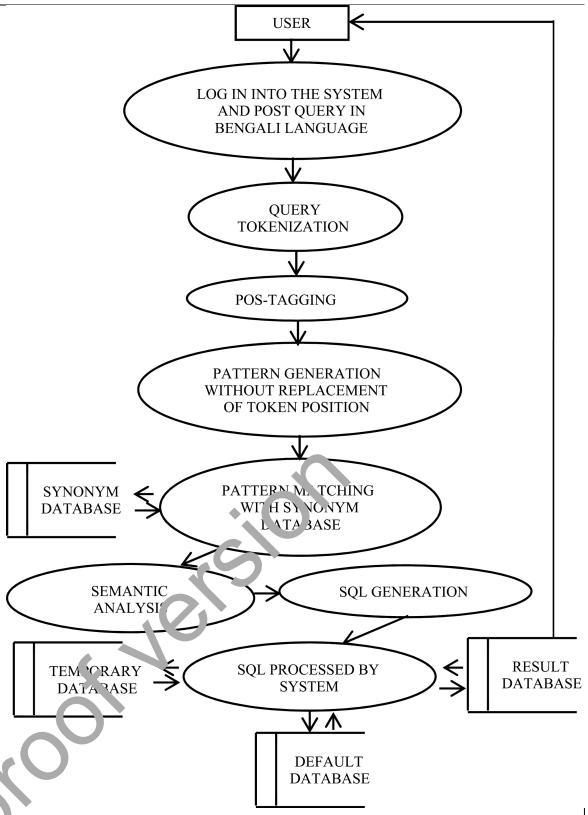

Fig. 1. Data flow diagram of BLQPS.

3. Architecture of the BLQPS

The architecture of the proposed system has been given in Fig. 1.

3.1. Algorithm

The block diagram of algorithmic step has been given in Fig. 2.

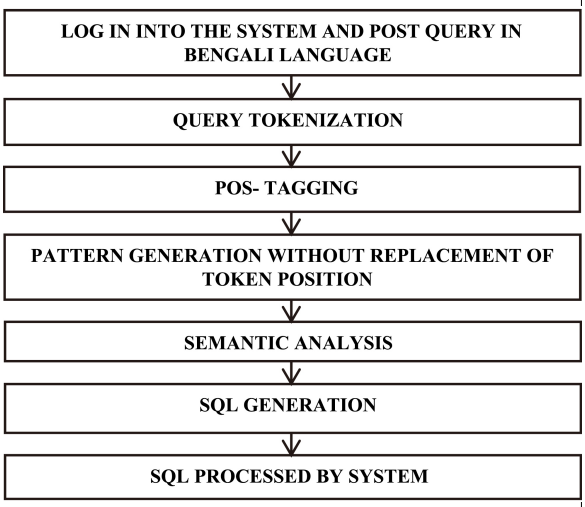

Fig. 2. Natural language query to SQL generation steps.

3.1.1. Log in into the system and post query in Bengali language

The user will log in into the proposed system and will post the query in Bengali language. A query has been given below as an example.

নদিয়া ও পুরুলিয়ায় কি কি হাসপাতাল আছে?

(nothyea o purulyeae ki ki haspathal ache?)

That means in English Language

What are hospitals available in Nadia and Purulia?

3.1.2. Query tokenization

The proposed system reads the query and slices it into meaningful linguistic units called tokens after removal of punctuation marks. These tokens will be stored in a string array. The pictorial representation of user given query after tokenization has been given in Table 1.

Table 1
Query String array for NL query after tokenization

Array index	0	1	2	3	4	5	6
The string array of given query	নদিয়া (nothyea)	ও (o)	পুরুলিয়ায় (purulyeae)	কি (ki) (What)	কি (ki) (What)	হাসপাতাল (haspathal) (Hospital)	আছে (ache) (Available)

Table 2
Score array with score value after POS tagging

Array index	0	1	2	3	4	5	6
Score array	1	5	1	2	2	1	1

3.1.3. POS-tagging

The BLQPS contains few stop words which are predefined list of Bengali words like interrogative words, pronouns, prepositions and conjunctions. These predefined Bengali words (stop words) have been stored in string arrays. The pre-defined words are termed as known words and remaining all other words are treated as unknown words. A numeric value has been assigned to each string array that contains predefined words termed as score. The score of string array of interrogative words, pronouns, preposition and conjunction are 2, 3, 4 and 5 respectively. The score of all unknown words are 1. The pre-defined string arrays have been given below with score.

i. list_interrogative[ ] = {কি (ki) (What), কোথায় (kothae) (Where), কেন (keno) (Why)…} and score is 2.
ii. list_pronoun[ ] = {আমি (ami) (I), আমরা (amra) (We), আমাদের (amader) (Our)…} and score is 3.
iii. list_preposition[ ] = {দ্বারা (dara) (By), উপরে (upre) (Above), নিচে (niche) (Under)…} and score is 4.
iv. list_conjunction[ ] = {এবং (ebong) (And), ও (o) (And), আর (ar) (And)…} and score is 5.

When a query is posted in Bengali language, it is first tokenized. The tokens may contain predefined words and unknown words. The tokens are placed in a query string array. The unknown words have a score of 1. Each of the known tokens of the query string array will be compared with predefined words stored in predefined string arrays as mentioned in List i through iv, wherein, list i corresponding to interrogative has score 2, list ii corresponding to list pronoun has score 3, list iii corresponding to list preposition has score 4 and list iv corresponding to list conjunction has score 5. If the corresponding token of query string array matches with any predefined word in predefined string array (list i through iv), then score of corresponding string array will be assigned to the token of the query string that is compared to the string array containing predefine words list as in i through iv). Thereby, a score array (as shown in Table 2) is derived from the query string array(shown in Table 1). The score array length is same as token array. The BLQPS maintains this score array to keep track the score of all token(s) after POS tagging.

The BLQPS selects every token from the tokenized query and compares with each word of interrogative's word list. If token matches with any word of interrogative word's list then the score of the selected token will be same as score of the interrogative's word list i.e. 2. Otherwise token will be compared with each word of pronoun's list. If token matches with any word of pronoun's list then the score of the selected token will be same as score of the pronoun's list i.e. 3. Otherwise in the similar way token will be compared with preposition's list and conjunction's list respectively. If token matches then corresponding array list score will be assigned. If the selected token does not match with any of the above mention known word list, then the proposed system will determine that the selected token is unknown and score will be 1. Using above mentioned procedure the proposed system will assign a specific score in the score array after POS tagging. The score value of first token from token array will be assigned at 0^{th} index in score array. The next score value of second token from token array will assigned at 1^{st} index in score array and other score value of token(s) or word(s) will be assigned at the position of score array as per their array indexing number in token array. The proposed system simply ignores known token(s) or word(s) by identifying score value other than 1. The BLQPS keeps track of all the consecutive and nonconsecutive unknown words which are a very useful component for pattern generation. The user given query after POS-tagging and score value calculation has been given in Table 2.

3.1.4. Pattern generation without replacement of token position

In this step, the BLQPS generates all possible patterns of unknown tokens. The main objective of pattern generation is composition creation of two or more than two unknown words. The proposed system recognizes as unknown token whose score is 1. The unknown tokens or words which are not consecutive or which are separated by another known token or word

Table 3
Semantic table

Id	Entity_name	Attribute_name	Primary_key	Foreign_key	Candidate_key	Value
1	Hospital	hos_district	hos_id			নদিয়া (nothyea)
2	Hospital	hos_district	hos_id			পুরুলিয়ায় (purulyeae)
3	Hospital		hos_id			

Table 4
Desired result after processing NL query

Hos_id	Hos_name	Hos_add	Hos_district	Hos_state
2	নদিয়া জেলা হাসপাতাল (Nadia District Hospital)	চাপরা (Chapra)	নদিয়া (nothyea)	পশ্চিমবঙ্গ (West Bengal)
3	পুরুলিয়া জেলা হাসপাতাল (Purulia District Hospital)	রঘুনাথপুর (Raghunathpur)	পুরুলিয়া (purulyeae)	পশ্চিমবঙ্গ (West Bengal)

Table 5
Entities table of synonyms database

Entity_id	Entity_name	Synonyms	Primay_key	Foreign_key	Candidate_key
1	Hospital	হাসপাতাল (Hospital)	hos_id	NULL	NULL
2	Hospital	বৈদ্যশালা (Hospital)	hos_id	NULL	NULL
3	Hospital	ক্লিনিক (Hospital)	hos_id	NULL	NULL
4	Hospital	স্বাস্থ্যনিবাস (Hospital)	hos_id	NULL	NULL
5	Hospital	আরগ্যশালা (Hospital)	hos_id	NULL	NULL
6	Hospital	আরগ্যনিকেতন (Hospital)	hos_id	NULL	NULL
7	Department	বিভাগ (Department)	dept_id	hos_id	NULL
8	Department	শাখা (Department)	dept_id	hos_id	NULL
9	Department	অংশ (Department)	dept_id	hos_id	NULL
10	Department	দপ্তর (Department)	dept_id	hos_id	NULL
11	Doctor	ডাক্তার (Doctor)	doc_id	hos_id	dept_id
12	Doctor	চিকি□সক (Doctor)	doc_id	hos_id	dept_id
13	Doctor	বৈদ্য (Doctor)	doc_id	hos_id	dept_id

shall have only one pattern. But, when the unknown tokens are consecutive and not separated by any known token or word, they shall generate pattern(s) of tokens or words, where the order of the tokens or words will not be changed in the pattern. The known token(s) or word(s) will not be considered for pattern generation because the proposed system will generate the semantic table using unknown token(s) or word(s).

The query নদিয়া ও পুরুলিয়ায় কি কি হাসপাতাল আছে? (nothyea o purulyeae ki ki haspathal ache?) (What are hospitals available in Nadia and Purulia?) has been tokenized and shown in Table 1. After POS tagging, the BLQPS maps known and unknown token or word by considering index of token array and score array as well as score value of the score array. The nonconsecutive unknown word generates only one pattern. Here নদিয়া (nothyea), পুরুলিয়ায় (purulyeae) are two nonconsecutive unknown words because there is a known word ও between them. So the token নদিয়া (nothyea) will be the single pattern. Similarly the token পুরুলিয়ায় (purulieae) will be another single pattern. But two consecutive unknown tokens or words are হাসপাতাল (haspathal) and আছে (ache) (Available). So these two consecutive unknown tokens shall generate more than one pattern. The token order occurrence is maintained in the generated pattern which is made up of two or more than two tokens. The generated pattern হাসপাতাল (haspathal) আছে (ache) (Available) is made up of two tokens. The token হাসপাতাল (haspathal) occurs before the token আছে (ache) (Available). That why the pattern হাসপাতাল আছে (haspathal) (ache) (Hospital Available) will be generated and the pattern আছে (ache) (Available) হাসপাতাল (haspathal) will not be generated by the BLQPS. All generated pattern has been given below.

i. নদিয়া (nothyea)
ii. পুরুলিয়ায় (purulieae)
iii. হাসপাতাল (haspathal) (Hospital)
iv. আছে (ache) (Available)
v. হাসপাতাল আছে (haspathal) (ache) (Hospital) (Available)

3.1.5. Semantic analysis

The BLQPS contains two databases. One is synonym database and other is default database. The synonym database contains entities and attributes tables. Table 5 represents entities table and Table 6 represents attributes table. The default database contains Hospital

Table 6
Attributes table of synonyms database

Attribute_id	Entity_ name	Attribute_name	Synonyms	Primary_ key	Foreign_ key	Candidate_key
1	hospital	hos_id	হাসপাতালের প্রমাণপত্র (Hospital's Identity)	hos_id		
2	hospital	hos_name	হাসপাতালের নাম (Hospital's Name)	hos_id		
3	hospital	hos_add	হাসপাতালের ঠিকানা (Hospital's Address)	hos_id		
4	hospital	hos_district	হাসপাতালের জেলা (Hospital's district)	hos_id		
5	hospital	hos_state	হাসপাতালের রাজ্য (Hospital's state)	hos_id		
6	doctor	doc_id	ডাক্তারের নম্বর (Doctor's Identity)	doc_id	hos_id	dept_id
7	doctor	doc_name	ডাক্তারের নাম (Doctor's Name)	doc_id	hos_id	dept_id
8	doctor	doc_qualification	যোগ্যতা (Qualification)	doc_id	hos_id	dept_id
9	doctor	doc_specialist	বিশেষজ্ঞ (Specialist)	doc_id	hos_id	dept_id
10	department	dept_id	বিভাগ নম্বর (Department's Identity)	dept_id	hos_id	
11	department	dept_name	বিভাগ নাম (Department's Name)	dept_id	hos_id	

Table 7
Hospital table of default database

Hos_id	Hos_name	Hos_add	Hos_district	Hos_state
1	মুর্শিদাবাদ জেলা হাসপাতাল (Murshidabad District Hospital)	লালগোলা (Lalgola)	মুর্শিদাবাদ (Murshidabad)	পশ্চিমবঙ্গ (West Bengal)
2	নদিয়া জেলা হাসপাতাল (Nadia District Hospital)	চাপরা (Chapra)	নদিয়া (Nadia)	পশ্চিমবঙ্গ (West Bengal)

Table 8
Doctor table of default database

Doc_id	Doc_name	Doc_qualification	Doc_specialist	Hos_id	Dept_id
1000	কেয়া গরাই (Keya Gorai)	এম.বি.বি.এস. (M.B.B.S.)	অপথ্যালমোলজি বিশেষজ্ঞ (Ophthalmologist)	1	10
1010	শমীতা দাশগুপ্ত (Shamita Dasgupta)	এম.ডি. (M.D.)	অস্থি বিশেষজ্ঞ (Orthopaedist)	2	20

Table 9
Department table of default database

Dept_id	Dept_name	Hos_id
10	অপথ্যালমোলজি বিভাগ (Ophthalmology Department)	1
20	অস্থি চিকি□সা বিভাগ (Orthopedic Department)	2

(Table 7), Doctor (Table 8) and Department (Table 9) tables. The pattern (নদিয়া, পুরুলিয়ায়, হাসপাতাল, আছে, হাসপাতাল আছে) will be generated by the BLQPS in step iv. Each pattern may be an entity name, an attribute name or a value. Each pattern will be checked with entities table. If the corresponding pattern matches with any value of synonyms field in entities table, then corresponding row values will be fetched like entity name, primary key, foreign key, candidate key except synonyms value and the corresponding row value will be inserted into the semantic table; else the pattern will go for matching in attributes table. If the corresponding pattern matches with any value of synonyms field in attributes table then corresponding row values will be fetched like entity name, attribute name, primary key, foreign key and candidate key except synonyms value and the system will insert these into the semantic table else the pattern will go to the default data base for matching.

If the pattern matches with any value in any table in the default database then the column name of the corresponding table will be selected and the attributes table of the synonyms database will be searched again corresponding to the column name wherein the entire row values with corresponding columns name are selected.

i. নদিয়া (nothyea) – This will be selected as default database value and corresponding row value from attributes table will be fetched.
ii. পুরুলিয়ায় (puruliẹae) – This will be selected as default database value and corresponding row value from attributes table will be fetched.
iii. হাসপাতাল (haspathal) (Hospital) – This will be selected as entity from enttities table and corresponding row value will be fetched from entity table.
iv. আছে (ache) (Available) – This will not be selected.
v. হাসপাতাল আছে (haspathal) (ache) (Hospital) (Available) – This will not be selected.

After completion of synonyms database matching, corresponding row values of নদিয়া (nothyea), পুরুলিয়ায় (puruliẹae), হাসপাতাল (haspathal) from attributes and entities tables will be fetched and inserted into the semantic table except value of synonyms attribues from both table.

The representation of above mention query after semantic analysis has given in Table 3.

3.1.6. SQL generation

In this phase, the BLQPS generates SQL from the semantic table. The general format for retrieving data from table(s) is SELECT attribute 1, attribute 2, attribute 3… attribute n FROM entity 1 (table 1), entity 2 (table 2), entity 3 (table 3)… entity n (table n) WHERE condition 1 and condition 2 and condition 3… and condition n. SELECT, FROM and WHERE clauses are fixed in the structure of a SQL query for retrieving data, that why the proposed system has to find attribute(s), entities and condition(s). There are several cases for finding attribute(s), entities and condition(s).

Case i: The system identifies attributes if the attribute_name field is NOT NULL, value field is NULL and entity_name field is also NOT NULL; then values in the attribute_name field of Semantic Table shown in Table 3 is treated as attribute(s). The system concatenates those attribute(s) with their corresponding entity name by "." operator.

Case ii: The BLQPS considers all attributes if the attribute_name field is NULL, value field is NULL and entity_name field is NOT NULL in the semantic table. The system concatenates the "*" to their corresponding entity name by "." operator.

Case iii: If the attribute_name field is NOT NULL and value field is also NOT NULL then whatever attributes are contained in the attribute_name field is treated as condition. The system concatenates those attribute(s) followed by "=" and value with their corresponding entity name by "." operator.

Case iv: The BLQPS finds entity name from the value of entity_name field in semantic table. In case the entity_name field contains duplicate value, the proposed system selects distinct entity name.

Case v: If the value of primary_key field of one entity matches with the value of foreign_key or candidate_key of other entity in the semantic table then the system will perform joining operation between these two entities.

Case vi: If two or more entries in value field is NOT NULL and their corresponding attribute_name, entity_name field contains same value, it means the particular attribute of an entity has a list of values. In this case system will use IN clause.

Case vii: Single entry in value field is NOT NULL and their corresponding attribute_nameentity_ namefield is also NOT NULL. In this case system will use "=" operators.

Case viii: If two or more condition exists, the system concatenates all condition(s) using AND.

After SQL generation the above mentioned query i.e. নদিয়া ও পুরুলিয়ায় কি কি হাসপাতাল আছে? (nothyẹa o purulyẹaẹ ki ki haspathal ache?) will be converted to following SQL SELECT hospital.* FROM hospital WHERE hospital.hos_district IN ('নদিয়া' (nothyẹa), 'পুরুলিয়ায়' (purulyẹaẹ)).

3.1.7. SQL processed by system

Finally the SQL is executed and the desired result is fetched from the default database by the BLQPS. The result has been given in tabular format in Table 4.

3.2. Knowledge representation of the BLQPS

The knowledge data has been stored in default database. The default database has been used for knowledge extraction using natural language query. The Database Administrator, Knowledge Administrator, System Administrator or any other resource person may update the proposed knowledge database. The synonyms database has been used to generate the semantic table. The synonym database contains entities table and attributes table. The default database contains of hospital table, doctor table and department table.

3.2.1. Synonyms database

3.2.1.1. Structure of entities table

The entities table consists of six fields. These fields are entity_id, entity_name, synonyms, primary_key, foreign_key and candidate_key. In this table, entity_id field is the primary key. Entity_name field contains participating entity name corresponding to table names of default database like hospital, doctor and department. The synonyms field contains all possible synonyms of the entities in Bengali language. The primary_key field contains the name of primary key field of their respective entity. Similarly the foreign_key and candidate_key fields contain the name of the foreign key and candidate key field of their corresponding entity if value corresponding to foreign_key and candidate_key exists, otherwise they shall be NULL respectively. The structure of entities table has been given in Table 5.

3.2.1.2. Structure of attributes table

The attributes table consists of seven fields. These fields are attribute_id, entity_name, attribute_name, synonyms, primary_key, foreign_key and candidate_

Table 10
Query String array for NL query after tokenization

Tokens	বাকুড়া (bãnkoora)	পুরুলিয়া (puruliẹa)	ও (o) (And)	নদিয়ার (nothyar)	হাসপাতালের (haspathaler)	মধ্যে (mothe)	অস্থি (osthi) (Orthopedic)	চিকিৎসা (chykitsa) (Treatment)	বিভাগ (bybhag) (Department)	কোথায় (kothaẹ) (Where)	কোথায় (kothaẹ) (Where)	আছে (ache) (Available)
Array index	0	1	2	3	4	5	6	7	8	9	10	11

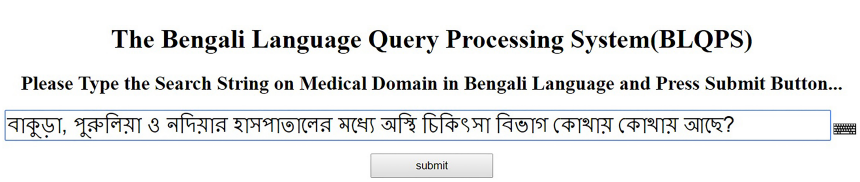

Fig. 3. Natural language query in BLQPS.

key. In this table, attrubute_id field is the primary key. The entity_name contains the entity name (Table name) of default database. The attribute_name field contains all attributes of entities in default database. The synonyms field contains all possible synonyms of attributes of entities. Synonym words have been stored in Bengali language. The primary_key field contains the name of primary key field of corresponding entity. Similarly the foreign_key and candidate_key fields contain the name of foreign key and candidate key fields of corresponding entity if the value of them exists; otherwise they shall be NULL respectively. The structure of attributes table has given in Table 6.

3.2.2. Default database

3.2.2.1. Structure of hospital table

The hospital table consists of five field. These fields are hos_id, hos_name, hos_add, hos_district, hos_state. The hos_id field is the primary key of this table. Using this hos_id field the BLQPS uniquely identify each entity instance of the table. The hos_name field contains hospital name, hos_add field contains the hospital address, hos_distrct field contains district name where hospital is situated. Similarly hos_state field contains state name where the hospital is located. The structure of hospital table has given below in Table 7.

3.2.2.2. Structure of doctor table

The doctor table consists of six fields. These fields are doc_id, doc_name, doc_qualification, doc_ specialist, hos_id and dept_id. The structure of doctor table has given in Table 8.

3.2.2.3. Structure of department table

The department table consists of three fields. These fields are dept_id, dept_name, hos_id. The structure of department table has given in Table 9.

4. Methodology and tools used

HTML, PHP, MySQL and Avro Bengali software have been used to develop the proposed system. HTML has been used as front end to design the web pages structure. PHP is a server side scripting language that has been used in back end. The knowledge database (default database) and synonyms database has been implemented in MySQL. All Bengali queries in Bengali transcript has followed IPA notation as per Help: IPA/Bengali given in website https://en.wikipedia.org/wiki/Help:IPA/Bengali

4.1. Step i

The Bengali Language Query Processing System (BLQPS) is a domain specific natural language query processing system. The system has been designed to handle medical related queries in Bengali. The user will log into the system and will post a query in Bengali. The query window of the proposed system has given in Fig. 3.

For example the user posts a query in Bengali. The query has been given.

বাকুড়া, পুরুলিয়া ও নদিয়ার হাসপাতালের মধ্যে অস্থি চিকিৎসা বিভাগ কোথায় কোথায় আছে? (bãnkoora, puruliẹa o nothyar haspathaler mothe osthi chykitsa bybhag kothaẹ kothaẹ ache?) (Where is the orthopedic department available among Bankura, Purulia and Nadia's Hospital?).

The BLQPS tokenizes the query into twelve tokens and stores them into a query string array after removing punctuation marks.

The array of tokens and array index of given query string after tokenization has been given in Table 10.

Table 11
Score array with score value

Array index	0	1	2	3	4	5	6	7	8	9	10	11
Score array	1	1	5	1	1	4	1	1	1	2	2	1

4.2. Step ii

After tokenization, The BLQPS selects every token from the tokenized query and compares with each word of interrogative's word list. If token matches with any word of interrogative's word list then the score of the selected token will be same as score of the interrogative's word list i.e. 2. Otherwise token will be compared with each word of pronoun's list. If token matches with any word of pronoun's list then the score of the selected token will be same as score of the pronoun's list i.e. 3. Otherwise in the similar way token will be compared with preposition's list and conjunction's list respectively. If token matches then corresponding array list score will be assigned. The selected token is not matched with any of the above mention known word list then the proposed system will determine the selected token is unknown and score will be 1. Using above mentioned procedure the proposed system will assign a specific score in the score array after POS tagging. The score value of first token from token array will be assigned at 0^{th} index in score array. The next score value of second token from token array will assigned at 1^{st} index in score array and other score value of token(s) or word(s) will be assigned at the position of score array as per their array indexing number in token array. The proposed system simply ignores known token(s) or word(s) by identifying score value other than 1. The score will be calculated by the BLQPS after POS tagging.

The score of all tokens of user given has been given in Table 11.

4.3. Step iii

বাকুড়া, পুরুলিয়া ও নদিয়ার হাসপাতালের মধ্যে অস্থি চিকিৎসা বিভাগ কোথায় কোথায় আছে?

(bãnkoora, puruliẹa o nothyar haspathaler mothe osthi chykitsa bybhag kothaẹ kothaẹ ache?) (Where is the orthopedic department available among Bankura, Purulia and Nadia's Hospital?), has been tokenized and shown in Table 10. After POS tagging the BLQPS maps known and unknown token or word by considering index of query string array(token array) and score array as well as score value of the score array. The nonconsecutive unknown word generates only one pattern. Here আছে (ache) is one nonconsecutive unknown token or word. So the token আছে (ache) will be the single pattern. But two consecutive unknown tokens or words are বাকুড়া (bãnkoora) and পুরুলিয়া (puruliẹa). So these two consecutive unknown tokens shall generate more than one pattern. The token order occurrence is maintained in the generated pattern which is made up of two or more than two tokens. The generated pattern বাকুড়া পুরুলিয়া (Bakuṛa puruliẹa) is made up of two tokens. The token বাকুড়া (bãnkoora) occurs before the token পুরুলিয়া (puruliẹa). That why the pattern বাকুড়া পুরুলিয়া (Bakuṛa puruliẹa) will be generated and the pattern পুরুলিয়া বাকুড়া (puruliẹa Bakuṛa) will not be generated by the BLQPS. Similar way all consecutive tokens will generate patterns. All generated pattern has been given below.

i. বাকুড়া (bãnkoora)
ii. পুরুলিয়া (puruliẹa)
iii. বাকুড়া পুরুলিয়া (Bakuṛa puruliẹa)
iv. নদিয়ার (nothyar)
v. হাসপাতালের (haspathaler)
vi. নদিয়ার হাসপাতালের (nothyar haspathaler)
vii. অস্থি (osthi) (Orthopedic)
viii. চিকিৎসা (chykitsa) (Treatment)
ix. বিভাগ (bybhag) (Department)
x. অস্থি চিকিৎসা (osthi chykitsa) (Orthopedic Treatment)
xi. চিকিৎসা বিভাগ (chykitsa bybhag) (Treatment Department)
xii. অস্থি চিকিৎসা বিভাগ (osthi chykitsa bybhag) (Orthopedic Department)
xiii. আছে (ache) (available)

4.4. Step iv

The BLQPS will compare each pattern (বাকুড়া (bãnkoora), পুরুলিয়া (puruliẹa), বাকুড়া পুরুলিয়া (Bakuṛa puruliẹa), নদিয়ার (nothyar), হাসপাতালের (haspathaler), নদিয়ার হাসপাতালের (nothyar haspathaler), অস্থি (osthi) (Orthopedic), চিকিৎসা (chykitsa) (Treatment), বিভাগ (bybhag) (Department), অস্থি চিকিৎসা (osthi chykitsa) (Orthopedic Treatment), চিকিৎসা বিভাগ (chykitsa bybhag) (Treatment Department), অস্থি চিকিৎসা বিভাগ (osthi chykitsa bybhag)(Orthopedic Department), আছে (ache) (available) with synonym database as well as default database and insert into semantic table the matched value when a match occurs.

i. বাকুড়া (bãnkoora) – This will be selected as default database value and corresponding row value from attributes table will be fetched.

Table 12
Instances of semantic table

Id	Entity_name	Attribute_name	Primary_key	Foreign_key	Candidate_key	Value
1	hospital	hos_district	hos_id			বাকুড়া (bãnkoora)
3	hospital	hos_district	hos_id			পুরুলিয়া (puruliẹa)
7	hospital	hos_district	hos_id			নদিয়া (nothyar)
8	hospital		hos_id			
11	department		dept_id	hos_id		
14	department	dept_name	dept_id	hos_id		অস্থি চিকি□সা বিভাগ (osthi chykitsa bybhag) (Orthopedic department)

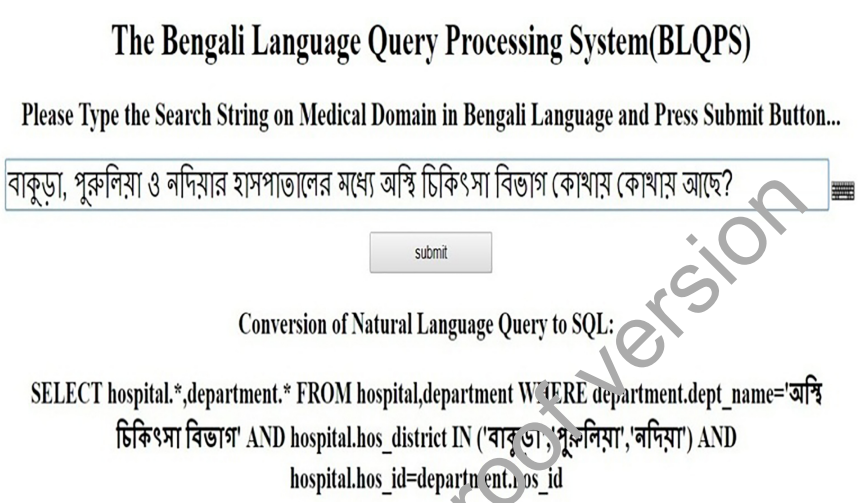

Fig. 4. Conversion of NL query to SQL.

ii. পুরুলিয়া (puruliẹa) – This will be selected as default database value and corresponding row value from attributes table will be fetched.
iii. বাকুড়া পুরুলিয়া (Bakuṛa puruliẹa) – This will not be selected.
iv. নদিয়ার (nothyar) – This will be selected as default database value and corresponding row value from attributes table will be fetched.
v. হাসপাতালের (haspathaler) – This will be selected as entity from enttities table and corresponding row value will be fetched from entity table.
vi. নদিয়ার হাসপাতালের (nothyar haspathaler) – This will not be selected.
vii. অস্থি (osthi) (Orthopedic) – This will not be selected.
viii. চিকি□সা (chykitsa) (Treatment) – This will not be selected.
ix. বিভাগ (bybhag) (Department) – This will be selected as entity from enttities table and corresponding row value will be fetched from entity table.
x. অস্থি চিকি□সা (osthi chykitsa) (Orthopedic Treatment) – This will not be selected.
xi. চিকি□সা বিভাগ (chykitsa bybhag) (Treatment Department) – This will not be selected.
xii. অস্থি চিকি□সা বিভাগ (osthi chykitsa bybhag) (Orthopedic Department) – This will be selected as default database value and corresponding row value from attributes table will be fetched.
xiii. আছে (ache) (available) – This will not be selected.

After completion of synonyms database matching, corresponding row values of বাকুড়া (bãnkoora), পুরুলিয়া (puruliẹa), নদিয়ার (nothyar), অস্থি চিকি□সা বিভাগ (osthi chykitsa bybhag) (Orthopedic department), হাসপাতালের (haspathaler), বিভাগ (bybhag) (Department) from attributes and entities tables will be fetched and inserted into the semantic table except value of synonyms attribues from both table. The representation of above mention query after semantic analysis has given in Table 12.

4.5. Step v

After SQL generation the above mentioned query i.e. বাকুড়া, পুরুলিয়া ও নদিয়ার হাসপাতালের মধ্যে অস্থি চিকি□সা বিভাগ কোথায় কোথায় আছে?

(bãnkoora, puruliẹa o nothyar haspathaler mothe osthi chykitsa bybhag kothaẹ kothaẹ ache?) (Where is the

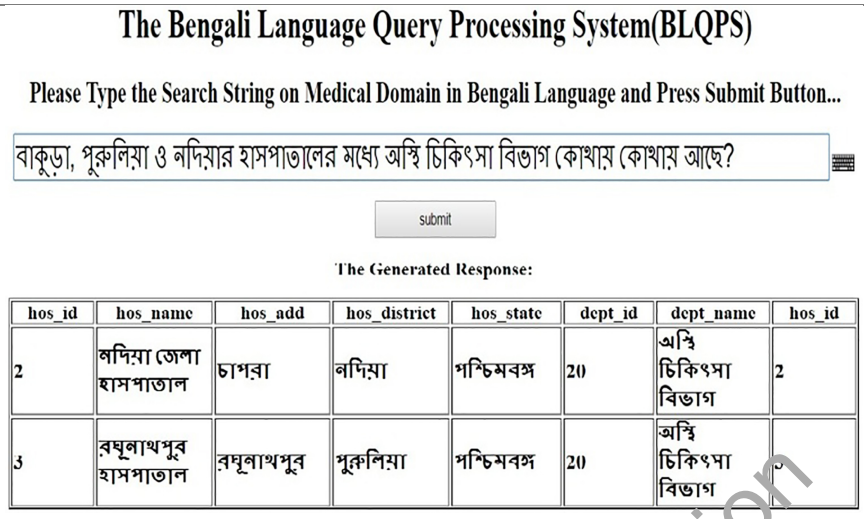

Fig. 5. User request with generated response.

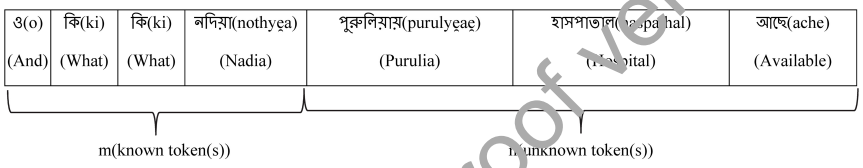

Fig. 6. Numbers of known and unknown tokens.

orthopedic department available among Bankura, Purulia and Nadia's Hospital?) will be converted to following SQL SELECT hospital.*,department.* FROM hospital, department WHERE department.dept_name = 'অস্থি চিকি□সা বিভাগ' (osthi chykitsa bybhag) (Orthopedic department) AND hospital.hos_district IN ('বাকুড়া' (Bakuṛa), 'পুরুলিয়া' (puruliẹa), 'নদিয়া' (nothyar)) AND hospital.hos_id = department.hos_id. Conversion of natural language query to SQL has been in Fig. 4.

4.6. Step vi

Finally the SQL will be executed by the BLQPS and the desired result will be fetched from default database. The user request with generated response has been given in Fig. 5.

4.7. Time complexity of the BLQPS

i. After tokenization, let p tokens be present in the user given string. Let $p = m + n$. After POS tagging, the proposed system identifies m numbers of known tokens known and n numbers of unknown tokens.

ii. Let there be q numbers of pre-defined word list present. Each list contains a numbers of words.
Time taken to search 1st token in the 1st list of words $= a$ unit.
Time taken to search 1st token in the 2nd list of words $= a$ unit.
Time taken to search 1st token in the 3rd list of words $= a$ unit.
…
…
…
Time taken to search 1st token in the q^{th} list of words $= a$ unit.
Therefore, total time taken by the 1st token $= a + a + a + \ldots (q \text{ times}) = qa$unit.
$\therefore$ Total time taken by p numbers of tokens to search in the q numbers of list of words $= pqa$ unit in the POS tagging phase.
After POS tagging known and unknown tokens have been given in Fig. 6.
Above example discussed in Section 3(iii) has been taken in the Fig. 6.

iii. If token taken one at a time $= n$ number of pattern will be generated.

Table 13
General features based comparative study of the proposed system with similar type system

Sl. No.	Author(s) & name of the system	General features of other systems	General features of BLQPS (proposed system)
1	P. Kaur et al., Conversion Of Natural Language Query To SQL [17]	i) In this system, uttered speech is identified by Hidden Markov Model (HMM). ii) This system uses WordNet. iii) Pre-defined grammar rules have been used in this system. iv) The time complexity has not been mentioned here.	i) The BLQPS uses pattern generation to identify the word. ii) The BLQPS does not use WordNet. It uses synonym database and default database. iii) Pre-defined grammar rules are not used in this system. iv) The time complexity has been computed here.
2	K.M.A. Hasan et al., Recognizing Bangla Grammar Using Predictive Parser [12]	i) This system uses predictive parser to identify Bangla grammar. ii) The proposed system uses the pre-defined XML dictionary for parts of speech tagging. iii) The time complexity has not been mentioned here.	i) The BLQPS uses scoring and pattern generation technique to identify words. ii) This system uses pre-defined string arrays for parts of speech tagging. iii) The time complexity has been computed here.
3	K.N. ElSayed, An Arabic Natural Language Interface System for a Database of the Holy Quran [18]	i) An Arabic Natural Language Interface System for a Database of the Holy Quran parses the Arabic sentence using context free grammar rules. ii) This system contains Arabic word and their corresponding SQL command. iii) The time complexity has not been mentioned here.	i) The BLQPS uses scoring and pattern generation. ii) SQL query is dynamically generated. There is no need to give every word to its equivalent SQL command. iii) The time complexity has been computed here.
4	A. Sawant et al., Natural Language to Database Interface [20]	i) This system uses SQL template to identify attribute(s) as well as table(s). ii) The time complexity has not been mentioned here.	i) The BLQPS uses some pre-defined condition to identify attribute(s) and table(s) dynamically. ii) The time complexity has been computed here.
5	R. Alexander, et al., Natural Language Web Interface for Database (NLWIDB) [19]	i) Natural Language Web Interface for Database (NLWIDB) performs checking operation to see if the question string is present in Data Dictionary. ii) The NLWDB uses SQL template string to convert the NL to SQL element. iii) The time complexity has not been mentioned here.	i) This system does not contain Data Dictionary. ii) The BLQPS uses some pre-defined condition to construct SQL dynamically. iii) The time complexity has been computed here.
6	J. Kaur et al., Implementation of query processor using Automata and natural language processing [15]	i) This automata based query processing system processes interrogative statements. ii) This system contains a Data Dictionary which stores all possible pre-defined words of a particular system. iii) The time complexity has not mentioned here.	i) The proposed system can process interrogative as well as assertive statements. ii) The BLQPS does not contain such type of Data Dictionary. iii) The time complexity has been computed here.
7	M.M. Anwar et al., Syntax Analysis and Machine Translation of Bangla Sentences [13]	i) Syntax Analysis and Machine Translation based system works on pre-defined grammar rules. ii) This system uses trained corpus to identify a word. iii) The time complexity has not been mentioned here.	i) The BLQPS works on scoring and pattern generation technique. ii) The proposed system uses synonym database, default to identity word. iii) The time complexity has been computed here.
8	K. Muntarina et al., Tense Based English to Bangla Translation Using MT System [16]	i) The Tense Based English to Bangla Translation Using MT System uses pre-defined lexicon to identify words. ii) It uses a set of production rules to convert English sentence to its corresponding Bengali sentence. iii) The time complexity has not been mentioned here.	i) The BLQPS uses synonym database, default database to identify word. ii) This system uses scoring and pattern generation to convert Bengali sentence to its corresponding SQL. iii) The time complexity has been computed here.
9	A. Kataria et al., Natural Language Interface for Databases in Hindi Based on Karaka Theory [9]	i) The NLIDB on Hindi language has been developed using Paninian Framework and Karaka theory. ii) This system uses shallow parser. iii) This Hindi language based NLIDB finds root word from given word. iv) The Graph Generator determines the relationship among the command, table name, attribute name and conditional part. v) The time complexity has not been mentioned here.	i) This has been developed on scoring and pattern generation. ii) The BLQPS does not use shallow parser. iii) This system does not find root word. It finds patterns of words. iv) The proposed system determines the relationship among the command, table(s), attribute(s) and conditional part using few pre-defined rules. v) The time complexity has been computed here.

Token taken two at a time $= n-1$ number of pattern will be generated.
Token taken three at a time $= n-2$ number of pattern will be generated.
…
Token taken n at a time $= 1$ number of pattern will be generated.
Total number of patterns generation without position replacement

$$=n+(n-1)+(n-2)+\ldots+1=\frac{n(n+1)}{2}$$

iv. The synonym database search – There are entity table, attribute table, default database tables.
Let there are x numbers of rows and y numbers of columns in entity table. There are z numbers of rows and w numbers of columns in attribute table.
There are s numbers of rows and t numbers of columns in default database table.
Time taken by 1st pattern to search in entity table $= xy$ unit time.
$\therefore$ Time taken by $\frac{n(n+1)}{2}$ pattern to search in entity table $= xy\frac{n(n+1)}{2}$ unit time.
Similarly, time taken by $\frac{n(n+1)}{2}$ pattern to search in attribute table $= zw\frac{n(n+1)}{2}$ unit time.
Similarly, time taken by $\frac{n(n+1)}{2}$ pattern to search in default database tables $= st\frac{n(n+1)}{2}$ unit time.
So the total time taken $= \{xy+zw+st\}\frac{n(n+1)}{2}$ unit.
v. Time taken to create SQL $= u1$ unit time.
vi. Time taken to generate response $= u1$ unit time.

$\therefore$ Total time complexity

$$\begin{aligned}
&= f(p,m,n,a,c,x,y,z,w,s,t,u1,u2)\\
&= p+pqa+\frac{n(n+1)}{2}+\\
&\qquad \frac{n(n+1)}{2}\{xy+zw+st\}\\
\therefore f(n) &\cong n+n\times n\times n+\frac{n(n+1)}{2}+\\
&\qquad \frac{n(n+1)}{2}\{n\times n+n\times n+n\times n\}\\
&= n+n^3+\frac{n(n+1)}{2}\{1+n^2+n^2+n^2\}\\
&= n+n^3+\frac{n(n+1)}{2}\{3n^2+1\}\\
&= n+n^3+\frac{(n^2+n)}{2}\{3n^2+1\}\\
&= n+n^3+\frac{3n^4}{2}+\frac{n^2}{2}+\frac{3n^3}{2}+\frac{n}{2}\\
&= \frac{3n^4}{2}+\frac{5n^3}{2}+\frac{n^2}{2}+\frac{3n}{2}\\
&= \frac{1}{2}(3n^4+5n^3+n^2+3n)=O(n^4)
\end{aligned}$$

5. General features based comparative study of the proposed system with similar type system

The comparative study of proposed system with other similar type system is features based. Few similar type systems have been considered for comparative study. Prabhudeep Kaur et al. have developed Conversion of Natural Language Query to SQL. This system is based on pre-defined grammar rules and WordNet that can able to convert speech to SQL using Hidden Markov Model (HMM) has been implemented in [17]. Another Bengali parser has been developed by Hasan et al. The Bengali parser works on creation of Bengali grammar from Bengali sentences. Authors have considered top down parsing method and avoided left recursion in context free grammar (CFG) as in [12]. Access of the Holy Quran has grown rapidly with the grown of huge numbers of smart mobiles, tablets and laptops. The system has developed by Khaled Nasser ElSayed to access the database of the Holy Quran. The primary features of this system are translation of natural Arabic question or imperative sentences to SQL command and answer extraction from the Holy Quran database. Parsing technique and little morphological process have been used to make the interface of this system that are based on Arabic context free grammar rules has been described in [18]. Aarti Sawant et al. have described natural language to database that can manage natural language question as an input. Authors have stated that the proposed system is able to generate textual response from relational database using natural language query. The natural language interface simplifies the textual data extraction from relational database without having essential knowledge of SQL has been developed in [20]. Other similar type systems like NLWIDB [19] of Alexander et al., natural language interpretation using automata [15] of Kaur et al., Bangla parser [13] of Anwar et al., English to Bangla Translation Using MT System [16] of Muntarina et al. and Natural Language Interface for Databases in Hindi Based on Karaka Theory [9] of Kataria et al. have been discussed details in Table 13.

6. Conclusion and future work

The Bengali Language Query Processing System (BLQPS) is an automated system which shall be able to handle Bengali language user queries. The user shall submit the query in Bengali Language. Then the BLQPS processes the query and generates response in Bengali language. The BLQPS is designed in such a way that naive Bengali users can interact with computerized system with their own language (i.e. Bengali). Queries containing adjectives cannot be processed by the proposed system, like "*What are the best hospitals in Bankura"* cannot be processed as best is an adjective. Hence, queries with qualitative terms defined by adjectives cannot be processed which is a limitation of the BLQPS.

From the time complexity analysis it is found that time complexity is in $O(n^4)$ which works well when the number of unknown words/tokens are few. However, as the number of unknown tokens increase, the time complexity increases greatly which is another drawback of this proposed system. Further comparative analysis needs to be done with other similar type systems using time complexity, amortized analysis (process based) so as to improve upon the time complexity of the proposed system. The searching technique needs to be improved in future work so that the searching time is reduced. Alternative techniques of searching and querying the database need to be developed which is the scope of future work.

Acknowledgments

This research work has been done at Research Project Lab under Dept. of Computer Science and Engineering of National Institute of Technology (NIT), Durgapur, The Authors would like to thank Dept. of Computer Science and Engineering, NIT, Durgapur, India for academically support to this research work.

References

[1] Reshamwala A, Mishra D, Pawar P. Review on natural language processing. Engineering Science and Technology: An International Journal 2013; 3(1): 113-116.

[2] Nihalani N, Motwani M, Silakari S. Natural language interface to database using semantic matching. International Journal of Computer Applications 2011; 31(11): 29-34.

[3] Kaur S, Bali RS. SQL generation and execution from natural language processing. International Journal of Computing and Business Research 2012. Available from: http://www.researchmanuscripts.com/isociety2012/54.pdf.

[4] Warren DHD, Pereira FCN. An efficient easily adaptable system for interpreting natural language queries. American Journal of Computational Linguistics 1982; 8(3-4): 110-122.

[5] Sujatha B, VishwanathaRaju S, Nagaprasad S. Efficient natural language query interface to databases. International Journal of Advanced Research in Computer Engineering and Technology 2014; 3(9): 3300-3308.

[6] Mukherjee P, Chakraborty B. A comparative analysis of permutation combination based and grammatical rule based knowledge provider system. Intelligent Decision Technologies 2017; 11(1): 39-60. doi: 10.3233/IDT-160276.

[7] Soumya MD, Patil BA. An interactive interface for natural language query processing to database using semantic grammar. International Journal of Advanced Research 2017; 3(4): 193-198.

[8] Borkar PS, Gahane L, Raut A, et al. Hindi language gui for transport system using natural language processing. International Research Journal of Engineering and Technology 2017; 4(3): 1293-1298.

[9] Kataria A, Nath R. Natural language interface for databases in Hindi based on karaka theory. International Journal of Computer Applications 2015; 122(7): 39-43.

[10] González JJ, Juárez JR, Fraire HJ, et al. Semantic representations for knowledge modeling of a natural language interface to databases using ontologies. International Journal of Combinatorial Optimization Problems and Informatics 2015; 6(2): 28-42.

[11] Mridha MF, Saha AK, Das JK. Solving semantic problem of phrases in NLP using universal networking language (UNL). International Journal of Advanced Computer Science and Applications 2014. Available from: https://thesai.org/Downloads/SpecialIssueNo9/Paper_3Solving_Semantic_Problem_of_Phrases_in_NLP_Using_Universal_Networking_Language.pdf.

[12] Hasan KMA, Mahmud A, Mondal A, et al. Recognizing Bangla grammar using predictive parser. International Journal of Computer Science and Information Technology 2011; 3(6): 61-73.

[13] Anwar MM, Anwar MZ, Bhuiyan MAA. Syntax analysis and machine translation of Bangla sentences. International Journal of Computer Science and Network Security 2009; 9(8): 317-326.

[14] Mahmud MR, Afrin M, Razzaque MA, et al. A rule based Bengali stemmer. International Conference on Advances in Computing, Communications and Informatics 2014. p. 2750-2756. doi: 10.1109/ICACCI.2014.6968484.

[15] 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): 1-5.

[16] Muntarina K, Moazzam MG, Bhuiyan MAA. Tense based English to Bangla translation using MT system. International Journal of Engineering Science Invention 2013; 2(10): 30-38.

[17] Kaur P, Shruthi J. Conversion of natural language query to SQL. International Journal of Engineering Sciences and Emerging Technologies 2016; 8(4): 208-212.

[18] ElSayed KN. An Arabic natural language interface system for a database of the Holy Quran. International Journal of Advanced Research in Artificial Intelligence 2015; 4(7): 9-14.

[19] Alexander R, Rukshan P, Mahesan S. Natural language web interface for database (NLWIDB). Proceedings of the Third International Symposium 2013. Available from: https://arxiv.org/ftp/arxiv/papers/1308/1308.3830.pdf.

[20] Sawant A, Lambateand P, Zore AS. Natural language to database interface. International Journal of Engineering Research and Technology 2014; 3(2): 1365-1368.

[21] Hamon T, Grabar N, Mougin F. Querying biomedical linked data with natural language questions. Intelligent Decision Technologies 2017; 8(4): 581-599. doi: 10.3233/SW-160244.

[22] Quarteroni S. Lightweight integration and natural language querying of heterogeneous data services. Intelligent Decision Technologies 2012; 6(2): 149-162. doi: 10.3233/IA-120037.

[23] Kaljurand K, Kuhn T, Canedo L. Collaborative multilingual knowledge management based on controlled natural language. Intelligent Decision Technologies 2015; 6(3): 241-258. doi: 10.3233/SW-140152.