



## Comparative Analysis of Question Answering System in Indian Languages

Jaspreet Kaur<sup>#1</sup>,  
<sup>#ME Student,</sup>

Dept. of Computer Science and Engineering,  
Panjab University, Chandigarh, India

Vishal Gupta<sup>\*2</sup>

<sup>\*Assistant Professor,</sup>

Dept. of Computer Science and Engineering,  
Panjab University, Chandigarh, India

**Abstract**— As there is increasingly high advancement in technology, so Question Answering is becoming major area of research for the researchers. Different queries are provided by the user in aim of getting accurate answers in Question Answering Systems. Question Answering provides perfect solution to retrieve valid and accurate answers to user question asked in natural language instead of query. Hindi, Telugu, Bengali etc are popular languages that are spoken in India. Currently these languages are taken into consideration by the researchers and a lot of work is being done in these and other Indian languages. In this paper we compare Question Answering Systems performance for different Indian languages. We discuss the best features of Question Answering systems built in different Indian languages and compare their performances.

**Keywords** — Indian, Hindi Question Answering System, Algorithms

### I. INTRODUCTION

In this paper, we analyze the previous work for Question Answering systems in different Indian languages and evaluate their performances and provide suggestions for further improvement. Comparing Indian languages with other languages, word segmentation is a key problem in Indian question answering. We review studies on different techniques used for Indian languages and discuss important issues which are helpful for building QA systems. Machine learning approaches currently represent the main stream on many QA research issues, we believe, by efficiently utilizing the above resources, the performance of machine learning approaches can be improved further in Indian question answering. Different Question Answering Systems being built in Indian languages are discussed in this paper and their performance is evaluated.

### II. PRASHNOTTAR: A Hindi Question Answering System

Now a day, internet includes websites in different languages and no. of natural languages with their dialects is approximate 4000. Hindi holds 5<sup>th</sup> position among top 100 spoken languages in the world, with no. of speakers being close to 200 million. The developed QA system uses shallow parser. The parser gives the analysis of a sentence in terms of morphological analysis, POS tagging and chunking. Immediate output of individual modules along with the final output are generated, all being in Shakti Standard Format (SSF). PRASHNOTTAR: Hindi QA system' is considerable if used for four mentioned class of questions: 'when', 'where', 'what time' and 'how many'. Semantic approach with probability distribution scenario, if considered, for the algorithm, can result in a more scalable system in terms of variety of questions asked. Furthermore, the system can be extended to searching dynamic dataset (available on internet) instead of searching a static dataset.

#### A. Architecture

A question or query is input by the user in Hindi, which is used to extract all possible answers for the input question [1]. Architecture of Hindi Question Answering System is shown in Figure 1.

Following modules constitute the system:

1. *Query processing*: the module processes and analyzes the input question leading to its classification. The input question can belong to question classes like when, where, how many, what time, etc.

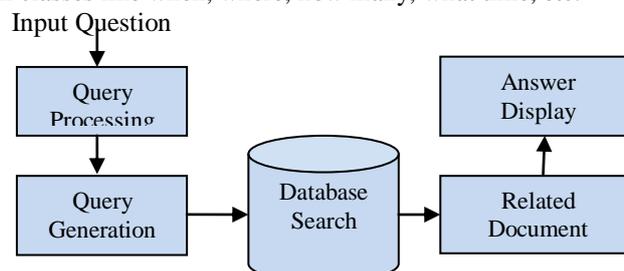


Fig. 1: Architecture of Hindi Question Answering System

2. *Query generation*: query logic language (QLL) is used to express the input question.
3. *Database search*: stored database is searched for possible results. The relevant results may be with keywords and rules are input to the next module.
4. *Related document*: results so generated, in the previous module, are stored as documents.
5. *Answer display*: result is converted into Hindi text from wx format and displayed to user in user-interface.

### B. Implementation

Different steps in implementation are explained below:

1. *Question preprocessing*: Query logic language (QLL), a subset of Prolog, was used to process input questions. Rules were developed to translate query in Hindi into its logical form. Different predicates were defined for each class of question (e.g. when, where, how many, etc).
2. *Question classification*: The step processed the question to identify its class [2]. Processing of question included parsing by "Hindi shallow parser"; which identified a list of parts of speech (POS) like noun, adjective, verb; and information regarding what is being asked.
3. *Answer extraction*: complexity of answer extraction depends on the following factors:
  - Complexity of the question asked
  - Repository searched for possible answers
  - Search technique
  - Focus of question and its context
  - Many a times, irrelevant answers may be extracted by a QA system. Traditional systems used to take each word as an independent unit during search process, thereby ignoring the relation between words in a phrase or neighborhood [3]. Thus, there may be some results which contain most of keywords but still are irrelevant as a possible answer to the question asked [4]. Also, term position, term sequence, synonym etc were altogether ignored which also affected the accuracy of possible answer returned [5].

### C. Analysis and Results

The system was designed majorly for 4 types of questions: when, where, what time and how many. 15 questions of each type were given as input to the system and no. of errors was manually counted. The system had an overall accuracy of about 68.00%. The accuracy of 'when', 'what time' and 'how many' were relatively higher than 'where' questions as identification of time and date is easier than identification of proper nouns [6] [7]. Also, the system did no further processing on the questions which failed to generate answers, also a reason for low efficiency of the system.

TABLE I  
ANSWER EXTRACTION

| Question Type | No. of Question | No. of Error | Accuracy |
|---------------|-----------------|--------------|----------|
| When          | 15              | 5            | 66.66%   |
| Where         | 15              | 7            | 53.00%   |
| How many      | 15              | 4            | 73.33%   |
| What time     | 15              | 3            | 80.00%   |
| Total         | 60              | 19           | 68.00%   |

### III. Hindi-English Cross-Lingual Question Answering System

A cross-lingual question-answering (CLQA) system for Hindi and English was developed in one month. Main sources of errors in the system were evaluated which suggested several ways for improving the accuracy of the CLQA system.

#### A. System Overview

There are two kinds of components: those for tagging, in advance, the names and numeric expressions in a Hindi newspaper (taggers) and those that comprise the main question answering system (system components) [8]. The taggers are designed to tag possible answers in the text; they are applied in advance to annotate the text [9]. When a question to the QA system is analyzed, the type of expected answer is determined. Three NE types are defined in the shared NE task of the SLE (person, organization, and location) and 15 numeric types are developed. Once the type of expected answer is determined, the QA system tries to find answer candidates in the text that match the type and relate to the question. All texts are automatically tagged by the NE and numeric taggers in advance, so that answers may easily be found at runtime [10].

1. *Taggers*: In basic question answering systems, there are number of word or phrase types but in this system only three NE types are defined in the SLE's shared NE task and 15 numeric types [11] [12] [13].
  - 1.1. *NE Tagger*: The common NE task for the SLE involved three types of names: names of people, names of organizations, and names of locations. Using these specifications, about 600,000 words of data were considered

cooperatively by BBN, the Linguistic Data Consortium, and our group at NYU [14]. When tested on a sample of 20 hand annotated Hindi BBC news reports, the tagger had a precision of 82% and recall of 74%.

- 1.2. *Numeric Expression Tagger*: The Numeric expression tagger was developed by using simple pattern matching; designed 15 types [15].
2. *System Components*: It consists of four components- QE, CLIR, AF and MT. Each component is developed using existing programs.
  - 2.1. *Question Examiner (QE)*: The program checks the question, and outputs the expected type of answer and keywords in English, which will be used by the next components. The input is examined by a part-of-speech tagger and chunker; prepared key question patterns are then applied to identify the expected answer type.
  - 2.2. *CLIR System*: Cross-lingual information retrieval (CLIR) is done in order to find the list of documents that might contain the answer candidates. The system looks up the keywords in an English- Hindi bilingual dictionary and a word list of all possible Hindi translations of the English keywords is created.
  - 2.3. *Answer Finder (AF)*: Top 20 newspaper articles retrieved by the CLIR component to find the answer candidates. Candidates were scored based on their distance from Hindi keywords and keyword weight. The score for a candidate expression is calculated by the formula shown in equation given below.

$$\text{score}(e) = \sum_{k \in \text{keywords}} \frac{c_1}{\text{dist}(k, e) + c_2} * \text{keyword\_weight}(k) * \text{article\_score}$$

Where  $c_1$  and  $c_2$  are constants,  $\text{dist}(k, e)$  is the distance between the candidate and keyword,  $\text{keyword\_weight}$  is the weight assigned by the question examiner and  $\text{article\_score}$  is the score assigned by the CLIR system.

- 2.4. *MT (Machine Translation) System*: Answer candidates are now in Hindi and need to be translated into English. Two options are provided: one is to use ISI's machine translation (MT) server, and the other is to use word-to-word translation. ISI's MT system is also a product of SLE and was available to us as an MT server. The word-to-word translation was just a look-up of the bilingual dictionary with all possible translations listed for each word of the context.
3. *User Interface*: The system is accessible via the Web. There are two pages; one to accept a question and the other to display the answer candidates and context sentences.

#### B. Evaluation

CLQA System was evaluated using questions created by a native speaker of Hindi.

1. *Creating Question and Answers*: Native Hindi speaker was asked to read the newspaper articles and create few questions for the system. The native speaker used the retrieval tool for newspaper articles and gave it some random keywords, and then when an interesting article was found, question was made about it. There were a total of 56 questions.
2. *Evaluation Results*: The system was run for the 56 questions and evaluated the result. For 25 questions, the system found the correct answer in English. The number of questions for which the correct answer was found at a given rank is shown in Table II. The MRR (mean reciprocal rank) for the top 5 answers is 0.25 for the 56 questions. Although the question's level of difficulty is a very important factor, and it is a bit dangerous to compare different evaluations, the MRR score is not much lower than that of some monolingual QA systems.

TABLE II  
NUMBER OF QUESTIONS AND CORRECT ANSWER RANKING

| Rank  | Hindi | English |
|-------|-------|---------|
| 1     | 10    | 9       |
| 2     | 2     | 2       |
| 3     | 8     | 6       |
| 4     | 1     | 1       |
| 5     | 1     | 1       |
| 6     | 2     | 2       |
| 7     | 1     | 1       |
| 8     | 4     | 3       |
| 9     | 0     | 0       |
| 10    | 0     | 0       |
| Total | 29    | 25      |

#### IV. Bengali Question Classification: Towards Developing Qa System

Bengali is an important eastern Indo-Aryan language. First step in developing a question answering system is to classify natural language question properly. Single-layer taxonomy is proposed which consists of only nine course-grained classes. The interrogatives in Bengali language are also studied and categorized. The proposed automated classification

work depends on various machine learning techniques. This system has achieved up to 87.63% accuracy using decision tree classifier [16].

#### A. Classification Module

Although there many supervised learning approaches for question classification [17] but all these approaches mainly differ in the classifier used by them [18] [19]. Assuming that Bengali is unambiguous, i.e. a question is having only one class. So, one label is assigned to the given question and thus described as follows:

$C = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8\}$ , where  $C$  is the set of possible classes

$Q = \{Q_1, Q_2, Q_3 \dots Q_{N-1}, Q_N\}$ ; where  $Q$  is the set of  $N$  given questions

The task of the designed question classifier is to assign the most likely classes  $C_k$  to a question  $Q_m$ . Latest studies also consider [20] one label for one question [19] [21]. In this different question classifiers used are: Naïve Bayes (NB), Kernel Naïve Bayes (KNB), Decision Tree (DT) and Rule Induction (RI). Among all of these, DT has best performed.

1. *Naïve Bayes (NB)*: It is a simple classifier based on applying Bayes theorem with strong independence assumptions that is it assumes that presence or absence of a class is unrelated to the presence or absence of any other feature, given the class variable. Suppose a class  $C$  with  $m$  classes,  $c_1, c_2 \dots c_m$ , and  $x$  be an attribute vector of all other attributes. The conditional probability of class label  $c_i$  can be expressed as follows:

$$P(C = c_i | x) = \frac{P(x | C = c_i)P(C = c_i)}{P(x)}$$

where  $P(C = c_i)$  is the probability of the class label  $c_i$  and can be approximated from the data directly.  $P(x)$ , probability of unknown sample does not have to be calculated as it does not depend on the class label and the class label with the best probability can be extracted without its knowledge.

2. *Kernel Naïve Bayes (KNB)*: It is the updated version of Naïve Bayes classifier that uses estimated kernel density. Conditional probability  $P(x | C = c_i)$  can be written as a kernel density estimate for class  $c_i$ .

$$P(x | C = c_i) = f_i(x) \text{ And } f_i(x) = \sum_{t=1}^n K_i(x, x_t) ;$$

Where,  $x_t$  are training points and  $K_i(x, x_t)$  is a kernel function.

3. *Rule Induction (RI)*: RI learns a trimmed set of rules with respect to the information gain. Initialising with the less important classes, the algorithm iteratively grows and trims rules until there are no positive examples left or the error rate is more than 50%. In the growing phase, for each rule greedily conditions are added to the rule till the rule is perfect. In the trim phase, for each rule any final sequences of the antecedents are trimmed with the trimming metric  $p / (p + n)$ .

4. *Decision Tree (DT)*: Decision trees are powerful classification methods that can also easily be understood. In order to classify an example, the tree is traversed bottom-down. Each node in a decision tree is marked with an attribute. The example's value for this attribute determines which of the out coming edges is taken. For nominal attributes, we have one outgoing edge per possible attribute value, and for numerical attributes the outgoing edges are labelled with disjoint ranges.

#### B. Experiments and Results

In this system 1100 questions have been selected and processed to extract the features. Two qualified annotators have been labelled the questions with an agreement score of 95.93%. 770 questions were used for training and remaining 330 questions were used to evaluate the classification model. Accuracy and Error are the two performance metrics used to asses results. Accuracy and Error can be calculated as follows:

$$\text{Accuracy} = \frac{TP + TN}{P + N}$$

$$\text{Error}(\%) = 100 - \text{Accuracy}$$

Where, TP = true positive samples; TN = true negative samples; P = positive samples; N = negative samples  
When lexical and semantic features together are considered and NB, KNB, RI and DT classifiers are applied respectively, then it has been noted that the performance of all the classifiers improve. Experiment results in which both lexical and syntactical features are considered produces results shown in Table III.

TABLE III  
CLASSIFIERS PERFORMANCE (LEXICAL AND SYNTACTICAL FEATURES)

| Features                    | Classifier | Accuracy (%) | Error (%) |
|-----------------------------|------------|--------------|-----------|
| Lexical<br>+<br>Syntactical | NB         | 81.34        | 18.66     |
|                             | KNB        | 82.37        | 17.63     |
|                             | RI         | 84.23        | 15.77     |
|                             | DT         | 85.69        | 14.31     |

Finally, when lexical, syntactical and semantic features together are used and applied for all the classifiers, improves the performance of NB and KNB. The results for all features used and the different classifiers applied on them are shown in Table IV. DT classifier has performed best of all classifiers on Bengali Questions.

TABLE IV  
CLASSIFIERS PERFORMANCE (LEXICAL, SYNTACTICAL AND SEMANTIC FEATURES)

| Features                                     | Classifier | Accuracy (%) | Error (%) |
|--|------------|--------------|-----------|
| Lexical<br>+<br>Syntactical<br>+<br>Semantic | NB         | 81.89        | 18.11     |
|  | KNB        | 83.21        | 16.79     |
|  | RI         | 85.57        | 14.43     |
|  | DT         | 87.63        | 12.37     |

26 interrogatives have been identified from the experimented corpus. Baseline System based on Naïve Bayes classifier has achieved 80.65% accuracy. Using all the features together i.e. Lexical, Syntactical and Semantic, gives the best performance as accuracy achieved is 87.63%.

#### V. Dialogue Based Question Answering System In Telugu

A dialogue based Question Answering system has been described for Telugu Railway information. The question Answering system resulted in precision of 96.34 and dialogue success rate 83.96%.

##### A. System Architecture

In this keyword based approach the query analyzer analysis the input query. The major components of the system are described below [22] [23]:

1. *Knowledge Base*: Words occurring in the database query for Railway information system includes words describing train name, reservation class, station name, and date/period of journey. Domain independent ontology is stored in the knowledge base.
2. *Query Analyzer*: It consults the domain-dependent ontology for recognizing tokens and keywords.
3. *Query Frame Decision*: In this module the keywords in the input query are detected. Appropriate query frame is identified based on the tokens and keywords.
4. *Dialogue Manager*: Main function of this module is to control the flow of dialogue by considering that how the system should respond to a user request and the other components in the system.
5. *SQL Generation*: After the selection of the query frame, the corresponding procedure for SQL query generation is called.
6. *Answer Generation*: After the generation of SQL statement for the input query, it is triggered on the database and the extracted information is used to represent the answer.

##### B. Evaluation

For evaluating this system queries were taken from Telugu speaking persons. Two evaluation metrics are considered: Precision and Dialogue Success Rate. 26 sets of dialogue consisting of 95 questions were used. Dialogue Success Rate and Precision are described below:

Dialogue Success Rate for each set = No. of answers generated by system/ No. of turns issued by user.

Dialogue Success Rate =  $(\sum \text{Dialogue success rate for each set} / \text{No. of sets of dialogues}) * 100$

Precision =  $(\text{No. of correct answers given by the system} / \text{No. of answers given by the system}) * 100$

The total Dialogue Success rate for 26 sets was 21.83. Thus, Dialogue Success Rate =  $(21.83/26)*100 = 83.96\%$ .

From total of 95 questions, system generated answers for 82 questions of which 79 were right.

Precision =  $(79/82)*100=96.34\%$ .

#### VI. A Query Answering System For E-Learning Hindi Documents

Question Answering System for Hindi documents is developed which will be relevant for large no. of persons using Hindi as primary language. The user will be able to access information from E-learning documents in a user friendly way.

##### A. Methodology and Architecture

The main components included in the architecture of the system are explained below [24]:

1. *Automatic Entity Generator*: In this module the entities are identified in particular course to which the user wants to put questions. The system administrator on the server gives the directory of files as input. The module then goes through the main headings and the sub-headings of the text files finding out the domain specific entities. To remove the elementary words Word filtering is done. Entity file stores the output for further use.
2. *Question Classification*: In this module the interrogative words are identified and questions are put into categories such as questions that require reasoning, questions that ask for numerical data, questions requesting certain events,

questions requiring person as answer, questions that need answers from different passages and domain specific category.

3. *Question Parsing*: In this module the domain specific and question specific entities are discovered after removing the stop words. Longest phrase from the question is extracted for further use.
4. *Query Formulation*: In this module the question is transformed to query which is then fed into the retrieval engine to extract answers. To recognize the domain specific entities the system uses an entity file. And build a hash table of these entities. Keywords are given maximum weightage of 2, stop words are given weightage 0 and rest is given weightage 1.
5. *Answer Extraction*: An information retrieval engine is needed to extract passages from the collection of documents. The answer to the query is the location where there is local similarity to the query [25].
6. *Answer Selection and Presentation*: Answer candidates are the top returned passages which are further processed to select those answer passages that will be presented to the user. Top three answers are presented along with the links to the concerned location in relevant document.

### B. Experiments and Results

Hindi Unicode documents are used that are available on site of IIT Hyderabad. Two domains are considered i.e. agriculture and science and documents containing these domains are picked up. Total 60 questions were selected for evaluating the system.

The result is shown in Table V. The system directly answered 75% questions and failed to get correct answers of 13% questions. By extending the classification rules and resources the failure can be solved.

TABLE V  
EXPERIMENTAL RESULTS OF THE SYSTEM

| Data set domain | Question Type | No. of questions | Failed | Accuracy (%) |
|-----------------|---------------|------------------|--------|--------------|
| Agriculture     | Type 1        | 15               | 3      | 66.7         |
|                 | Type 2        | 15               | 2      | 80.0         |
| Science         | Type 1        | 15               | 2      | 73.3         |
|                 | Type 2        | 15               | 1      | 80.0         |
| Overall         |               | 60               | 8      | 75.0         |

The implementation of the system is done in Visual C++. The system is based on searching in context by using similarity heuristic and utilizes syntactic and partial semantic information.

## VII. Multilingual Restricted Domain Question Answering System (Bengali And Telugu)

In this system user specifies a query by starting a dialogue with the system. Every question is received by the language specific shallow parser. Tagging of the input query is semantically done using the domain ontology present in the linguistic models. The tagged words are accumulated into chunks. Topic of the query or query frame may be determined by the keyword chunks or information chunks. The domain model includes the words that are discovered in the input query in each language describing train name, station name etc. together with the type of the chunk. The inflections, post-positions or route words and the keywords for handling the inflected words and to identify the query topic unambiguously are included in linguistic model. The user details like name, service, designation, office and home address and the journey class are included in user model. To get missing information in the user query from the user model or the dialogue history or to generate a sub dialogue with the user, Dialogue Manager (DM) is used [26]. After getting the required information, SQL generation procedure corresponding to the query frame is called to generate all necessary SQL statement(s) using the information words [27]. DM also keeps track of the anaphoric / elliptical queries from the user that constitutes the dialogue using dialogue history information. The correct answers are extracted from the database by using the SQL statements. This result is forwarded to answer generator via DM to produce a natural language answer [23]. The Answer Generator consults the language-specific Answer/Response Template base to retrieve the suitable answer templates and also by the DM to extract the valid response templates.

### A. Evaluation

The evaluation of the system is done with and without the dialogue management. The system without dialogue management means one question and one answer. Precision and Recall are the measure used for evaluation. First of all the results for system without dialogue management are as follows:

1. *Bengali Query System*: 70 questions were used for evaluating system. System generated answers for 64 questions and out of 64, 56 were correct and for 8 questions system generated no answer. Therefore, Precision =  $(56/64)*100 = 87.50\%$  and Recall =  $(56/70)*100 = 80.00\%$ .
2. *Telugu Query System*: System is evaluated by using 132 questions and generated answers for 127 questions out of which 124 were correct and for 3 questions generated wrong answers. Therefore, Precision =  $(124/127)*100 = 97.63\%$  and Recall =  $(124/132)*100 = 93.93\%$ .

By evaluating the system with dialogue management we get following results:

1. *Bengali Query System*: The total dialogue success rate for 24 sets was 17.5. Dialogue Success Rate =  $(17.5/24) = 72.91\%$ . For 58 questions, out of which 49 questions were generated by the system and 41 were correct. Hence Precision =  $(41/49)*100 = 83.67\%$ .
2. *Telugu Query System*: The total dialogue success rate for 32 sets was 28.5. Dialogue Success Rate =  $(28.5/32)*100 = 89.06\%$ . For 62 questions, 57 answers were generated by the system of which 55 were correct. Hence, Precision =  $(55/57)*100 = 96.49\%$ .

#### VIII. Performance Analysis Of The Above Mentioned Question Answering Systems

Performance of the above mentioned Question Answering Systems is shown in Figure 2 and the value of performance metrics for all systems is shown in Table VI. PRASHNOTTAR: Hindi QA system' is considerable if used for four mentioned class of questions: 'when', 'where', 'what time' and 'how many'. Semantic approach with probability distribution scenario, if considered, for the algorithm, can result in a more scalable system in terms of variety of questions asked. Furthermore, the system can be extended to searching dynamic dataset (available on internet) instead of searching a static dataset. This system achieved overall accuracy of 68%.

TABLE VI  
DIFFERENT QUESTION ANSWERING SYSTEM AND VALUES OF THEIR EVALUATION METRIC

| Method                            |                             |                                | Performance Evaluation Metric and its Value | No. of Questions on which it is Tested |
|-----------------------------------|-----------------------------|--------------------------------|---|--|
| PRASHNOTTAR                       |                             |                                | Accuracy (68%)                              | 60                                     |
| Hindi-English Cross Lingual       |                             |                                | MRR (25%)                                   | 56                                     |
| Bengali                           |                             |                                | Accuracy (87.63%)                           | 1100                                   |
| Dialogue based in Telugu          |                             |                                | Precision (96.34%)                          | 95                                     |
|                                   |                             |                                | Dialogue Success Rate (83.96)               |  |
| E-learning Hindi Documents        |                             |                                | Accuracy (75%)                              | 60                                     |
| Multilingual (Bengali and Telugu) | Without Dialogue Management | Bengali                        | Precision (87.50%)                          | 70                                     |
|                                   |                             |                                | Recall (80%)                                |  |
|                                   |                             | Telugu                         | Precision (97.63%)                          | 132                                    |
|                                   |                             |                                | Recall (93.93%)                             |  |
|                                   | With Dialogue Management    | Bengali                        | Precision (83.67%)                          | 58                                     |
|                                   |                             |                                | Dialogue Success Rate (72.91%)              |  |
| Telugu                            |                             | Precision (96.49%)             | 62  |  |
|                                   |                             | Dialogue Success Rate (89.06%) |   |  |

A cross-lingual question-answering (CLQA) system for Hindi and English was developed in one month. Main sources of errors in the system were evaluated which suggested several ways for improving the accuracy of the CLQA system. This system achieved MRR of 0.25 which is satisfactory. This can be further improved by adding more question types and working on the type "Why". Performance of the system needs to be evaluated on more data set and built a system to that level so that it can participate in different workshops held in Question Answering System in different languages by TREC.

In Bengali System different interrogatives for Bengali language are studied and classified into three categories. Baseline System based on Naïve Bayes classifier has achieved 80.65% accuracy. Using Lexical, Syntactical and Semantic features together gives the best performance as accuracy achieved is 87.63%.

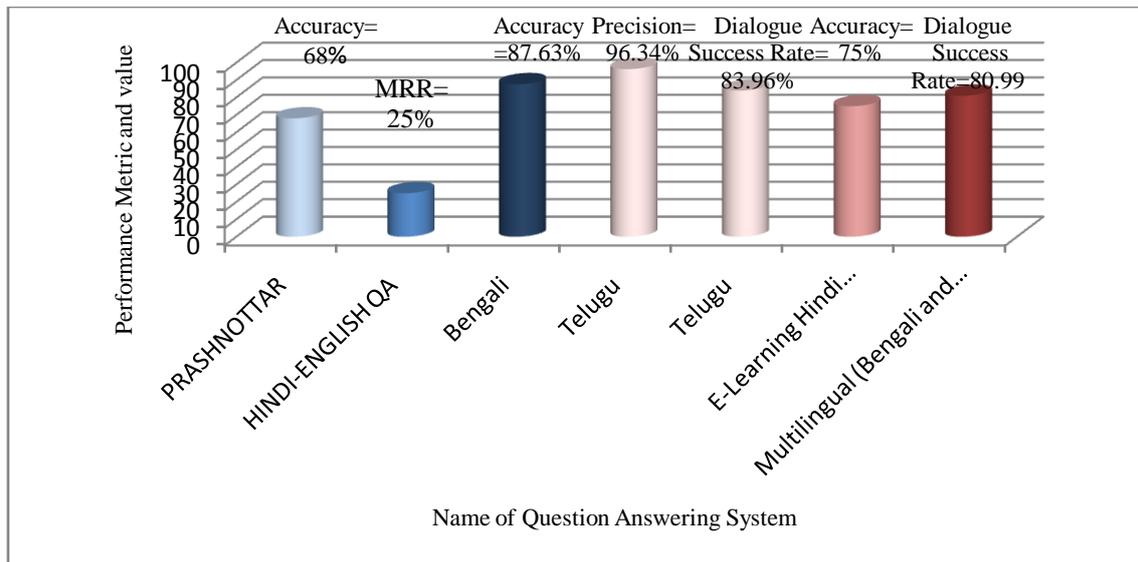


Fig. 2: Performance Analysis of Question Answering System in Indian Languages

In Dialogue based Telugu Question Answering System, two performance evaluation metrics named Precision and Dialogue Success Rate were used. For 26 sets of dialogues, Dialogue Success Rate was 83.96% and for 95 questions, system generated answers for 82 questions and out of which 79 were correct. Thus, Precision = 96.34%.

A Query Answering System for E-learning Hindi Documents directly answered 75% of the questions. The user was directed to the relevant documents for 12% questions and the system failed to get correct answers for 13% questions. System can be improved further by extending the classification rules and the resources. Multilingual Restricted Domain Question Answering System with dialogue management for Bengali and Telugu separates the dialogue control from the application logic and provides a dialogue manger that is portable and user friendly. The system by considering the dialogue management control gives Precision (83.9%) and Dialogue Success Rate (72.91%) for Bengali Query system and Precision (96.49%) and Dialogue Success Rate (89.06%) for Telugu Query System. We have calculated the average value for Dialogue Success Rate and it is 80.99% as shown in Figure 2.

### IX. Conclusion

We analyzed that among the above described Hindi Question Answering Systems, second system is built in one month and is having MRR of 0.25 which is satisfactory by considering the time in which it is built. First system that is PRASHNOTTAR gives the accuracy of 68.00% which is also satisfactory and the system can be further improved by elaborating the different question types and evaluating the system on a large data set. The third system that we studied focuses on the Question Classification module of Question Answering. It show us that how the accuracy of the system can be improved. In this when only lexical, syntactical and semantic features are used and the different classifiers are applied, system achieves accuracy of 87.63%. In fourth system that is Dialogue based Question Answering System in Telugu, Precision = 96.34%. In this system performance is evaluated on a restricted no. of questions and can be improved by extending certain features in it. In fifth system that is a Query Answering system for e-learning Hindi documents achieves accuracy of 75% and can be further improved by employing better methods for extracting semantic information to increase accuracy. The last system, Multilingual Restricted Domain Question Answering System with dialogue management achieves an average Dialogue Success Rate of 80.99% and can be further improved by converting speech input to textual query.

### ACKNOWLEDGMENT

I would like to express my sincere gratitude to Mr. Vishal Gupta, Assistant Professor of Computer Science and Engineering Department in UIET, Department of Panjab University Chandigarh for his immense help and guidance in accomplishing this task.

### REFERENCES

- [1] S. Sahu, N. Vasnik, and D. Roy, "Prashnottar: A Hindi Question Answering System", *International Journal of Computer Science and Technology*, Vol 4, pp. 149-158, 2012.
- [2] M. V. Vera, E. Motta, and J. Domingue, "AQUA: An Ontology-Driven Question Answering System", *American Association for Artificial Intelligence*, 2003.
- [3] W. Zhang, T. Yoshida, and X. Tang, "TFIDF, LSI and Multi-word in Information Retrieval and Text Categorization", in *Proceedings of International Conference on Systems, Man and Cybernetics*, IEEE, 2008.
- [4] S. Dave, P. Bhattachary, and D. Klakowya, "Knowledge Extraction from Hindi text", in *Journal of Institution of Electronic and Telecommunication engineers*, 18(4).

- [5] H. Gahlot, A. A. Krishnarao, and D. S. Kushwaha , “Shallow Parsing for Hindi - An extensive analysis of sequential learning algorithms using al arge annotated corpus” , in *Proceedings of IEEE International Advance Computing Conference Patiala, India, 6-7 March 2009* .
- [6] M. E. Sucunuta and G. E. Riofrio, “Architecture of a Question-Answering System for a Specific Repository of Documents”, in *Proceedings of 2nd International Conference on Software Technology and Engineering (ICSTE)*, 2010.
- [7] C. Yuan and C. Wang, “Parsing Model for Answer Extraction in Chinese Question Answering System”, IEEE, 0-7803-9361-9105, 2005.
- [8] E. Voorhees and D. Tice, “Building a question-answering test collection”, in *Proceedings of the 23<sup>rd</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, New York, 192-199, 2000.
- [9] S. Harabagiu, M. Pasca, and S. Maiorano, “Experiments with open-domain textual question answer”, in *Proceedings of the International Conference on Computational Linguistics*, 292-298, 2000.
- [10] S. Sekine, K. Sudo, Y. Shinyama, C. Nobata, K. Uchimoto, and H. Isahara, “ NYU/CRL QA system, QAC question analysis and CRL QA data”, in *Proceedings of the NTCIR Workshop 3 Meeting Question Answering Challenge (QAC-1)*, 79-86, 2002.
- [11] E. Hovy, U. Hermjakob, and D. ravichandran, “A question/answer typology with surface text pattern”, in *Proceedings of the Human Language Technology Conference*, 2001.
- [12] S. Sekine, K. Sudo, and C. Nobata, “Extended named entity hierarchy”, in *Proceedings of the Language Resource and Evaluation Conference (LREC)*, 2002.
- [13] G. Mann, “Fine-grained proper noun ontologies for question answering”, in *SemaNet: Building and Using Semantic Networks*, 2002.
- [14] D.M. Bikel, S. Miller, R. Schwartz, and R. Weischedel, “Nymble: A high-performance learning name-finder”, in *Proceedings of the Applied Natural Language Processing Workshop*, 1997.
- [15] S. Sekine, K. Sudo, and C. Nobata, “Extended named entity hierarchy”, in *Proceedings of the Language Resource and Evaluation Conference (LREC)*, 2002.
- [16] S. Banerjee and S. Bandyopadhyay, “Bengali Question Classification: Towards Developing QA System”, in *Proceedings of the 3<sup>rd</sup> Workshop on South and Southeast Asian Natural Language Processing (SANLP)*, pages 25-40, Mumbai, December 2012.
- [17] X. Lee and D. Roth, “Learning Question Classifiers”, in *Proceedings of 19<sup>th</sup> international conference on Computational linguistics*, pages 1-7, Association Computational Linguistics, 2002.
- [18] P. Blunsom, K. Kocik and J. R. Curran, “Question Classification with log-linear models”, in *Proceedings of the 29<sup>th</sup> annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pages 615-616, NY, USA, 2006.
- [19] Z. Huang, M. Thint, and Z. Qin, “Question Classification using head words and their hypernyms”, in *Proceedings of EMNLP*, pages 927-936, 2008.
- [20] D. Zhang and W. S. Lee, “Question Classification using Support Vector Machines”, in *SIGIR*, 2003.
- [21] J. Silva, L. Coheur, A. Mendes, and A. Wichert, “From Symbolic to sub-symbolic information in Question Classification”, *Artificial Intelligence Review*, 35(2):137-154, 2011.
- [22] R. R. Reddy and S. Bandyopadhyay, “Dialogue based Question Answering System in Telugu”, in *EACL Workshop on Multilingual Question Answering*, 2006.
- [23] F. E. Annika and J. Arne, “Dialogue and Domain Knowledge Management in Dialogue Systems”, in *Proceedings of 1<sup>st</sup> SIGIDIAL Workshop at ACL*, 2000.
- [24] P. Kumar, S. Kashyap, and A. Mittal, “A Query Answering System for E-Learning Hindi Documents”, in *South Asian Language Review*, Vol. 13, January- June, 2003.
- [25] O. D. Kretsar and A. Moffat, “Effective Document Presentation with a Locality-Based Similarity Heuristic”, in *Proceedings of 22<sup>nd</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp 113-120, San Francisco, 1999.
- [26] P. Pakray, “Multilingual Restricted Domain QA System with Dialogue Management (Bengali and Telugu) “, Thesis Report in partial fulfillment of Master of Computer Science and Engineering in Jadavpur University, Kolkata, 2007.
- [27] F. E. Annika, “A Survey of Knowledge Sources in Dialogue Systems”, in *Proceedings of the IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, pages 41-48, Murray Hill, New Jersey, International Joint Conference on Artificial Intelligence, 1999.