INTELLIGENT MCQ OPTION GENERATOR AND ANSWERING SYSTEM FOR BANGLA LANGUAGE

RAZIA MARZIA ASHEQUE SIDDIQUE
(BSc Engg., MIST)

A PROJECT SUBMITTED
FOR THE DEGREE OF MASTER OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MILITARY INSTITUTE OF SCIENCE AND TECHNOLOGY

2020

APPROVAL OF BORAD OF EXAMINERS

The project titled "INTELLIGENT MCQ OPTION GENERATOR AND ANSWERING SYSTEM FOR BANGLA LANGUAGE" submitted by Razia Marzia Asheque Siddique, Roll No: 1014140006, Session: 2014-15 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of Master of Engineering in Computer Science and Engineering on 06 July 2020.

BOARD OF EXAMINERS

1._____

Dr. Mohammad Nurul Huda   Chairman
Professor and MSCSE Director
Department of Computer Science and Engineering
United International University (UIU)
Dhaka-1212, Bangladesh.

2._____

Md. Afzal Hossain   Member (Internal)
Senior Instructor
Department of Computer Science and Engineering
Military Institute of Science and Technology (MIST)
Mirpur Cantonment, Dhaka - 1216.

3._____

Lt Col Dr. Muhammad Nazrul Islam   Member (Internal)
Associate Professor (Instructor Class A)
Department of Computer Science and Engineering
Military Institute of Science and Technology (MIST)
Mirpur Cantonment, Dhaka - 1216.

4._____

Dr. Md Forhad Rabbi   Member (External)
Associate Professor
Department of Computer Science and Engineering
Shahjalal University of Science and Technology (SUST)
Sylhet.

DECLARATION

I hereby declare that this project is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the project.

_____

Razia Marzia Asheque Siddique

Department of Computer Science and Engineering

Military Institute of Science and Technology

6 July 2020

SUMMARY

This study incorporates automatic question answering (QA) system, which is one of the most important sub-fields of Natural Language Processing (NLP). Here, the system comprises two phases. In the first phage, the system generates four automatic options of the answer for the input question based on the given topic. On the other phase, the system provides multiple choice questions where an examinee can answer the questions within a time constraint. Based on the examinees answers he/she will get a score. It may be mentioned that the question may be multiple choice and descriptive, but this report only deals with factoid or multiple-choice questions. Both parts of the research were implemented for Bangla Language which is one of the low resource languages. Because of the scarcity of resources, the research based on Bangla Language does not reach up to the mark. Nowadays the availability of internet resource for the Bangla Language attract researchers for doing research at different dimensions though the accuracy depends on whether the corpus is domain oriented or generalized. For domain-oriented corpus, the accuracy of the application is very good but the application based on the generalized corpus provides accuracy which is not satisfactory in some cases. It is observed from the system investigation that the first and the second phases provide the low and high accuracy respectively based on the domain-oriented data. The originality of the project is phase-I which generates four options based on the input question for Bangla Language by using the n-gram language model.

Keywords: QA system, Answer Extraction, Information Retrieval, Distractor Generation, Natural Language Processing (NLP).

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

QAS	Question Answering System
NLP	Natural Language Processing
NLU	Natural Language Understanding
NLI	Natural Language Interpretation
NLG	Natural Language Generation
IR	Information Retrieval
MCQ	Multiple Choice Question
TOEFL	Test of English as a Foreign Language
IELTS	International English Language Testing System
MCQAS	Multiple Choice Question Answering System
DQAS	Descriptive Question Answering System
BNLP	Bangla Natural Language Processing
BIC	Bangla Informative Chatbot
TREC	Text Retrieval Conferences
SGD	Stochastic Gradient Descent
TF	Term Frequency
IDF	Inverse Document Frequency
GUI	Graphical User Interface
ML	Machine Learning
DL	Deep Learning

DT	Decision Tree
NN	Neural Network
POS	Part of Speech
AI	Artificial Intelligence
RNNs	Recurrent Neural Networks
CNNs	Convolutional Neural Networks
BLTK	Bangla Language Toolkit
BLE	Bangla Language Expression

CHAPTER 1

INTRODUCTION

Bengali or *Bangla* is a language of the Indo-Iranian subfamily of the Indo-European language family Indic group. It is the official and most widely spoken language in Bangladesh, and second most widely spoken of India's 22 scheduled languages behind Hindi. With around 228 million native speakers and a further 37 million second language speakers, Bangla is the fifth most frequently spoken native language and the seventh most frequently spoken language in the world[1].

However, there are very few works in computational language with Bangla compared to English. In Bangla, there are some automated question answering (QA) systems that will answer our queries by processing the Bangla language. A program that can handle the user's natural language in such a way that it is difficult to distinguish between the real human being and the computer is known as an automated question answering system. Question Answering (QA) system is one of the text mining techniques in Natural Language Processing (NLP). Two types of QA systems are available for different foreign languages specially in English which are:

1. **Factoid/ Multiple Choice Question (MCQ)**

The simplest kind of question answering is addressing factoid questions. because the name implies, the answers to factoid questions are simple facts which will be found briefly text strings. Factoid questions, therefore, are questions whose answers are found in small spans of text and correspond to a specific, easily characterized, category, often a named entity.

---

[1] **https://en.wikipedia.org/wiki/Bengali_language**

The answer processing system of a factoid question is much easier than the complex one. Some examples of factoid questions are:

When the liberation war occurred in Bangladesh?

Where Shundarban is located?

Some sample Bangla factoid questions are:

বাংলাদেশের প্রধানমন্ত্রী কে ?

বাংলদেশের জনসংখ্যা কত ?

Each of these questions require a single information which can be answered directly with a text string that contain the time, location, name of person, a numeric value respectively.

2. Descriptive/ Complex

The complex form of question answering system. Here description or cause and effect can be asked. Some examples of factoid questions are:

Why the liberation war occurred in Bangladesh?

How can we make Shundarban better?

Some sample Bangla factoid questions are:

বাংলাদেশের বর্তমান প্রধানমন্ত্রীর ছোটবেলা কেমন ছিল?

বাংলদেশের আশঙ্কাজনকহারে ক্রমবর্ধিত জনসংখ্যা রোধে কি কি পদক্ষেপ নেয়া যায়?

Each of these questions require descriptive answer which can't be answered using single word on single line.

A lot of QA systems are developed over the years and a lot of researches has been done for question answering system. Though most of the QA systems are for the English language as data is very available. Research for the Bangla QA system for Bangla language is also going on though it is a challenging job.

1.1 Motivation

MCQ systems are already used in TOEFL (Test of English as a Foreign Language), IELTS (International English Language Testing System) etc. but in Bangladesh most of the cases the system is handed manually instead of computer-based intelligent system. Because of this the system proposed in this project deals with online Bangla MCQ system which comprises two parts: option generation from given Bangla document and finding answer based on given input question in Bangla. Since Benchmark training data set for Bangla language is rarely available like other languages and there is no specific rule for Bangla sentence construction. So, data pre-processing as well as keyword generation becomes very difficult for Bangla QA system. The most challenging part of developing an MCQ based examination system for Bangla language is finding the options automatically from the given document. So, further research is required in this field

.

1.2 Objective

The objective of this project is to create a platform where multiple choice question will be given and options/distractor (along with the answer) of the question will be generated automatically from the corpus given by the user.

On the distractor generation which is also known as options of MCQ, it explores the corpus in-order to retrieve distractor candidates belonging to the same context. The platform presents a user interface to review and edit the multiple-choice question options which are generated by the automated pattern discovery system, where the question will be given by the user.

1.3   Organization of the Remaining Chapters

While the introductory chapter has presented the basic introduction to Question Answering (QA) system, the motivation of this project and objective of this project, the following chapters will attempt to convey further details and approach to build Bangla MCQ system where options, as well as the answer, will be generated automatically.

The next chapter *Background Study* highlights Natural Language Processing, its components, business use of NLP. It also describes various type QA system. The chapter *Literature Review* highlights the research about MCQ generation system also factoid-based QA system. The chapter *Methodology* describes the detailed approaches of NLP tasks, proposed system methodology, detailed analysis of other QA systems in English and Bangla and system architecture. The chapter *Experiment and Result* describes the evaluation of the proposed system.

Finally, the last chapter *Conclusion* summarizes the main outcome of this project, practical usage of this project, challenges on working with Bangla QA system, and future work of this project.

CHAPTER 2

BACKGROUND STUDY

A lot of researches have been done and are also ongoing on a question answering system in different languages. Besides, there are also works on question classification, question features, question taxonomies, and answer extraction which are the sub-parts of question answering systems. Nowadays there is a remarkable increase in the research for Bangla language also in the field of Natural Language Processing (NLP). Question answering systems being one of the hot topics the research works for question answering system in Bangla language is also increasing at this time.

2.1   Natural Language Processing

Natural Language Processing or NLP is an artificial intelligence field that enables the machines to read, understand, and extract meaning from human languages. It is a discipline that deals with the relations with data science and human language, and are scaling to many industries. Today the NLP is progressing due to the massive changes in data access and the rise in computing capacity, which enables practitioners to produce significant outcomes in fields such as healthcare, education, media, finance, and human resources[2]. The input and output of NLP system can be *Speech* and *Written Text*. The components of NLP are[3]:

    1. Natural Language Understanding (NLU)

    2. Natural Language Generation (NLG)

---

[2] **https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1**
[3] **https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_natural_language_processing.htm**

2.1.1   Natural Language Understanding (NLU)

Natural language understanding (NLU) is also known as natural language interpretation (NLI). The interpretation of the natural language interprets the meaning conveyed by the consumer and classifies it into proper intent. For example, understanding one another is relatively easy for people who speak the same language, although incorrect pronunciations, vocabulary choices, or phrases may complicate this. NLU is responsible for this task of distinguishing what is meant by applying a variety of processes such as text categorization, content analysis, and sentiment analysis that allows the computer to solve complex inputs.

2.1.2   Natural Language Generation (NLG)

This is the method of generating coherent phrases and sentences from any internal expression in the form of a natural language. It performs following tasks:

- **Text planning**, includes retrieving the relevant content from knowledge base.

- **Sentence planning**, includes choosing required words, forming meaningful phrases, establishing tone of the sentence.

- **Text Realization**, includes mapping sentence plan into sentence structure.

2.2   Businesses Use of Natural Language Processing

Human interaction is that the actuation of most businesses. Whether it's a brick-and-mortar store with inventory or an outsized SaaS brand with many employees, before, during, and after a purchase, customers and businesses have to communicate. This means NLP has endless chances to move in and change the way an organization runs. This is particularly true of huge companies seeking to maintain track of, promote, and evaluate thousands of consumers experiences to enhance their product or service. There's a large number of

different NLP applications. The examples for the frequent use of NLP for business are the following[4]:

- Form Spell Check
- Search Auto Complete
- Search Auto Correct
- Smart Search
- Machine Translation
- Messenger Bots
- Virtual Assistants
- Knowledge Base Support
- Customer Service
- Automation Alexa Skills
- Survey Analytics
- Social Media Monitoring
- Descriptive Analytics
- Automatic Insights

2.2   Question Answering System

An automated approach that retrieves correct responses to the questions asked by human language is thought of as Question Answering (QA) Systems. The essential thought behind QA system is to help man-machine interaction. There are two types of question answering system:

1. Descriptive Question Answering System.

2. Multiple Choice Question Answering System.

The focus of this paper is about MCQAS also known as factoid question answering system.

2.3.1 Descriptive question answering system

Normally, a Descriptive Question Answering system is divided into three steps:

1. Question analysis,

2. Passage retrieval, and

3. Answer extraction with text summarization

The most time-consuming part of DQAS is sentences selection for answering the question.

2.3.2 Multiple choice question answering system (MCQAS)

Normally, a Multiple Choice Question Answering system is divided into four steps:

1. Question analysis

2. Passage retrieval

3. Answer extraction

4. Option or distractor generation

The most time-consuming part of MCQAS is option generation. There are some factoid question answering systems in English language, where very few generates options regarding specific topic.

CHAPTER 3

LITERATURE REVIEW

The world now has a big eye on Natural Language Processing (NLP) field. Someone is researching text processing, others are researching text understanding, summarizing text, finding answers to specific questions, and generating questions, etc. For NLP all of these activities are a number. Some of the previous works on the generation of questions are listed below.

Coniam [1] put forth one of the earlier MCQ generation attempts. In the different phases of development, they used word frequencies for an analyzed corpus. To build the test items, they combined parts of speech and word frequency of each test item with similar word class and word frequency choices.

Mitkov and Ha [2] and Mitkov et al. [3] used NLP techniques such as shallow parsing, term extraction, sentence transformation and semantic distance computing in their work to automatically generate MCQ semi from an electronic text. They used frequency count to term extraction from the text, generated stems using a set of linguistic rules and selected distractors.

Brown [4] developed a system to generate vocabulary assessment questions automatically. To create the questions as well as the distractors they used WordNet to find meaning, synonym, antonym, hypernym and hyponym.

Aldabe et al. [5] and Aldabe and Maritxalar [6] developed Basque-language MCQ generation systems. They divided the task into six phases: text selection (based on learners and text length), marking blanks (manually), distractor generation, distractor collection, learner evaluation and item analysis.

Papasalouros et al. [7] suggested an ontology-based approach for the automatic creation of an MCQ framework. Agarwal et al. [8]- [9] introduced a system that uses discourse connectives to automatically generate questions from natural language text.

Plamondon et al. [10] developed a system where the question had to be asked in English, the collection of documents was in English and the answer was extracted in English. Le An Ha and Victoria Yaneva introduced a fully automatic system to produce suggestions for distractors for multiple-choice questions used in high-stakes medical examinations. As an input, taking a question steam and the correct answer the system produces a list of recommended graded distractors based on their stem similarity and the correct response. They used the information retrieval methods combination for building this system. They presented the assessment as a predictive task in which they tried to "predict" the distractors produced by humans used in large sets of medical questions, i.e. if a distractor developed by their method is sufficiently good, it is likely to be on the list of distractors produced by the human item-writers. The results reveal that combining concept embedding with approaches to information retrieval significantly improves the generation of plausible distractors and allows us to match around 1 in 5 of the distractors produced by human beings. The approach proposed in this paper is generalized to all scenarios where concept is referred to by the distractors [11]- [12].

S. Mann introduced a short answer extractor using Mutual Information (MI) statistics trained to estimate question class / semantic tag correlation on un-annotated trivia questions [13].

Somnath Banerjee and Sivaji Bandyopadhyay worked on the classification of questions step towards the construction of a question answering system in Bangla which is the important first step. In 2012 They suggested a single-layer taxonomy comprising of just nine course-grained classes. They also studied and categorized the Bengali-language interrogations. The proposed automated classification research is based on the various techniques of machine learning [14]. In 2013 they proposed fine-grained classes using bagging and boosting on lexical, syntactical and semantic features of Bengali questions for the question classification task [15].

Haque et al. [16] developed the transliteration-based question answering framework and the table look-up as a medical domain GUI. The system is in no way a complete QA system; however, it provides the basis for the implementation of a complete Bengali QA system. This implementation required only the generation of medical-domain queries and only simple questions ('Wh' questions).

Somnath et al. [17] developed a factoid-based question answering system for Bangla language. There was also a proposal to extract and rank relevant sentences. For the Bengali question answering system it is also suggested extraction strategy of the ranked answers from the relevant sentences. They called their system "BFQA" which has a three-component pipeline architecture, namely a) query analysis, b) extraction of sentences, and c) extraction of answers.

Fasihul Kabir, Sabbir Siddique, Mohammed Rokibul Alam Kotwal, Mohammad Nurul Huda [18] introduced a system which categorized Bangla document using Stochastic Gradient Descent (SGD) classifier. They divided their system into three steps: (1) feature extraction incorporating term frequency (TF) and inverse document frequency (IDF), (2) classifier design using the Stochastic Gradient Descent (SGD) algorithm by learning the distinct features, and (3) performance measure using F1-score.

Md. Aminul Islam, Md. Fasihul Kabir, Khandaker Abdullah-Al-Mamun, and Mohammad Nurul Huda [19] proposed a Bangla word/phrase based QA system where answer is classified.

Shomi Khan, Khadiza Tul Kubra, and Md Mahadi Hasan Nahid from the Department of Computer Science and Engineering of Shahjalal University of Science and Technology developed a QA system using anaphora-cataphora resolution [20].

Md. Kowsher, Farhana Sharmin Tithi, M Ashraful Alam, Mohammad Nurul Huda, Mir Md Moheuddin, Md. Golam Rosul proposed AI-based Bangla Chatbot named "Doly" which is based on machine learning algorithms and Bengali Natural Language Processing (BNLP) [21]. Md. Kowsher, M Ashraful Alam, Md. Jashim Uddin, Md. Rafiqul Islam, Nuruzzaman Pias, and Abu Rayhan Md Saifullah introduced an algorithmic Bangla Informative Chatbot (BIC) based on information that is significant mathematically and statistically [22]. Recently Sourav Sarker, and Syeda Tamanna Alam Monisha students of the Department of Computer Science and Engineering of Shahjalal University of Science and Technology, developed a QA system based on factoid questions [23]. They provided the system with knowledge based and question database, it will be able to extract answers from them.

CHAPTER 4

METHODOLOGY

NLP adopts three basic approaches for performing NLP tasks to understand and interpret human language[5]. All three methods are well known and commonly used in several segments. The evolutionary methods have individual advantages and tend to assist the NLP challenge in achieving better outcomes. The three methods are:

1. Rule-Based Approach.

2. Machine Learning or *Traditional* Approach.

3. Neural Network and Deep Learning.

Short details of these approaches are described below:

4.1   Rule-Based Approach

This is the oldest, most tried and tested methodology that has, by its performance, proved its efficacy. In line with the traditional grammar rules, it is a handmade set of rules focused on linguistic mechanisms that reflect the history of human language development or grammar use. When rules are added to the language, it may give several insights: consider what to decode with meaningless language by merely finding out all of the terms are nouns, or whether verbs end with 'ing' or if a trend identifiable as Python code may be detected. This method is easy to build and helps you to still test how a question can be answered or how it can be accomplished. For example; when utilizing the extension of translation rules and the foundation of a good synonym, performing for different functions and data types can be quickly updated; without any significant adjustments to the core structure.

---

[5] **https://sunartek.com/main-approaches-to-nlp/**

When learning a particular language concept, a rule-based method is perfect because it essentially decodes the linguistic interaction between terms to interpret the message. Advantages of this method are given below:

1. It is easy to adapt

2. It is simple to debug

3. Huge training corpus not required

4. Comprehends the language

5. High perfection

Disadvantages of this method are given below:

1. Skilled developers & linguists required

2. Slow production of the parser

3. Moderate recall (coverage)

4.2   Machine Learning Approach

The 'Traditional' or Machine Learning (ML) approach is an algorithm-based method that provides statistical methods that help the computer learn and understand the language. The system does not need specific programming here, it starts to evaluate the data already represented or the training set (annotated corpus) to build its learning, create its structure and its classifiers. The approach to machine learning involves probabilistic analysis, optimizing probability, and linear classifiers. Because the results are based on phenomena of chance, it provides fewer guarantees that cannot fully observe complicated text and struggle through the butterfly effect. For example, A very hard cash or a little amount of

recent data can drastically alter or end in a distinct outcome, leading the system even to its author acting unpredictably. It is ideal for a task-based activity such as document classification or word clustering which has a lot of data points and helps the computer to learn the word given in the task itself from its statistical clues. Advantages of this method are given below:

1. It can scale with minimal effort

2. Learnability without distinct programming

3. Rapid development if dataset available

4. High recall coverage

Disadvantages of this method are given below:

1. Training corpus with annotation needed

2. Difficult to debug

3. Zero understanding of the language

4.3   Neural Network and Deep Learning Approach

Deep learning is a machine learning branch and it is the re-branded name for neural networks a family of learning new techniques inspired by brain computations. It may be defined as learning mathematical differential function parameterized. The term deep learning comes from the fact that many layers of these differentiable functions are clustered together frequently. While machine learning is based on assumptions and past observations, while deep learning relies on representing the data correctly to be predicted. This encompasses different approaches. The visual system of humans is without a doubt one of

the world's wonders. It's amazing to make sense of what our eyes give us as a supercomputer doing most of the unconscious processing in our brain. The difficulties in the interpretation of visual patterns, such as deciphering a handwritten symbol, is evident while trying to write a computer algorithm because understanding is not as easy to express as precise rules. Neural networks tackle the issue differently. Taking a large pool of handwritten numbers, called examples of training, it builds a system that can learn from those examples of training. To put it simply, the neural network is using the training examples to automatically infer rules for handwritten digit recognition. Besides, by increasing the number or amount of training examples, the network can learn more about handwritten numbers and improve their accuracy. There are two types of neural network:

1. **Convolutional Neural Networks (CNNs):**

CNN has a dedicated architecture that surpasses those vulnerable to word order when extracting local patterns in the data. This can be used in long sentences/documents to define suggestive phrases of words or idioms of up to a fixed length.

2. **Recurrent Neural Networks (RNNs):**

RNN is the specialized model for sequential data. It is used as an input transformer that is trained to provide the CNN with informative representation to work on top of it.

4.4   Proposed System Methodology

For the faster development here, rule based approach is selected for developing the system. As it is a project so the methodology is shown by the flow chart given in figure 4.1.
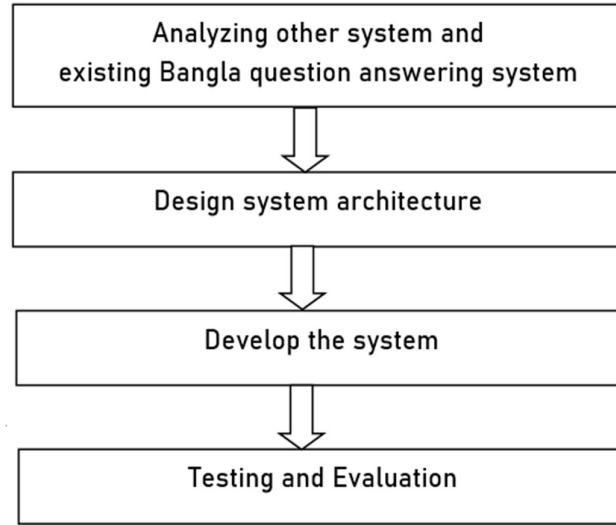
```
┌─────────────────────────────────────┐
│      Analyzing other system and       │
│  existing Bangla question answering system │
└─────────────────────────────────────┘
                    ⇓
┌─────────────────────────────────────┐
│        Design system architecture      │
└─────────────────────────────────────┘
                    ⇓
┌─────────────────────────────────────┐
│            Develop the system           │
└─────────────────────────────────────┘
                    ⇓
┌─────────────────────────────────────┐
│           Testing and Evaluation         │
└─────────────────────────────────────┘
```

Fig. 4.1: Project Methodology

4.5   Analyzing Other System and Existing Bangla Question Answering Systems

Multiple choice questions are an assessment form in which respondents are asked to choose the best possible answer from a list of choices. Generating multiple choice question automatically is a relatively new and important research area and potentially useful in Education Technology. Question Answering systems have been transformed in much parallel with all-natural language processing in the last four decades [24]. Here, we present an earlier work on QA system development and its function. The earliest system was developed in 1959 (in the tradition of the Conversation Machine era), and since the 1960s, an oversized number of QA systems are developed and BASEBALL (Green et al. 1961 in NL DB systems) is one among the foremost popular systems [25]. While able to answer very complex questions, BASEBALL was, not surprisingly restricted to questions about baseball games played in the American League over a season, to

questions about baseball statistics, and most question answering systems were for a long time confined to front-end organized databases. They developed QA system PROTOSYNTHEX in 1963, which allowed users to ask a question in English, accepted statements in (sub-set of English) as input to their database and accepted quotations as a query to the database. And for reading and solving the kind of word problems they create (Problem-solving systems) STUDENT QAs program in 1964 that can read and solve the kind of word problems contained in algebra books of high school. The program is often cited as an early realization of AI in the processing of natural languages [26].

Weizenbaum et al. introduced QA device ELIZA in early 1966 that could communicate with the user) and this interaction can be replicated with a psychologist [27]. It was able to talk about any subject by resorting to very simple rules that detected important words in the input of the individual. In 1972 Terry Winograd developed SHRDLU which offered the possibility of operating a robot in a toy world (the "blocks world") to be simulated with the ability to question the robot about the state of the world [28]. The SCHOLAR QA system was developed in 1973, a formal theory that characterizes the variety of feasible inferences that people use to ask questions about which they are incomplete. This integrates methods into the SCHOLAR system that have been learned from such work [29]. They developed the first automatic question answering system (SAM) in the same year 1973. TRIPSYS (HWIM) was the first QA to understand speech Question was developed after three years in 1976 by Woods ef al., TRIPSYS (HWIM) was created as the framework for a research project in continuous speech understanding- it understands and answers questions about expected and taken journeys, travel budgets and their status, costs of different modes of transport to

various places it is called HWIM [25]. The second well-known QA program (NL DB systems) was LUNAR problem-solving systems built in 1977 [30] to answer questions on the geological study of rocks returned by the Apollo moon missions.

Two QA systems were developed in 1977, the first one is GUS, which was a dialog system for airline booking, the second one was LIFER, which developed questions about U.S. Navy ships [31]. This program used semantic grammar with domain knowledge developed in 1978 to develop a system that deal with story understanding (NL DB systems) QUALM, an application that uses story understanding and this application is responsible for scripts and plans in a very restrictive environment. In 1983, Salton and McGill described the questions answering (QA) systems as providing more direct answers to questions. Finally, Kupiec (1993) used similar but rather simplified WH-questioning models to build a QA program [32].

LILOG, a QA system which also operated on the tourist information domain in a German town built in 1991. In 1993 they began to combine NLP with the use of an online encyclopedia by establishing a MURAX QA system that combines NLP with the use of an online encyclopedia with hand-coded annotations of sources. Through subsequent advances, QAS aimed at making linguistic analysis of the questions in a natural way to capture the expected requirements [33]. Through subsequent advances, QAS aimed at making linguistic analysis of the questions in a natural way to capture the expected requirements.

Due to the introduction of the Text Retrieval Conferences (TREC) question answering track, great progress has been made in the open domain answering questions in the last

90's (Voorhees 2001). Such systems make use of unregulated text as the primary information source. One such method, MASQUE (ANDROUTSOPOULOS et al., 1993), uses logic representation to represent natural language issues and converts the logic query into a database query for the retrieval of intended knowledge from the database. It distinguishes linguistic process from mapping. FAQ Finder (Burke et al., 1997) matched the question list compiled on a knowledge base by statistical similarity and semantic similarity, and for the technique for the understanding of syntax-based natural language. LASSO won the question answering challenge in 1999. The method used to identify problems and to grasp the natural language of syntax. Riloff and Thelen (QUARC) has also developed another QAS in (2000) that allows questions to be divided into several categories and uses a lexical and semantic indication to respond [34] [35].

Subsequently, the emphasis on developing QASs has been shifted to open domain QAS: the TREC Evaluation Project, which since 1999 every year had been underway to administer and query large volumes of data and represent mainly open-domain surveys which respond from unstructured sources. More than two-thirds of factual questions are now answerable by the best systems in this evaluation, the results are described and the evaluation methodology developed by Ellen Voorhees, while Buchholz and Daelemans examine the requirements for answering complex questions which have complete answers or many correct answers [36].

A paper by Lin and Pantel explains a new paraphrase algorithm that make it possible to model questions more precisely in the future. In the fourth paper by Light, Mann, Rilo and Breck, experiments are identified, systemically deciding questions that

respond to sub-problems of components. A list of 200 questions and a series of documents were provided in the first TREC assessment program. The responses in the collections were considered to be present. The maximum response lengths of 50 or 250 characters were permitted [37]. Systems were asked to give five classified lists of responses.

TREC-9 held in 2000 increased the number of questions and the scale of the paper collections. A new complexity regarding answers is found in 2001 in TREC-10. The response time was reduced to 50 words [36]. In the 2002 TREC-11 program the answer to questions was supposed to be correct and short. The assessment campaigns included the list of questions, definition issues and factoid issues in TREC from 2002 to 2007. There were temporary questions in TREC 2005 and TREC 2006, in addition to 75 topics containing different types of questions. TREC 2007 collected documents that include blog compilation and advancement competitions, resulting in an increasing complexity and scope of documents and questions and also increase the complexity of assessment strategies.

In 2000, they began to develop a Unix operating system QA system For Unix operating system answers questions [35]. This system was able to accommodate different types of users through the sentence answer and has a field with a comprehensive knowledge base. In 2001 there were two INSIGHT QA systems, the first using some super-fast patterns and TREC-10 answering the question, and the second being SiteQ's density-based extraction method for obtaining the corresponding parts in the first place and then extracting the exact answers, which can increase the extraction speed considerably.

STARTQA was the first web-based English QA system, which was established in 2002 and the same year Answer bus QA System (ODQA) was designed, which accepted several languages in which the response process was extended to the World Wide Web from the local data source and addressed many of the questions. The first web-based open source Factoid Questioning System was developed after a year of development by QA (ARANEA) and a sophisticated, automated QA system was created in 2003, combining the technical and ontological knowledge of natural language understanding, logical rationality, advanced knowledge extraction techniques and the ability to use the QA system in 2003 [37].

In 2008 they began developing a new QA program with different approaches and a list of questions was asked to return various instances of certain types of information. In 2012, QA Systems were created, which can also be included in this category in order to work with opinions instead of evidence. In 2014 QAKIS was designed by Cabrio et al and FREITAS14 and also INTUI3 by Dima, in 2015 three QA systems were first built by HAKIMOV15 from Usbeck et al [37] and second was QASYO by Hakimov et al.

AnswerBus is an open-domain question answering system [38]. The information concerning the response is collected by the web at sentence level in this framework. The authors used the web documents to extract answers to users' questions through five search engines (*Google, Yahoo, Altavista, WiseNut and Yahoo News*). The current rate of answers to 200 questions from TREC-8 (Text REtrieval Conference-8) is 70.5%.

'BFQA' is the first Bangla automated question answering system developed by Somnath Banerjee, Sudip Kumar Naskar, and Sivaji Bandyopadhyay [17]. The system architecture is given in Figure 4.2:
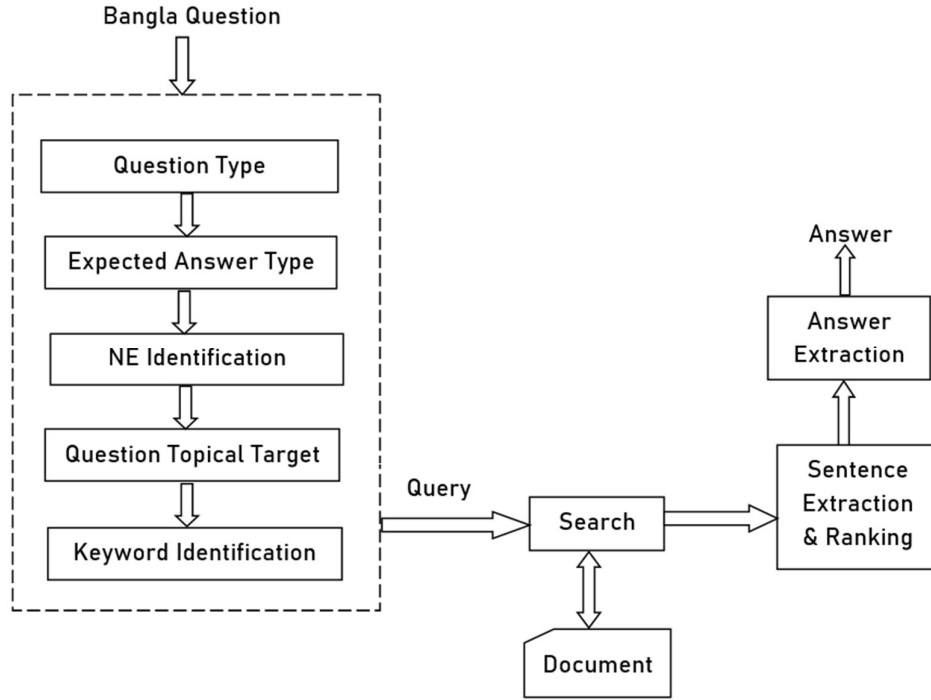

Fig. 4.2: BFQA Architecture

Bangla word/phrase based QA system was proposed by Md. Aminul Islam, Md. Fasihul Kabir, Khandaker Abdullah-Al-Mamun, and Mohammad Nurul Huda [19]. This is the first work on a machine learning approach to Bengali question classification using stochastic gradient descent (SGD) classifier method. They also classified the answer type. They achieved average precision 0.95562 for coarse and 0.87646 for finer.

Then a QA system using anaphora-cataphora resolution is developped by Shomi Khan, Khadiza Tul Kubra, and Md Mahadi Hasan Nahid [20]. Their proposed method is a model in which it is simple for users to get as accurate an answer as possible to their

question easily and reduces the difficulty of using nouns rather than pronoun for the requested answer with regard to the question queries provided for Bangla. It improves the extraction of the answer better than the naive approach.

'Doly', AI-based Bangla Chatbot was developed by Md. Kowsher, Farhana Sharmin Tithi, M Ashraful Alam, Mohammad Nurul Huda, Mir Md Moheuddin, Md. Golam Rosul [21]. They created the train function adapter to train the 'Doly' by encoding our bot data corpus (encoding = "utf8"). To take input, an input adapter was created and an output adapter was created to produce automatic responses to a user's input for output. They also used a machine-learning algorithm such as the search algorithm to find a suitable list of matching results from the corpus, and use the Naïve Bayesian algorithm to generate the correct response from data.

Bangla Informative Chatbot (BIC), developed by Md. Kowsher, M Ashraful Alam, Md. Jashim Uddin, Md. Rafiqul Islam, Nuruzzaman Pias, and Abu Rayhan Md Saifullah [22].Two algorithms demonstrate this paper to find out the lemmatization of Bengali words such as Tire and Dictionary Based Search by Removing Affix (DBSRA), as well as the exact lemmatization compared with Edit Distance. They presented the Bangla Anaphora resolution method using the Hobbs' algorithm to get the knowledge to be correctly expressed. The TF-IDF and Cosine Similarity are established as the behavior of chatbot replying algorithms to find out the exact response from the documents. They implemented a Bengali Language Toolkit (BLTK) and Bengali Language Expression (BLE) that make their task easier to infer. BIC gave 97.22% accurate answer over 19334 questions.

Sourav Sarker, and Syeda Tamanna Alam Monisha developed a QA system for Bangla language which was closed domain based on statistical approach [23]. The accuracy of their system was 66.2% and 56.8% with and without mentioning the object name. The system also hit around 72% documents from which we can extract the answer manually. Again, the sub-part of their system, the question and document classifier, gave the accuracy of 91% and 88.4% respectively.

4.6   System Architecture

Here 'BFQA' architecture is slightly followed but here NE identification and question topic classification is not used. The system architecture is shown in Figure 4.3:

Bangla Topic
⇩
Bangla Question from Input Topic
⇩

```
┌─────────────────────────────┐
│     Question Processing     │
└─────────────────────────────┘
```
⇩
```
┌─────────────────────────────┐
│    Sentence Ranking and     │
│    Extraction from given    │
│            topic            │
└─────────────────────────────┘
```
⇩
```
┌─────────────────────────────┐
│      Answer Extraction      │
└─────────────────────────────┘
```
⇩
```
┌─────────────────────────────┐
│       Other options         │
│        generation           │
└─────────────────────────────┘
```
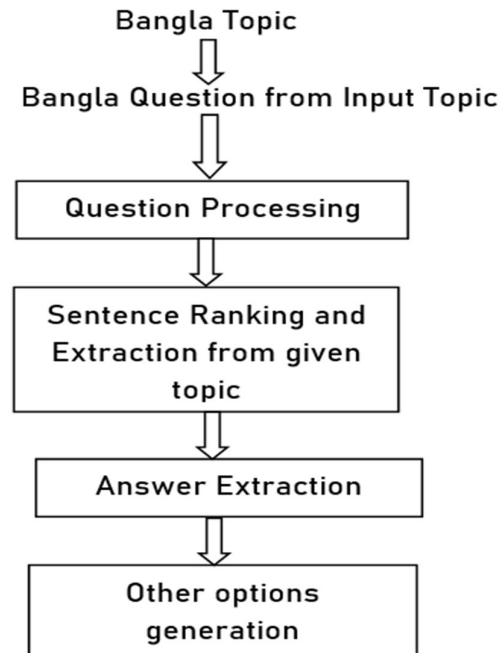
Fig. 4.3: System Architecture

4.6.1 Question processing

Question processing is the pre-processing step. When user gives the input question, we need to process the question. In our system we identify the question type from the given question. Then keywords are extracted from the question. This process is named as question processing as here only focus is question. The flowchart for question processing is given in Figure 4.4.
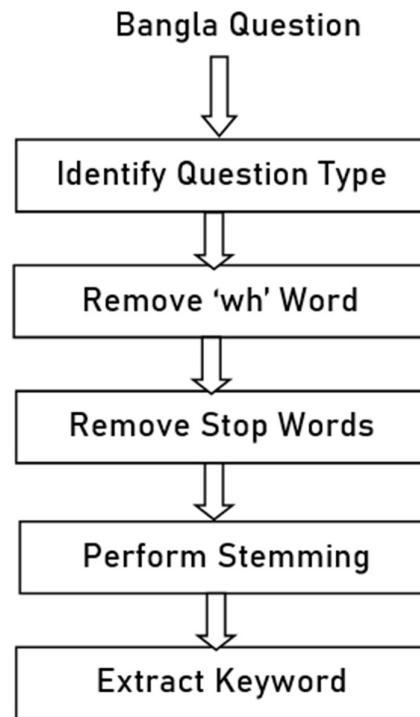

Fig. 4.4: Question Processing

As this system is rule based so we identify question type from our predefined question type list. The list is shown in Table 4.1:

Question Word	Question Type
কে	person
কখন	numeric
কবে	numeric
কোথায়	place

কি	noun
কার	person
কোনটা	noun
কত	numeric
কোনটি	noun
কতটি	numeric
কততম	numeric
কোন	noun
কাকে	noun
কাদেরকে	noun

Table 4.1: Predefined Question List

In computing, stop words are words that are filtered out before or after the natural language data (text) are processed. Although "stop words" usually refers to the most common words in a language, all natural language processing tools do not use a truly universal list of stop words, and not all tools even use such a list. Some examples of Bangla stop word are অথচ,

অথবা, এবং etc.

Stemming is the method of reducing inflected (or often derived) words to their word stem, basis, or root form in linguistic morphology and retrieval of information - generally a written word form. The stem does not need to be identical with the word's morphological origin; it is generally enough that similar words refer to the same stem, even though this stem is not a true root in itself. Research on stemming algorithms has been started since the 1960s. Now rule based Bangla stemmers are available for various programming language. For example, if we perform stemming on the sentence: "পুরকৌশল বিভাগের প্রথম ব্যাচ দিয়ে এমআইএসটির একাডেমিক কার্যক্রম শুরু হয়" then we will get the output as "পুরকৌশল, বিভাগ, প্রথম, ব্যাচ, দিয়, এমআইএস, একাডেমিক, কার্যক্রম, শুরু, হয়''. Algorithms for processing question is shown in algorithm 1 and 2.

Algorithm 1: Question Type Identifying

Input : question, document
Output: questionType
1 questionType ← null
2 **foreach** storedQuestionType in system **do**
3   **foreach** questionWord in question **do**
4   **if** storedQuestionType = questionWord **then**
5   questionType storedQuestionType
6   break
7   **end**
8   **end**
9 **end**

Algorithm 2: Question Keyword Extraction

Input : question, document

Output: questionKeyWordList

1 addKeyword ← false questionKeyWordList ← List **foreach** questionWord in question **do**
2   **foreach** storedStopWord in system **do**
3   **if** storedStopWord = questionWord **then**
4   addKeyword ← true
5   break
6   **end**
7   **end**
8   **if** addKeyword **then**
9   questionWord ← steam(questionWord)
10   **if** questionWord not in questionKeyWordList **then**
11   add questionWord to questionKeyWordList
12   **end**
13   **end**
14 **end**

4.6.2   Sentence ranking

Sentence ranking is that process where a number is given to each sentence according to the matched keyword with the question keyword. After removing stop words and performing stemming we will get the sentence keywords then we matched each keyword of the sentence to the question keyword and count the matching which will be the rank of the sentence. Sentence ranking flowchart is shown in figure 4.5 and algorithm is shown in algorithm 3.

Take a Sentence

Remove Stop Words

Perform Stemming

Extract Keywords

Count Matching
Sentence Keywords
with Question
Keywords

Give Count Score as the
Rank of the Sentence
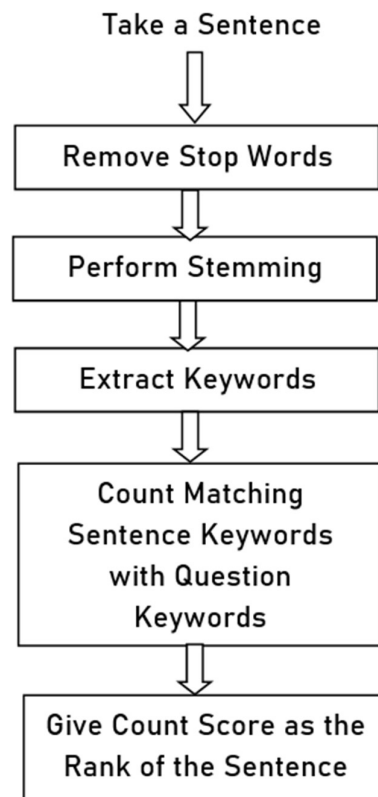
Fig. 4.5: Sentence Ranking

Algorithm 3: Sentence Ranking

Input : questionKeyWordList, document
Output: sentenceWithRankList

1   sentenceWithRankList ← List
2   **foreach** sentence in docuemnt **do**
3   sentenceRank ← 0.0 remove stopWord from sentence
4   **foreach** sentenceWord in sentence **do**
5   sentenceWord ← stem(sentenceWord)
6   **if** sentenceWord is equal to questionKeyWord **then**
7   sentenceRank + +
8   **end**
9   **end**
10   add sentence to sentenceWithRankList with rank
11 **end**

4.6.3   Extracting answer

To identify correct answer here we select the high ranked sentence, then process the

sentence in two ways:

■   **Find numeric question answer**:

For finding numeric question answer we have matched only numeric value not descriptive

numeric value. If the question type is matched with numeric type then this process will

work. We extract numeric value from the sentence.

■   **Find other question answer:**

For finding other question answer, we first find n-gram, find longest common

subsequence, remove the question keywords from the sentence and finally find the

answer. Answer extraction algorithm is shown in algorithm 4.

Algorithm 4: Answer Extraction

Input : questionKeyWordList, highestRankedSentence, questionType, question
Output: answer

1 **if** *questionType is equal numeric* **then**
2   answer = extractNumberValuefromsentence(highestRankedSentence);
3   answer = answer + numberUnit;
4 **end**
5 **else**
6   words = words of highestRankedSentence;
7   ngrams = Ngrams.ngrams(words, words.size() / 2);
8   answer = the ngram having the longestCommonSubsequence with question;
9 **end**

4.6.4   Generating distractor

To generate distractor here we select top four high ranked sentences and then process the sentence in two ways:

■ **Find numeric type distractors:**

For finding numeric question answer, we have matched only numeric value not descriptive numeric value. If the question type is matched with numeric type then this process will work. We extract numeric value from the sentence.

■ **Find other type distractors:**

For finding other question distractor we remove the question keywords from the sentence and then find the distractor. Then we randomly assign the distractor into a serial as MCQ question answer system accepts four options normally. Distractor generation is shown in algorithm 5.

Algorithm 5: Distractor Generation

Input : questionKeyWordList, fourHighestRankedSentence, questionType, question
Output: distractors

1 **if** questionType is equal numeric **then**
2   **foreach** sentence in fourHighestRankedSentence **do**
3   distractor = extractNumberValuefromsentence(sentence);
4   distractor = answer + numberUnit;
5   **end**
6 **end**
7 **else**
8   **foreach** sentence in fourHighestRankedSentence **do**
9   words = words of sentence;
10   ngrams = Ngrams.ngrams(words, words.size() / 2);
11   distractor = the ngram having the longestCommonSubsequence with question;
12   **end**
13 **end**

Here Table 4.2 shows some samples of the answer and option generation of this system

Topic Title	Question	Type	Option 1	Option 2	Option 3	Option 4	Answer
মিলিটারি ইন্সটিটিউট অফ সায়েন্স এ্যান্ড টেকনোলজি	কত সালে এটি বাংলাদেশ ইউনিভার্সিটি অব প্রফেশনালস-এর অন্তর্ভুক্ত হয়?	Numeric	১৯৯৮ সালে	২০০০ সালে	২০০৮ সালে	২০০২ সালে	২০০৮ সালে
বাংলাদেশ সরকার	প্রধানমন্ত্রী হতে হলে ন্যূনতম কত বছর হতে হবে?	Numeric	৩৫ বছর	৫৬ বছর	২৫ বছর	৫ বছর	২৫ বছর
ইবনে সিনা	মেওয়া বিক্রেতা কিসে বিশেষ পারদর্শী ছিল?	Noun	ভারতীয় গণিতশাস্ত্রে	লেখকদের বইয়ের অমূল্য ভান্ডার	সকল বিষয়ে আমি	জীবনের শেষ দিন পর্যন্ত এটি	ভারতীয় গণিতশাস্ত্রে
হুমায়ূন আহমেদ	বগুড়া জিলা স্কুলে কোন শ্রেণিতে ভর্তি হন?	Noun	বাংলাদেশের মুক্তিযুদ্ধ চলাকালে তৎকালীন পিরোজপুর	বাবা চাকুরী সূত্রে দেশের বিভিন্ন	নবম	ঢাকা কলেজে	নবম

Table 4.2: Automatically Generated Option & Answer

4.7 Develop the System

Develop the system consists of the steps of a) E-R diagram design, b) technological tools selection, c) feature list generation, and d) coding.

4.7.1 Design E-R diagram

In this step we have to design the system architecture and the database design for that we have to make E-R diagram and our system E-R diagram is given below:
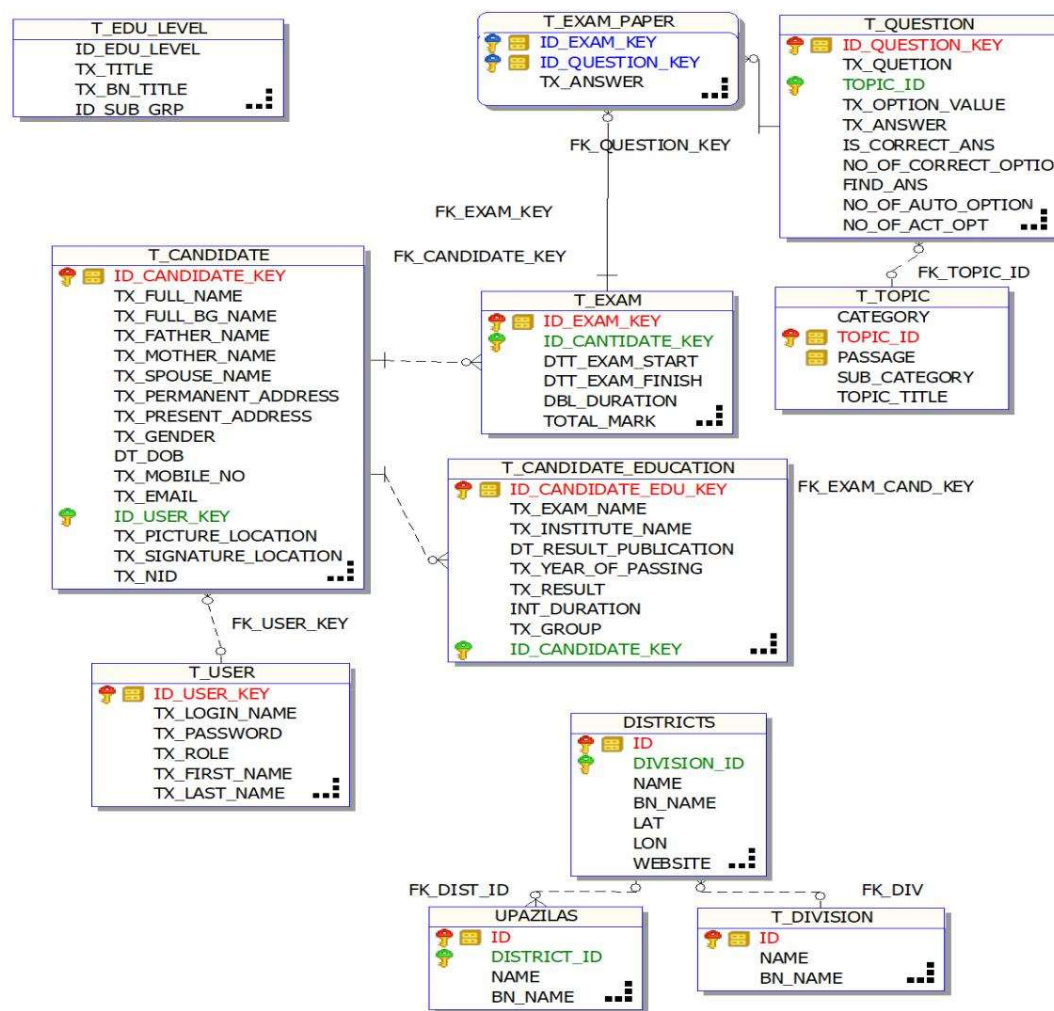
Fig. 4.6: E-R Diagram

4.7.2   Selecting technological tool

For selecting technological tool, we select the latest popular technology for software development in this time. The technology architecture is given below:

✠ **Back End:** Back end side in software development is the data storage part. For data storage purpose, we have selected oracle as database. We have used oracle 11g version.

✠ **Middle Tier:** Spring Framework, a Java based framework is used for middle tier development. We have used 5.0.3 release version of spring framework.

✠ **GUI:** For graphical user interface we have used ExtJs framework the popular product of Sencha and also known as Extensive JavaScript. We have used 6.2.0.981 version of ExtJs.

In many disciplines, teachers use MCQs as their primary evaluation method and it is estimated that 45-67% of student evaluations use MCQs. The rapid development of e-learning technologies has, in turn, stimulated the method of automatic generation of MCQs and has become an active theme in application-oriented NLP research today. MCQs can be used to evaluate a wide range of knowledge and yet take less time for administration. Therefore, they can be used to provide timely input to test takers. On the other hand, they are difficult to build and require considerable time to formulate each question. The manual construction of MCQs does not necessarily imply that they are well constructed in addition to the considerable preparation time. For the manual of the system see Appendix A.

CHAPTER 5

EXPERIMENT AND RESULT

This system consists of two main parts, one is answer extraction and another one is options generation. The evaluation of answer extraction and options generation is done separately.

5.1   Correctness of System

As this is a project, so we give a small number of questions just to evaluate the system. Answer extraction correctness is the value of correctly found answer among total answer which is 80% in our system. The equation is given below:

$$correctness\ of\ answer\ extraction = \frac{correctly\ extracted\ answer}{total\ extracted\ answer} \qquad (5.1)$$

Option generation correctness is the value of correctly found option among total found options which is 53% in our system. The equation is given below:

$$correctness\ of\ option\ generation = \frac{correctly\ extracted\ options}{total\ extracted\ options} \qquad (5.2)$$

The graphical representation of correctness of our system is given below:
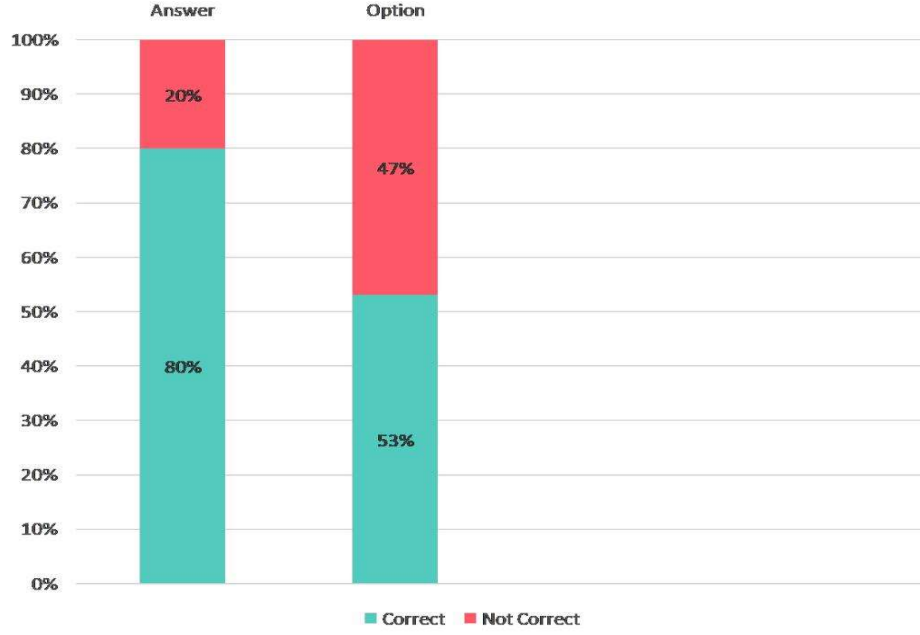
Fig. 5.1: Correctness of System

5.2   Evaluation Metric

In general, to evaluate the performance of an NLP system the evaluation metric is used. In this system we used three evaluation metric which is given below:

$$Precision = \frac{Relevant\ Items\ Retrieved}{Retrieved\ Items} \tag{5.3}$$

$$Recall = \frac{Relevant\ Items\ Retrieved}{Relevant\ Items} \tag{5.4}$$

$$FScore = \frac{2*(Precision*Recall)}{Precision + Recall} \tag{5.5}$$

The table 5.1 shows the result of the evaluation metric.

	Precision	Recall	FScore
Option Generation	0.63	0.57	0.6

Table 5.1: Option Generation Accuracy

5.3 Sample Screen

A sample screen for answer extraction and option generation is given below:

Fig. 5.2: Answer and option generation

CHAPTER 6

CONCLUSION

6.1   Main Outcome

Here we tried to build a Bangla question answering system and option generator to reduce the time of the examiner who set the question paper of MCQ examination. Finally, we are able to build a system where numeric question answer and options are generated automatically though there are some limitations.

Till this time, it is hard to find any established language tool for Bangla language. The structure of the Bangla language is very complex. In our system we can extract more correct answer for other type question if we can use *POS* tagger and named entity recognizer.

6.2   Practical Implication

Though this system has some limitations but overcoming these it can be used in various field of examination system. A huge number of MCQ based examination is done during the university admission test. MCQ based examination system is also done during employee requirement. In education MCQ based question is used in most of the stages. So, there are lots of area where this system can be used.

6.3   Challenges

Working with Bangla language in the field of NLP we faced a lot of challenges. Bengali is a low asset language, i.e. the scope of regular language handling instruments and phonetic assets are still little. For instance, the English language Wikipedia includes right around

four million articles while Bengali Wikipedia has minimal more than 20,000 articles. Research on language particular parts of Bengali data recovery is still in its earliest stages. For question answering system need to set a possible class labels for a given questions. It is hard to make a manual classifier with a constrained measure of standards. Next one is features extraction of the questions, the literary assets are more bound and the target content is less inclined to precisely coordinate content in the questions.

6.3.1   Lack of Established QA System

There are very few QA systems are found for Bangla language. Though many works are in-progress so it becomes hard to start the work.

6.3.2   Complex Language Structure

Bangla is a language having complex structure. There is no fixed structure so predicting the question is very hard. For example:

কবে বাংলাদেশ স্বাধীন হয় ?

বাংলাদেশ স্বাধীন হয় কবে ?

বাংলাদেশ কবে স্বাধীন হয় ?

Here no fixed position for the starting of question word is available.

6.3.3   Difficult Question Pattern

Even with classifier, certain questions are difficult for humans to identify them correctly. Turning to the question:

বাংলার শেষ স্বাধীন নবাব সিরাজউদ্দৌলার সাথে ব্রিটিশ ইস্ট ইন্ডিয়া কোম্পানির যে স্থানে যুদ্ধ সংঘটিত হয়েছিল তার নাম পলাশী ? this a complex type question.

6.3.4 Falling behind from State of the Art

A lot of established NLP tools are already developed for English language where we are behind from established Bangla NLP tools, though some tools are available over the internet like stemmer, POS tagger but most of them are rule based. We are lagging behind the established question classification model, name entity recognition model which makes tough for working with Bangla language.

6.3.5 Lack of Bangla NLP tools

The first need for the development of computer linguistics is a well labelled corpus. There is still no well tagged corpus in Bangla to date. Tools such as named entity recognizer, parts of speech tagger, stemmer, parser etc. also have a huge impact on the developed system. For Bengali language there is still no standard NER, POS tagger, Stemmer which is very important for answer processing phase. So, for Bengali language these NLP tools need to be developed.

6.3.6 Lack of common platform

There is no common platform for Bangla language which can accept the data uniformly. The data, the corpus collection in large sense along with subsidiary tagged or undefined words or sentence sharing platform has not yet been built.

6.4   Future Work

We worked for an automated options and answer extraction system for Bangla language. There are some limitations as we can extract numeric answer and options but for other type question the answer and options are not appropriate for most of the time. In future we can work for this. Furthermore, we used rule based technology and we can use ML (Machine Learning), DL (Deep Learning), NN (Neural Network) or hybrid process etc. in future and can evaluate each system and find the best one.

In future we can use *POS* tagger, named entity recognizer etc. for making the result accurate and also can implement AI here for making more smart software.

REFERENCES

[1] D. Coniam, "A preliminary inquiry into using corpus word frequency data in the automatic generation of english language cloze tests," *CALICO Journal*, vol. 14, 01 1997.

[2] R. Mitkov and L. A. Ha, "Computer-aided generation of multiple-choice tests," in *Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing*, 2003, pp. 17–22.

[3] R. Mitkov, L. An HA, and N. Karamanis, "A computer-aided environment for generating multiple-choice test items," *Natural Language Engineering*, vol. 12, no. 2, pp. 177–194, 2006.

[4] J. C. Brown, G. A. Frishkoff, and M. Eskenazi, "Automatic question generation for vocabulary assessment," in *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, ser. HLT 05. USA: Association for Computational Linguistics, 2005, pp. 819–826.

[5] I. Aldabe, M. L. de Lacalle, M. Maritxalar, E. Martinez, and L. Uria, "Arikiturri: An automatic question generator based on corpora and nlp techniques," in *Intelligent Tutoring Systems*, M. Ikeda, K. D. Ashley, and T.-W. Chan, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 584–594.

[6] I. Aldabe and M. Maritxalar, "Automatic distractor generation for domain specific texts," in *Proceedings of the 7th International Conference on Advances in Natural Language Processing*, ser. IceTAL10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 27–38.

[7] A. Papasalouros, K. Kanaris, and K. Kotis, "Automatic generation of multiple choice questions from domain ontologies," in *e-Learning*, 01 2008, pp. 427–434.

[8] M. Agarwal and P. Mannem, "Automatic gap-fill question generation from text books," in *Proceedings of the Sixth Workshop on Innovative Use of NLP for Building Educational Applications*. Portland, Oregon: Association for Computational Linguistics, Jun. 2011, pp. 56–64.

[9] M. Agarwal, R. Shah, and P. Mannem, "Automatic question generation using discourse cues," in *Proceedings of the Sixth Workshop on Innovative Use of NLP for Building Educational Applications*. Portland, Oregon: Association for Computational Linguistics, Jun. 2011, pp. 1–9.

[10] L. Plamondon and G. Foster, "Quantum, a french/english cross-language question answering system," in *Comparative Evaluation of Multilingual Information Access Systems*, C. Peters, J. Gonzalo, M. Braschler, and M. Kluck, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 549–558.

[11] L. A. Ha and V. Yaneva, "Automatic question answering for medical MCQs: Can it go further than information retrieval?" in *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*. Varna, Bulgaria: INCOMA Ltd., Sep. 2019, pp. 418–422.

[12] L. A. Ha and V. Yaneva, "Automatic distractor suggestion for multiple-choice tests using concept embeddings and information retrieval," in *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 389–398. [Online]. Available: https://www.aclweb.org/anthology/W18-0548.

[13] G. S. Mann, "A statistical method for short answer extraction," in *Proceedings of the Workshop on Open-Domain Question Answering - Volume 12*, ser. ODQA 01. USA: Association for Computational Linguistics, 2001, pp. 1–8.

[14] S. Banerjee and S. Bandyopadhyay, "Bengali question classification: Towards developing QA system," in Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing, WSSANLP@COLING 2012, Mumbai, India, December 8, 2012, V. Sornlertlamvanich and A. Malik, Eds. The COLING 2012 Organizing Committee, 2012, pp. 25–40.

[15] S. Banerjee and S. Bandyopadhyay, "Ensemble approach for fine-grained question classification in bengali," 01 2013.

[16] S. Sharma, M. K. Hossen, M. Islam, M. Chowdhury, S. Azad, M. Hoque *et al.*, "Automatic question and answer generation from bengali and english texts." *Computer Science & Telecommunications*, vol. 54, no. 2, 2018.

[17] S. Banerjee, S. K. Naskar, and S. Bandyopadhyay, "Bfqa: A bengali factoid question answering system," in *Text, Speech and Dialogue*, P. Sojka, A. Horák, I. Kopecek,ˇ and K. Pala, Eds. Cham: Springer International Publishing, 2014, pp. 217–224.

[18] F. Kabir, S. Siddique, M. R. A. Kotwal, and M. N. Huda, "Bangla text document categorization using stochastic gradient descent (sgd) classifier," in *2015 International Conference on Cognitive Computing and Information Processing(CCIP)*, 2015, pp. 1–4.

[19] M. A. Islam, M. F. Kabir, K. Abdullah-Al-Mamun, and M. N. Huda, "Word/phrase based answer type classification for bengali question answering system," in *2016 5th International Conference on Informatics, Electronics and Vision (ICIEV)*, 2016, pp. 445–448.

[20] S. Khan, K. T. Kubra, and M. M. H. Nahid, "Improving answer extraction for bangali q/a system using anaphora-cataphora resolution," in *2018 International Conference on Innovation in Engineering and Technology (ICIET)*, 2018, pp. 1–6.

[21] M. Kowsher, F. S. Tithi, M. Ashraful Alam, M. N. Huda, M. Md Moheuddin, and M. G. Rosul, "Doly: Bengali chatbot for bengali education," in *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, 2019, pp. 1–6.

[22] M. Kowsher, M. A. Alam, M. J. Uddin, M. R. Islam, N. Pias, and A. R. M. Saifullah, "Bengali informative chatbot," in *2019 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2)*, 2019, pp. 1–7.

[23] S. Sarker, S. T. A. Monisha, and M. M. H. Nahid, "Bengali question answering system for factoid questions: A statistical approach," in *International Conference on Bangla Speech and Language Processing (ICBSLP)*, vol. 27, 2019, p. 28.

[24] E. M. N. Alkholy, M. H. Haggag, and A. Aboutabl, "Question answering systems: Analysis and survey," *International Journal of Computer Science and Engineering Survey*, vol. 9, no. 6, pp. 1–13, 2018.

[25] B. Green, A. Wolf, C. Chomsky, and K. Laughery, *BASEBALL: An Automatic Question Answerer*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1986, pp. 545–549.

[26] J. L. Cook, "College students and algebra story problems: Strategies for identifying relevant information," *Reading Psychology*, vol. 27, no. 2-3, pp. 95–125, 2006.

[27] J. Weizenbaum, "Elizaa computer program for the study of natural language communication between man and machine," *Commun. ACM*, vol. 9, no. 1, pp. 36–45, Jan. 1966.

[28] V. Singh, "Different facets of text based automated question answering system," *International Journal for Research in Applied Science and Engineering Technology*, vol. 6, pp. 103–111, 2018.

[29] J. Weizenbaum, "Eliza - a computer program for the study of natural language communication between man and machine," *Commun. ACM*, vol. 9, pp. 36–45, 1966.

[30] W. A. Woods, "Progress in natural language understanding: An application to lunar geology," in *Proceedings of the June 4-8, 1973, National Computer Conference and Exposition*, ser. AFIPS 73. New York, NY, USA: Association for Computing Machinery, 1973, pp. 441–450.

[31] D. G. Bobrow, R. M. Kaplan, M. Kay, D. A. Norman, H. S. Thompson, and T. Winograd, "Gus, a frame-driven dialog system," *Artificial Intelligence*, vol. 8, no. 2, pp. 155–173, 1977.

[32] W. G. Lehnert, *The process of question answering: a computer simulation of cognition / Wendy G. Lehnert*. New York, NY, USA: L. Erlbaum Associates, 1978.

[33] J. Kupiec, "Murax: A robust linguistic approach for question answering using an on-line encyclopedia," in *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR 93. New York, NY, USA: Association for Computing Machinery, 1993, pp. 181–190.

[34] A. Huettner, "Question answering," 2000, 5th Search Engine Meeting.

[35] E. M. Voorhees, "The trec-8 question answering track report," in *In Proceedings of TREC-8*, 1999, pp. 77–82.

[36] K. Kwok, L. Grunfeld, N. Dinstl, and M. Chan, "Trec-9 cross language, web and question-answering track experiments using pircs," in *TREC*, 2000.

[37] E. M. Voorhees and D. Harman, "Overview of the ninth text retrieval conference (trec-9)," in *In Proceedings of the Ninth Text REtrieval Conference (TREC-9*, 2000, pp. 1–14.

[38] Z. Zheng, "Answerbus question answering system," in *Proceedings of the Second International Conference on Human Language Technology Research*, ser. HLT 02. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002, pp. 399–404.

APPENDIX A

MANUAL

Registration

Every applicant should register on the software otherwise s/he cannot get the access of

the software and the screens for registration looks like below:
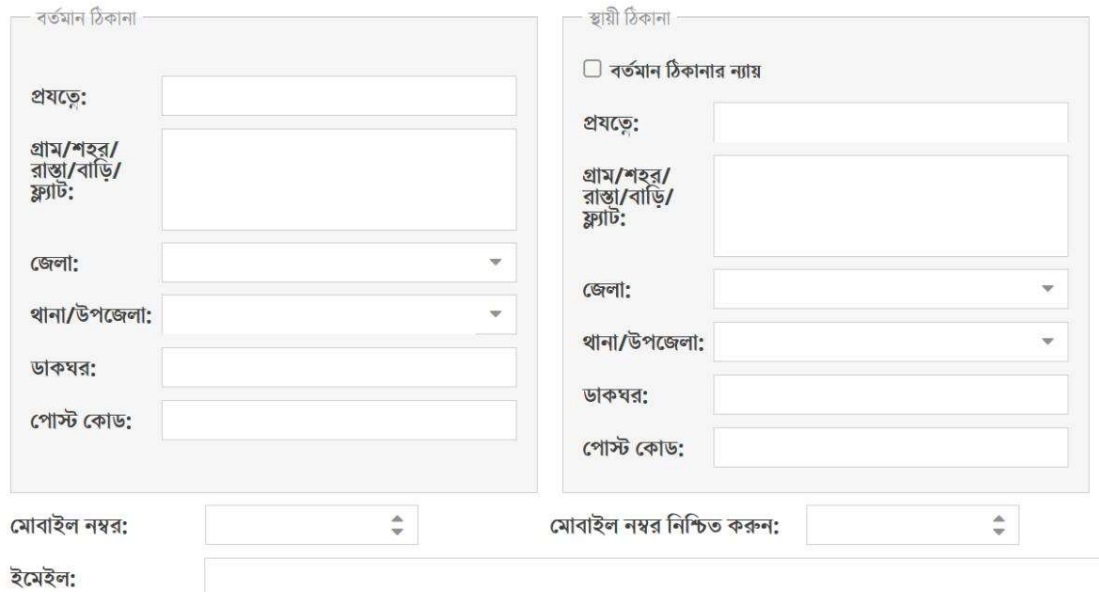
Fig. A.1: Registration Screen First Part

Here every input should be in English but the name in Bangla will take only Bangla name.
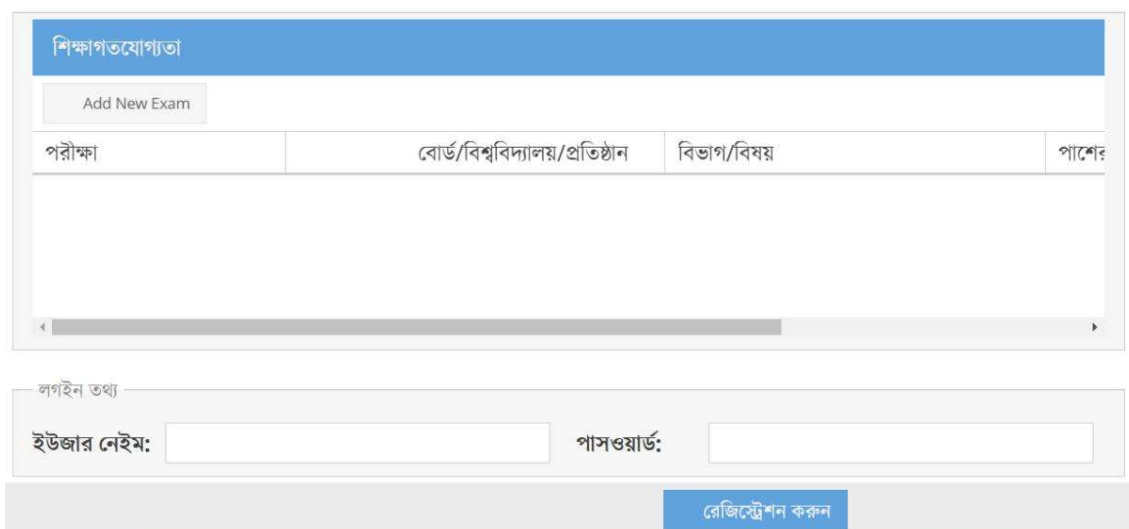
Present and permanent address should be given on the second part of the registration page.

Fig. A.2: Registration Screen Second Part

Educational qualification will be done by adding new exam where examination type wise education will be added against the applicant. User define his/her username and password for logged into the software.

Fig. A.3: Registration Screen Third Part

Login: There will be two types of user:

> **a. Admin User**, who can set up exam, question paper, edit user etc.

> **b. Applicant User**, who can participate in the examination.

Login screen is like just below:

Fig. A.4: Login Screen

Dashboard

Applicant dashboard contains total examination participated by the applicant and the screen looks like below:

Fig. A.5: Applicant Dashboard

Question paper: Applicant can participate exam after opening any topic. Questions will be appeared based on that topic. During examination time the browser window will be locked. There will be a "পরবর্তী প্রশ্ন" button from getting next question. When the question bank will be finished result will be shown immediately.

Fig. A.6: Question Paper

Test paper details: Applicant can see the full history of a test by clicking the exam list shown on dashboard. Test paper looks like below:
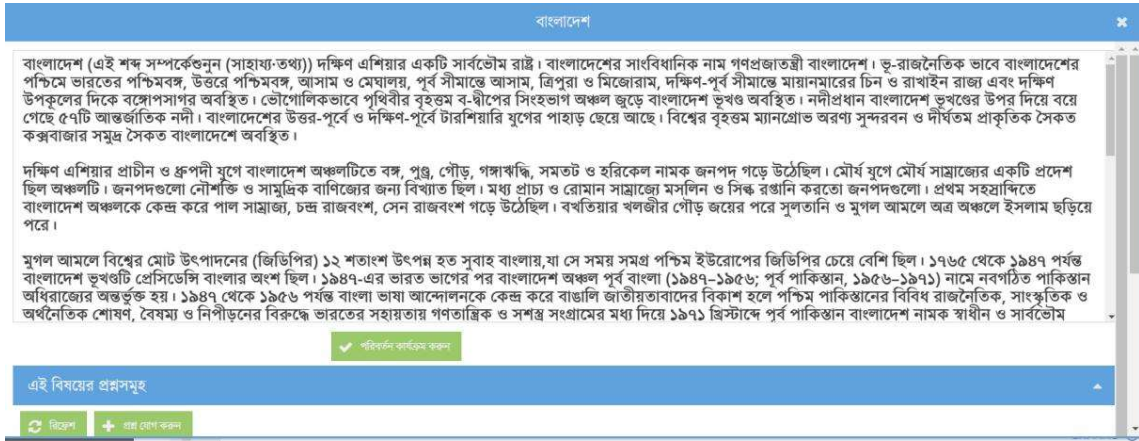

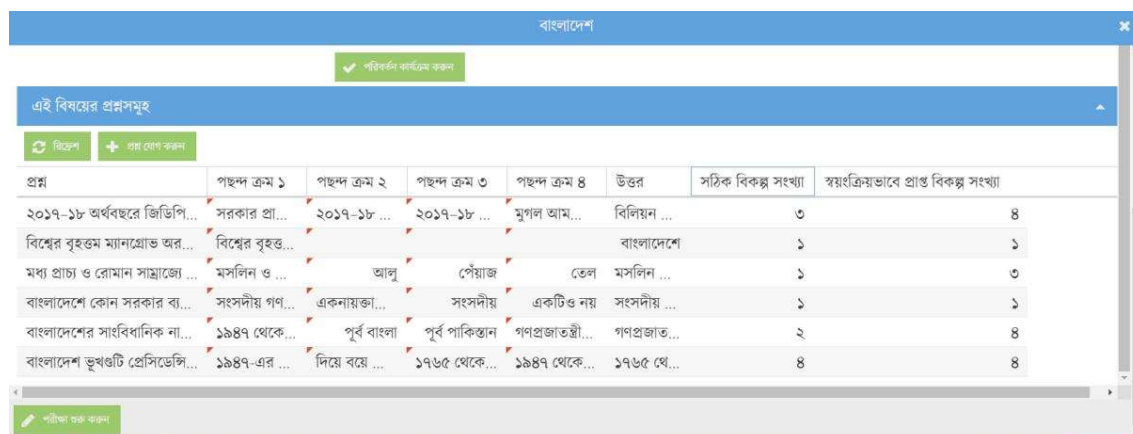**Fig. A.7: Test Paper**

Topic: This is the vital module of this software. This is the module which saves the maximum time of the question setter. Topic grid is the screen where existing topic will be shown with title and shot description and the screen looks like below:
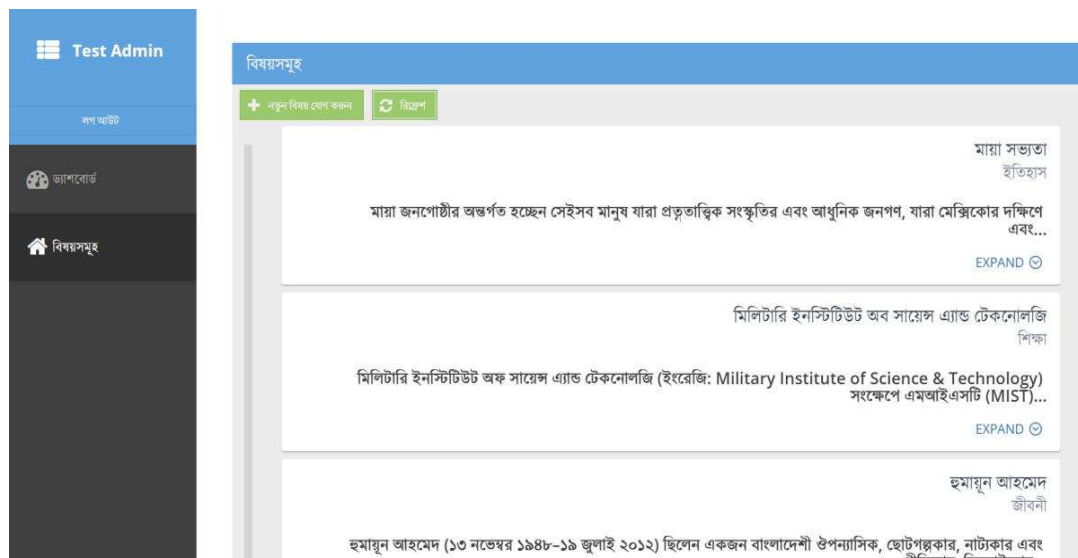
Fig. A.8: Topic Grid

Add topic: New topic can be added from the "নতুন বিষয় যোগ করুন" button and adding new topic screen looks like below:
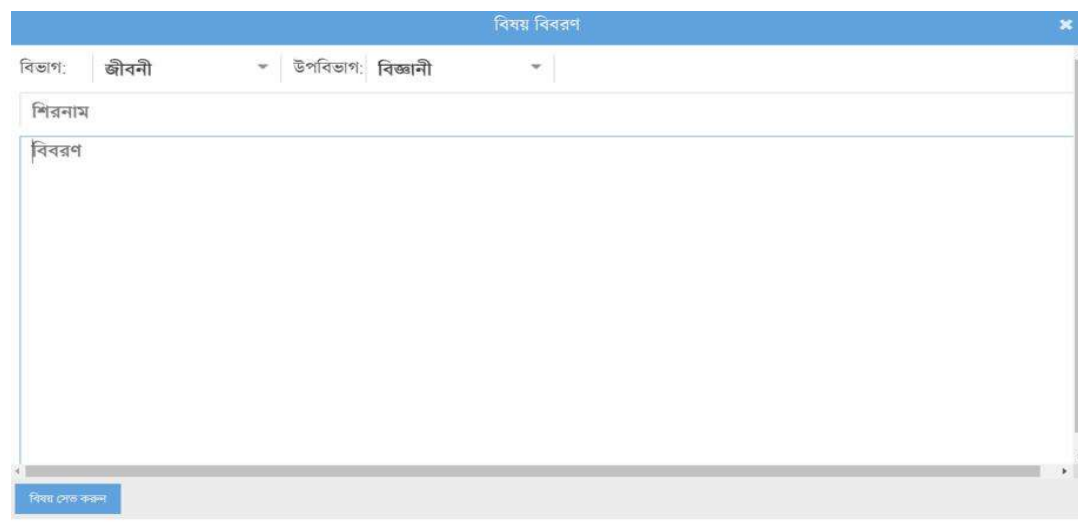
Fig. A.9: Add Topic

In this page there will be some predefined sections, under each section there will be some subsection. After selecting section and subsection user have to give input the title of the topic then details of the topic then after clicking "বিষয় সেভ করুন" the topic will be saved in database.

Update topic details: To update existing topic details user just need to select the topic from the topic grid then change the details and click "পরিবর্তন কার্যক্রম করুন" button and updating topic details screen looks like below:

Fig. A.10: Edit Topic Details

Question: This is the vital module of this software. This is the module which saves the maximum time of the question setter. Question grid is the screen where existing questions of a topic will be shown below the topic details and the screen looks like below:

Fig. A.11: Question Grid

Add question: New question can be added after clicking "প্রশ্ন যোগ করুন" button. After clicking the button a new window will open when user have to give the question as input and then clicking "সম্ভাব্য বিকল্প তৈরী করুন" button option against that question generated by the system will be appeared in the options field then user have to decide the correctness and edit options if applicable. Then user has to give a survey about the system generated option in the field where applicable. By clicking "প্রশ্ন সেভ করুন" the question will be saved in the database with the options. The screen looks like below:

Fig. A.12: Add Question Screen

That's all for the manual of this software. This software is made for ease the examination process of MCQ based exam. It is made user friendly so user can easily understand what she/he is going to do. User role based system makes clear to the users what accesses they have or not.