AI-BASED SMART ASSISTANT USING NATURAL LANGUAGE PROCESSING AND DEEP LEARNING

MD. REZWAN -A- ROWNOK

SAMIHA RAISA ZAMAN

SHAHRIAR RAHMAN KHAN

SHARMILA RAHMAN PRITHULA

A THESIS SUBMITTED
FOR THE DEGREE OF BACHELOR OF SCIENCE IN
COMPUTER SCIENCE AND ENGINEERING

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

2020

SUPERVISOR'S APPROVAL

This thesis paper titled **"AI-BASED SMART ASSISTANT USING NATURAL LANGUAGE PROCESSING AND DEEP LEARNING"**, submitted by Md. Rezwan -A- Rownok, ID: 201714035, Samiha Raisa Zaman, ID: 201714052, Shahriar Rahman Khan, ID: 201714055, and Sharmila Rahman Prithula, ID: 201714057 has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in 2020.

Col Siddharth Malik, SM
Senior Instructor
Department of Computer Science and Engineering
Military Institute of Science and Technology

DECLARATION

This is to certify that the work presented in this thesis paper, titled, “AI-BASED SMART ASSISTANT USING NATURAL LANGUAGE PROCESSING AND DEEP LEARNING”, is the outcome of the investigation and research carried out by the following students under the supervision of Col Siddharth Malik, SM, Senior Instructor, Department of Computer Science and Engineering, Military Institute of Science and Technology.

It is also declared that neither this thesis paper nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Md. Rezwan -A- Rownok
Roll: 201714035
14 March 2021

Samiha Raisa Zaman
Roll: 201714052
14 March 2021

Shahriar Rahman Khan
Roll: 201714055
14 March 2021

Sharmila Rahman Prithula
Roll: 201714057
14 March 2021

ABSTRACT

AI-based Chatbot is a computer program or an Artificial Intelligence software that can simulate a real human conversation with real time responses to users based on reinforced learning. AI Chatbots either use text messages, voice commands, or both. AI robots use natural language processing to communicate with Artificial Intelligence features embedded in them. The main aim of creating AI chatbots is to help customers make better informed decisions. Intelligent planning with the help of heuristics and ANN (Artificial Neural Network) aids an AI chatbot that learns to progress from one user request to another to come up with an intelligent response before completing the task. Planning is a process that relates to a sequence of actions performed by the AI chatbot leading to a structured conversation that involves acknowledgment, questions, and information relay. Internally the AI Bot has to perform intelligence building to form structured responses.

ACKNOWLEDGEMENT

We are thankful to Almighty Allah for his blessings for the successful completion of our thesis. Our heartiest gratitude, profound indebtedness and deep respect go to our supervisor, Col Siddharth Malik, SM, Senior Instructor, Department of Computer Science and Engineering, Military Institute of Science and Technology, for his constant supervision, affectionate guidance and great encouragement and motivation. His keen interest on the topic and valuable advices throughout the study was of great help in completing thesis. We are especially grateful to the Department of Computer Science and Engineering (CSE) of Military Institute of Science and Technology (MIST) for providing their all out support during the thesis work.

Finally, we would like to thank our families and our course mates for their appreciable assistance, patience and suggestions during the course of our thesis.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

ANN	:	Artificial Neural Network
RNN	:	Recurrent Neural Network
LSTM	:	Long Short-Term Memory
NER	:	Named Entity Recognition
NLU	:	Natural Language Understanding
ASR	:	Automatic Speech Recognition
ER	:	Entity Recognition
HCI	:	Human-Computer Interaction
CMC	:	Computer-Mediated Communication

CHAPTER 1
INTRODUCTION

1.1 Introduction

Chatbots are computer systems that use artificial intelligence to simulate human interaction (AI). It is intended to be the ultimate personal assistant for entertainment purposes, assisting with activities such as answering questions, getting driving instructions, turning up the temperature in a smart home, and playing one's favorite music, among others. Chatbots are becoming increasingly common with businesses due to their ability to minimize customer service costs and accommodate many users at once. However, in order to complete certain tasks, chatbots must be as efficient as possible.

Chatbots and virtual assistants are the future of marketing and customer support. The emergence of chatbots in the financial industry is the most recent huge phenomenon that has altered how consumers communicate. The use of Artificial Intelligence-powered chatbots, especially in the banking industry, has changed the face of the bank-customer contact interface [1]. Before the use of chatbots, many companies used call centers to provide customer care services. But to maintain the employees and cost of call centers, it becomes really difficult for most of the companies. That's why there are some tech companies that don't have any call centers rather they use chatbots to serve that purpose. For example Uber, Google, Facebook, Twitter, etc.

A growing number of businesses are using chatbots in lots of exciting ways - in fact, according to Facebook, there are now over 300,000 active bots on Messenger [2]. But, currently available chatbots are usually designed for unique situations and basic questions and commands. These programs are incapable of dealing with application domains with a wide range of related information and complex resources as most of these chatbots used by renowned companies are totally scripted [3].

But in Bangladesh, well-recognized companies like Daraz, Evaly, Robi Axiata, Pure-It, etc. provide customer care using human agents and a large chunk of the company's revenue goes to this customer-care sector. Also in most of the university websites, there is no proper chatbot that will provide all the frequently asked queries and necessary information to the students and other visitors [4]. Chatbots in university environments have had too little recognition, for example, for offering operational support for research, classes, and exams. In the scientific community, this area of expertise is also in its infancy [5].

Therefore, the objective of this research is to understand the fundamental requirements of developing a digital solution for providing e-customer support and to develop a web-based solution for enhancing the usability of chatbots in different renowned organizations in the context of Bangladesh.

1.2 Objectives

- An AI based digital solution of the old analog customer care systems.
- Saving time and cost for conducting customer care services.
- Providing services by 24/7
- Both textual and audio based queries. Works in both Bangla and English Language simultaneously.
- Can understand the queries asked in natural language.
- Easily customizable and integration for any organization.

1.3 Thesis outline

The contribution of this research is two-folded. At first, the functionality, design, and psychological components necessary to provide the most benefit to developing a chatbot are discovered; and then based on the requirements, a usable application is developed in the context of Bangladesh.

The rest of the thesis is organized as follows. Chapter 2 presents Literature Review. Findings of the requirement elicitation study are discussed in section 3. Next, in Section 4, the design and development of this system are illustrated briefly. The evaluation study with results is presented in Section 5. Finally, section 6 contains the conclusion and future expansion.

CHAPTER 2
LITERATURE REVIEW

A limited number of researches have been conducted focusing on the development of assistive technologies for supporting the AI-based Bangla assistant chatbot. This section briefly introduces these works.

2.1 Ways of Developing Chatbots

In recent years, chatbot technology has become more popular, and many conversational chatbots have been developed to replace conventional chatbots. A chatbot is a computer program that interacts with humans and satisfies their requirements. The chatbot responds to the user's questions, and they may even perform tasks on occasion [6].

A chatbot can be built in a variety of ways, but the Deep Learning bot needs Neural Networks to learn the input series. Among them, NLP, ANN, RNN, and LSTM are mostly used.

- **Natural Language Processing (NLP):** It is a subset of Artificial Intelligence (AI) that enables computer and human languages to communicate. NLP can be used not only for text translation but also for speech recognition. It produces a predetermined answer. Many NLPs in the 1980s were created using a series of hand-written rules. For language recognition, they have later supplemented with Machine Learning (ML) algorithms [6].

- **Recurrent Neural Network (RNN):** It's a variation on the general feed-forward network in which the network considers both the current input and the previous output while generating an answer. RNNs also have memories, which can be used to recall the sequence of inputs [6].

- **Long Short-Term Memory (LSTM):** RNN's biggest flaw is that they can't recall the input over a long series of events. This problem can be solved using LSTMs, which are a kind of RNN that can remember long data sequences [6].

2.2 Different Chatbots

Most large-scale conversational AI agents today, such as Alexa, Siri, or Google Assistant, rely on manually annotated data to train the system's various components, such as Automatic Speech Recognition (ASR), Natural Language Understanding (NLU), and Entity Resolution (ER). In the paper [7], a method was proposed that uses input signals from customer/system interactions to simplify learning without the need for manual annotation. Users of these programs also alter prior queries in the hopes of correcting a mistake in the previous turn and obtaining the desired results.

Recent research has shown that language models can be used in text-matching scenarios to build Bi-encoders that perform nearly as well as Cross-encoders while inferring at a much faster rate. In the paper [8], they have worked on this concept by creating a sequence matching architecture that uses the entire training set as a temporary knowledge base during inference.

Chat-Bot-Kit, a web-based tool for text-based chats that was created for computer-mediated communication research purposes (CMC). The platform is intended for use in wizard-of-oz experiments of Human-Computer Interaction (HCI) and chatbot (dialogue system) evaluations in Natural Language Processing (NLP) [9].

In the paper [10], they developed a chatbot that will assist survivors of sexual harassment cases. Machine learning classification, Named Entity Recognition (NER), and slot-filled-based chatbot were used to assist victims of such incidents by referring them to relevant institutions that will assist them, and improve incident reporting in order to collect additional details on abuse situations that are still underreported.

Different types of Chatbots can be used as conversational agents in various business domains, in order to improve customer experience and satisfaction. Any business area needs the creation of a knowledge base within that domain, as well as the development of an information retrieval system that can provide the user with documents or created sentences. Natural Language Understanding (NLU) is the central component of a Chatbot, and deep learning approaches have significantly enhanced NLU. The paper [11] focuses on Named Entity Recognition (NER) and Intent Classification models that can be built into a Chatbot's natural language understanding (NLU) operation.

In the paper [12], they discovered several flaws in currently existing healthcare chatbots and proposed a healthcare assistant chatbot that allows users to screen for signs of common ailments, get a referral to see a doctor if necessary, get fitness recommendations, monitor activity or workout routines, and access a concise exercise guide. Their proposed system is self-adapting based on the user's previous encounters with the bot and focuses on the consistency of human-computer interaction.

A vast percentage of users are unable to obtain technical career advice because it is costly; users waste hours scouring websites for specific necessary details; and third-party website information is not genuine, credible, or maintained on a consistent basis. To address these problems, a chatbot was proposed in paper [13]. It uses Feed forward neural network is used to train the chatbot.

There may be a lot of works done on different kinds of chatbots, but fewer works have been done on the Bangla chatbot. “Disha (Direction)”, a machine learning-based closed domain Bangla healthcare chatbot was developed that can converse in Bangla with the user using its knowledge base and learning from user experiences. It aids in the diagnosis of possible diseases based on signs entered, the tracking of a user’s health status, and the alerting of a user to potential health hazards [14].

To sum up, a very limited number of studies were carried out on the digital solution for developing AI-based Bangla chatbots as a smart assistant, especially for the people in Bangladesh. Moreover, only a few research discussed the design, development, and evaluation processes of any web application that provides an AI-based Bangla-assisted chatbot. Thus, this research focused on this issue in the context of Bangladesh.

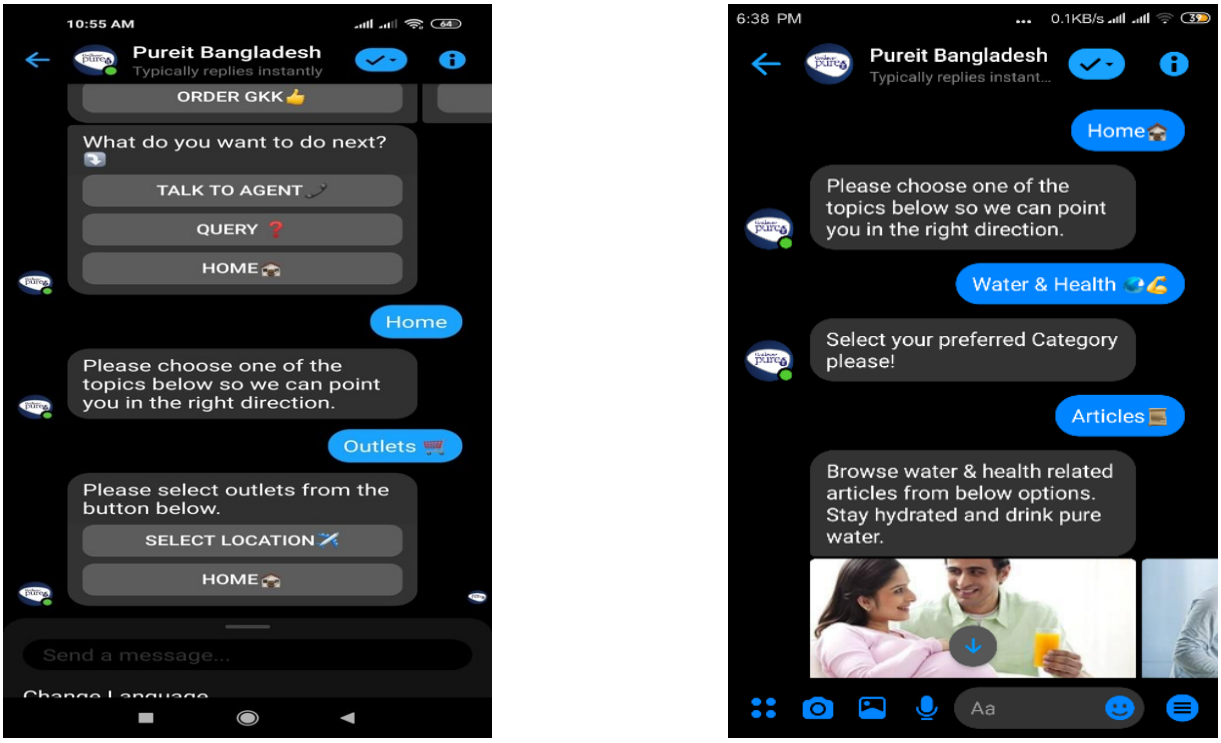
CHAPTER 3
REQUIREMENT ELICITATION

In this section, we have analyzed some chatbots of some renowned companies of Bangladesh. We analyzed the chatbots of Pureit, Robi, Grameenphone, BanglaLink etc.

3.1 Current Scenario

3.1.1 Pureit Chatbot of Bangladesh

Screenshots of Pureit chatbot is shown in figure 3.1. Here, messenger chatbot of pureit is really poor. All the questions are scripted. If a user asks something, that is out of the script, then, the chatbot doesn't understand the query. The user doesn't have any freedom of asking any question.



What do you want to do next?

Please choose one of the topics below so we can point you in the right direction.

Please select outlets from the button below.

Please choose one of the topics below so we can point you in the right direction.

Select your preferred Category please!

Browse water & health related articles from below options. Stay hydrated and drink pure water.

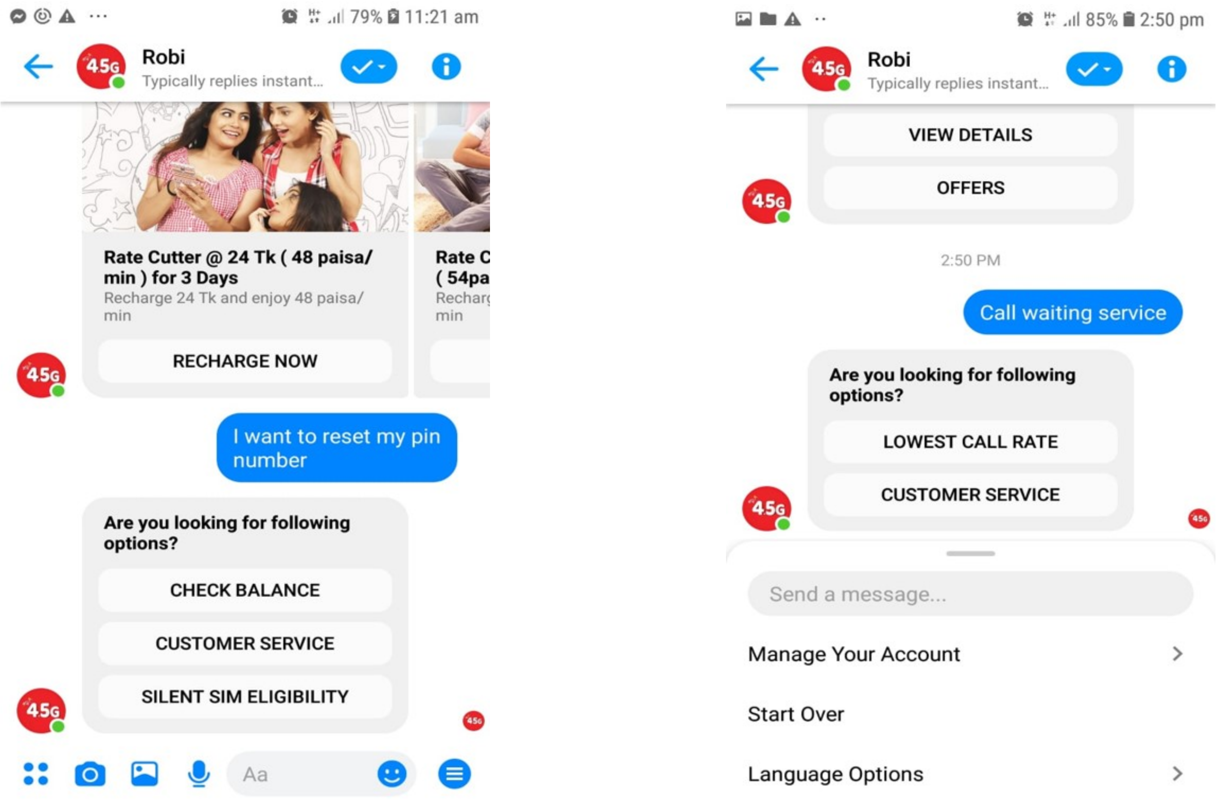
(a) screenshot 1

(b) screenshot 2

Figure 3.1: Screenshots of Pureit chatbot

3.1.2 Robi Chatbot of Bangladesh

Screenshots of Robi chatbot is shown in figure 3.2. Here, the messenger chatbot of robi is not good at answering all the user's queries. All the questions here are scripted as well. If a user asks something, that is out of the script, then, the chatbot doesn't understand the query. The user doesn't have any freedom of asking any question as well. At one point, the questions fall in a loop and there are no options left for the users to choose. The Robi chatbot doesn't understand and detect Natural Language.



Rate Cutter @ 24 Tk (48 paisa/ min) for 3 Days

Recharge 24 Tk and enjoy 48 paisa/ min

I want to reset my pin number

Are you looking for following options?

(a) screenshot 1

2:50 PM

Call waiting service

Are you looking for following options?

(b) screenshot 2

Figure 3.2: Screenshots of Robi chatbot

3.2 Findings from Existing Chatbots

After analyzing some renowned companie's chatbot, we have got some findings.

- The chatbots are totally scripted.
- There is no freedom to ask any type of question.
- These chatbots cannot understand the Natural language.
- That is why most of the big companies depends on the human based call-center.
- These chatbots doesn't have both audio and textual based query system.

CHAPTER 4
METHODOLOGY AND IMPLEMENTATION

In the world of machine learning and AI, there are many different kinds of chatbots. Some chatbots are virtual assistants; others are just there to talk to; some are customer support agents. Here, we will be creating a chatbot for MIST-website that will be used to answer frequently asked questions. This project has been done in 6 major steps.

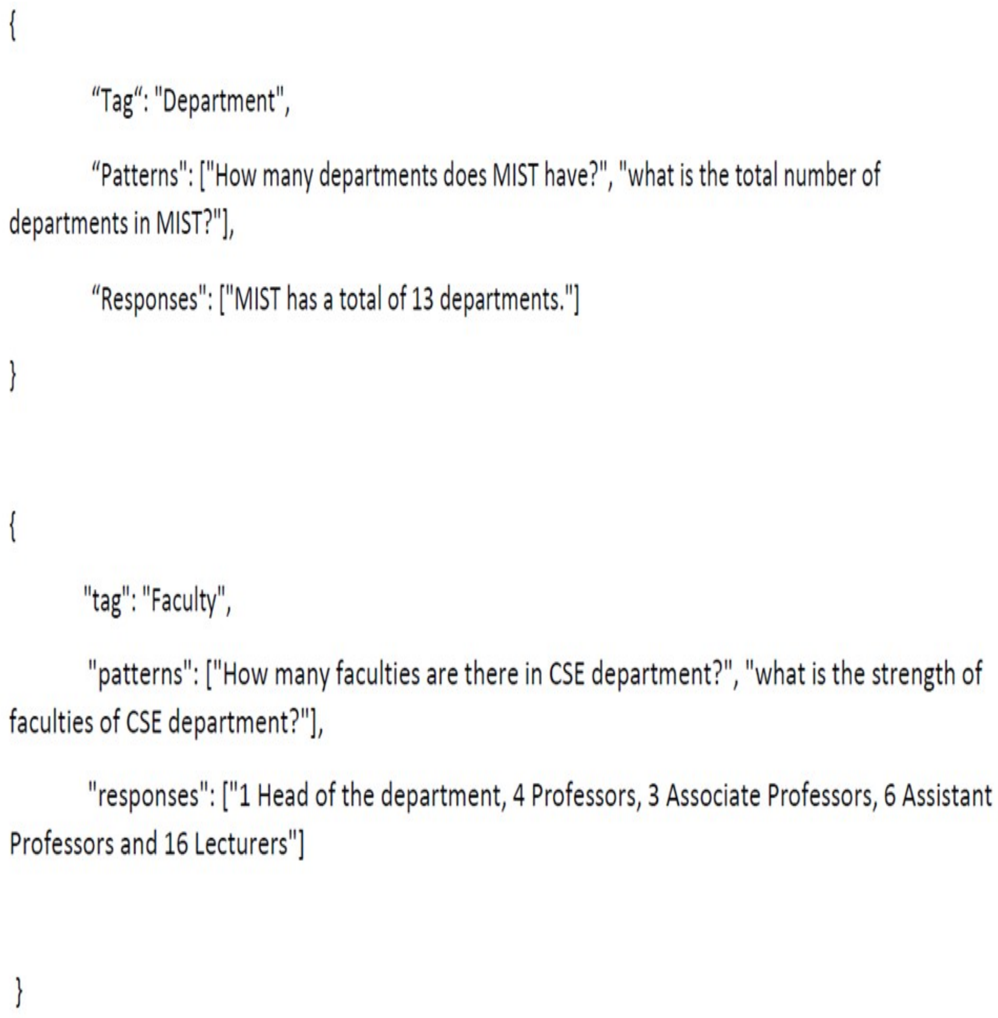
- Dataset creation for CSE department of MIST.
- Pre-processing the dataset for feeding it into the neural network.
- Creating the Neural Network Model.
- Predicting responses in real-time.
- Building a graphical user interface.
- Sending unsuccessful queries to Realtime database for further development.

4.1 Dataset Creation

The dataset is created in a JSON file. Each conversational intent contains:

- a tag (a unique name)
- patterns (sentence patterns for our neural network text classifier)
- responses (one will be used as a response)

A little portion of the dataset is shown in figure 4.1, 4.2, 4.3, 4.4, 4.5 and 4.6:

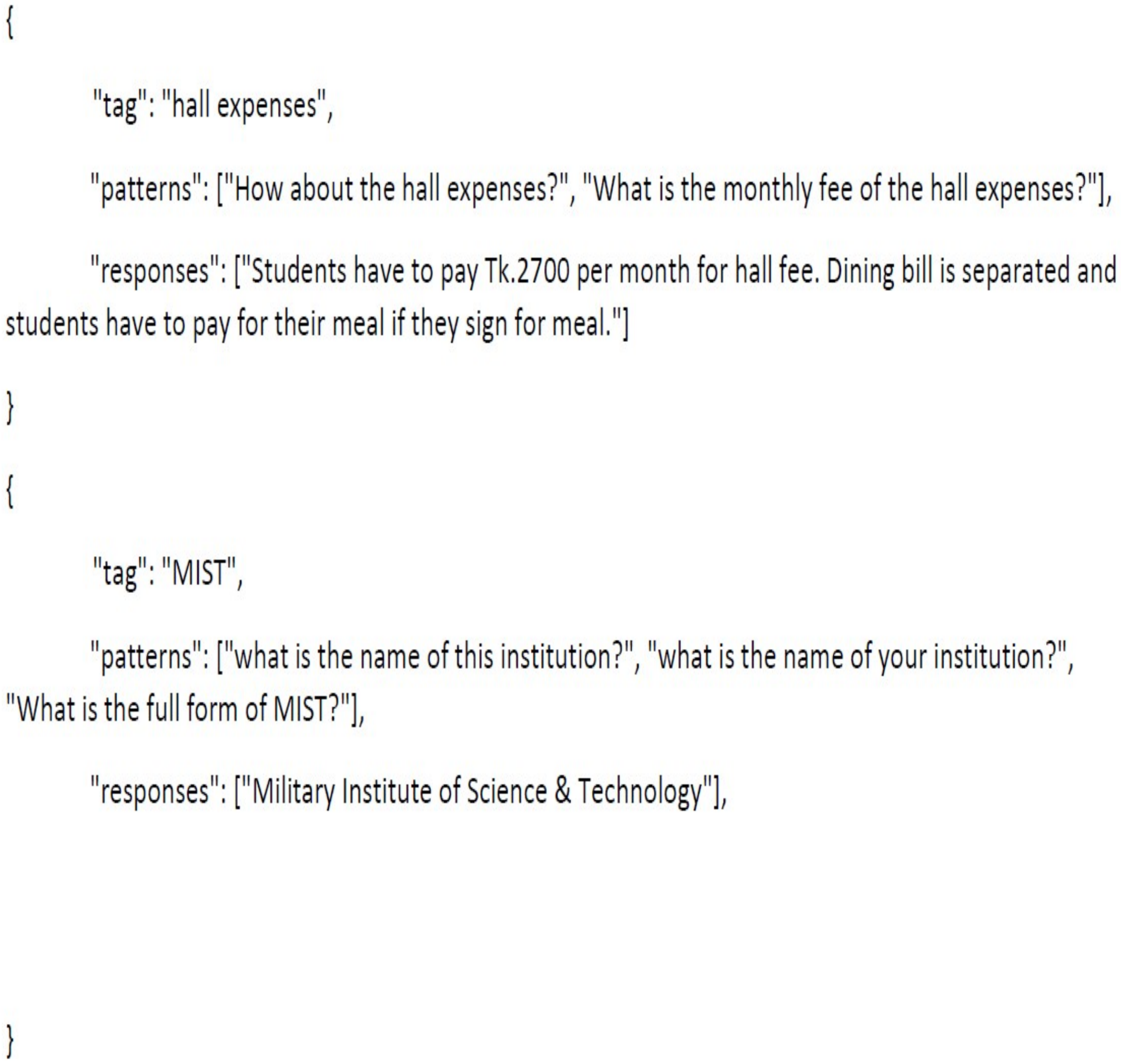


```
{

            “Tag“: "Department",

            “Patterns": ["How many departments does MIST have?", "what is the total number of
departments in MIST?"],

            “Responses": ["MIST has a total of 13 departments."]

}


{

           "tag": "Faculty",

            "patterns": ["How many faculties are there in CSE department?", "what is the strength of
faculties of CSE department?"],

            "responses": ["1 Head of the department, 4 Professors, 3 Associate Professors, 6 Assistant
Professors and 16 Lecturers"]


 }
```

Figure 4.1: Training Dataset 1

In figure 4.1, there are two tags: ”Department” and ”Faculty”. Under ”Department” tag, there are some question patterns like, ”How many departments does MIST have?”, ”What is the total number of departments in MIST?” For these questions, the responses are like, ”MIST has a total of 13 departments.”

Under ”Faculty” tag, there are some question patterns like, ”How many faculties are there in CSE department?”, ”What is the total number of faculties in CSE department?” For these questions, the responses are like, ”1 HoD, 4 Professors, 3 Associate Professors, 6 Assistant Professors and 16 Lecturers.” A random response will be chosen from these responses.

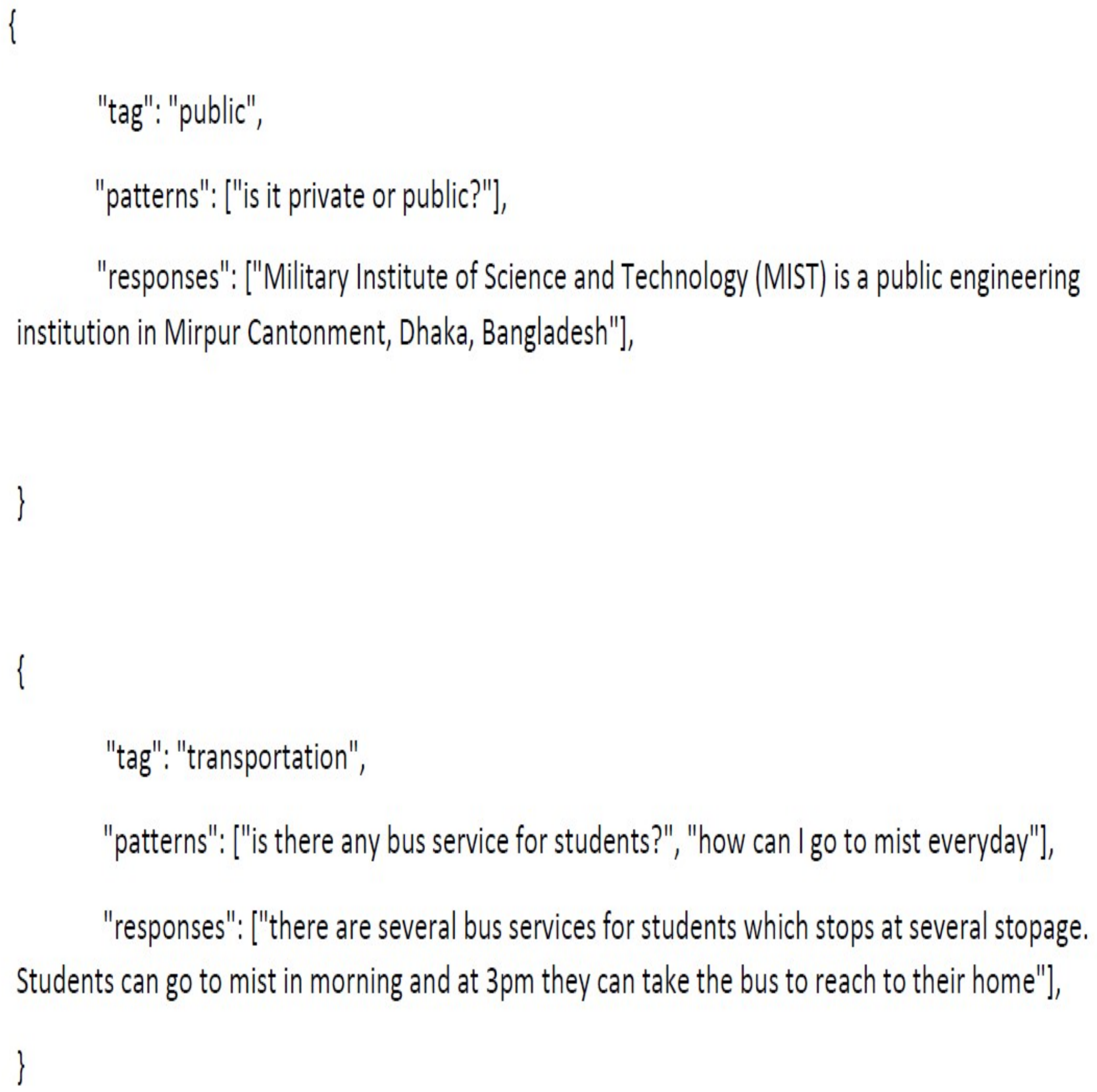


```
{
        "tag": "hall expenses",
        "patterns": ["How about the hall expenses?", "What is the monthly fee of the hall expenses?"],
        "responses": ["Students have to pay Tk.2700 per month for hall fee. Dining bill is separated and
students have to pay for their meal if they sign for meal."]
}
{
        "tag": "MIST",
        "patterns": ["what is the name of this institution?", "what is the name of your institution?",
"What is the full form of MIST?"],
        "responses": ["Military Institute of Science & Technology"],



}
```

Figure 4.2: Training Dataset 2

In figure 4.2, there are two tags: ”hall expenses” and ”MIST”. Under ”hall expenses” tag, there are some question patterns like, ”How about the hall expenses?”, ”What is the monthly fee of the hall expenses?” For these questions, the responses are like, ”Students have to pay Tk.2700 per month for hall fee. Dining bill is separated and students have to pay for their meal if they sign for meal.”

Under ”MIST” tag, there are some question patterns like, ”what is the name of this institution?”, ”what is the name of your institution?”, ”What is the full form of MIST?” For these questions, the responses are like, ”Military Institute of Science & Technology”. A random response will be chosen from these responses.

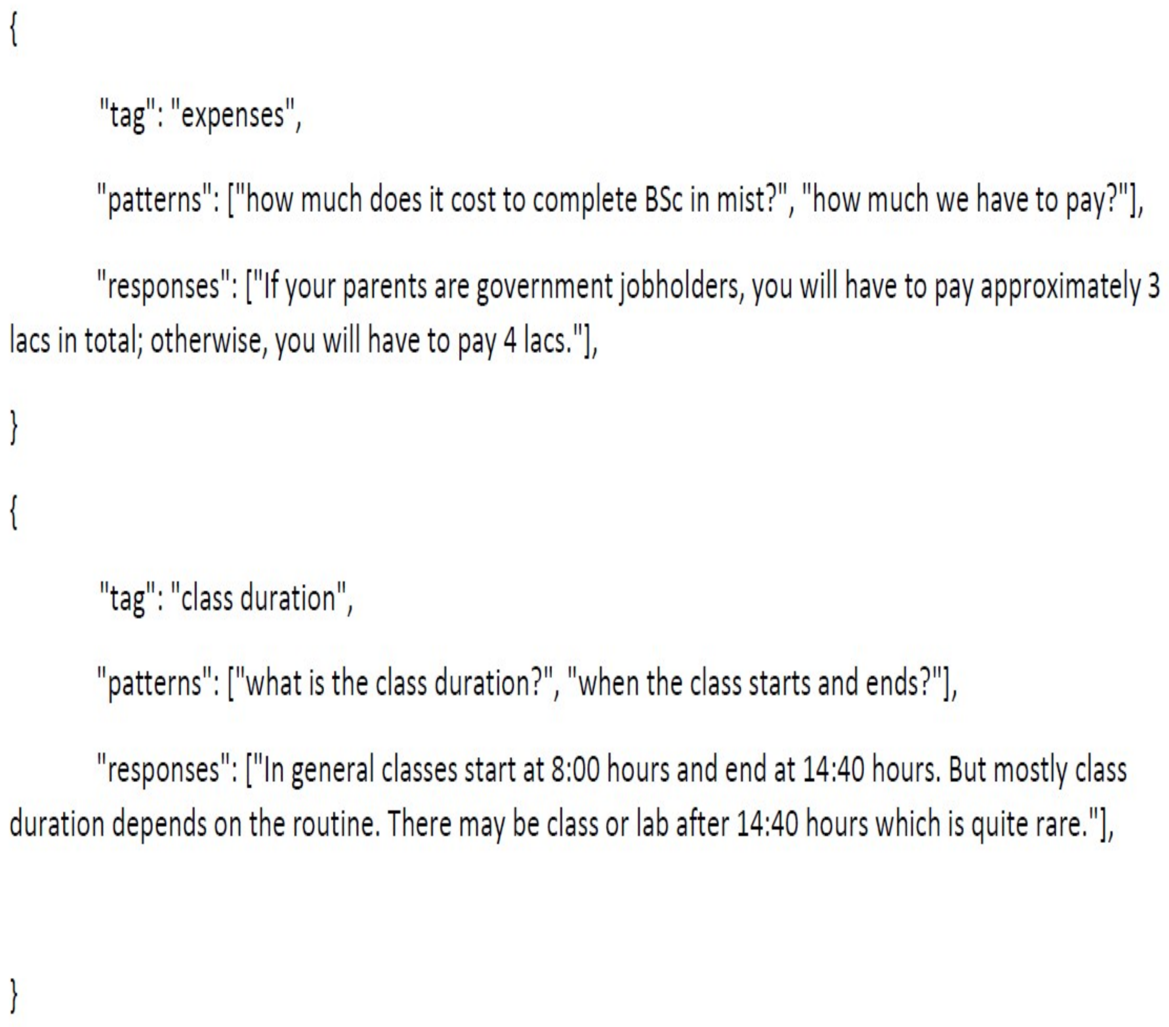


```
{
        "tag": "public",
        "patterns": ["is it private or public?"],
        "responses": ["Military Institute of Science and Technology (MIST) is a public engineering
 institution in Mirpur Cantonment, Dhaka, Bangladesh"],

 }

 {
         "tag": "transportation",
         "patterns": ["is there any bus service for students?", "how can I go to mist everyday"],
         "responses": ["there are several bus services for students which stops at several stopage.
 Students can go to mist in morning and at 3pm they can take the bus to reach to their home"],
 }
```

Figure 4.3: Training Dataset 3

In figure 4.3, there are two tags: ”public” and ”transportation”. Under ”public” tag, there are some question patterns like, ”Is it private or public?” For these questions, the responses are like, ”Military Institute of Science and Technology (MIST) is a public engineering institution in Mirpur Cantonment, Dhaka, Bangladesh.”

Under ”transportation” tag, there are some question patterns like, ”is there any bus service for students?”, ”how can I go to mist everyday?” For these questions, the responses are like, ”There are several bus services for students which stops at several stopage. Students can go to mist in morning and at 3pm they can take the bus to reach to their home”. A random response will be chosen from these responses.



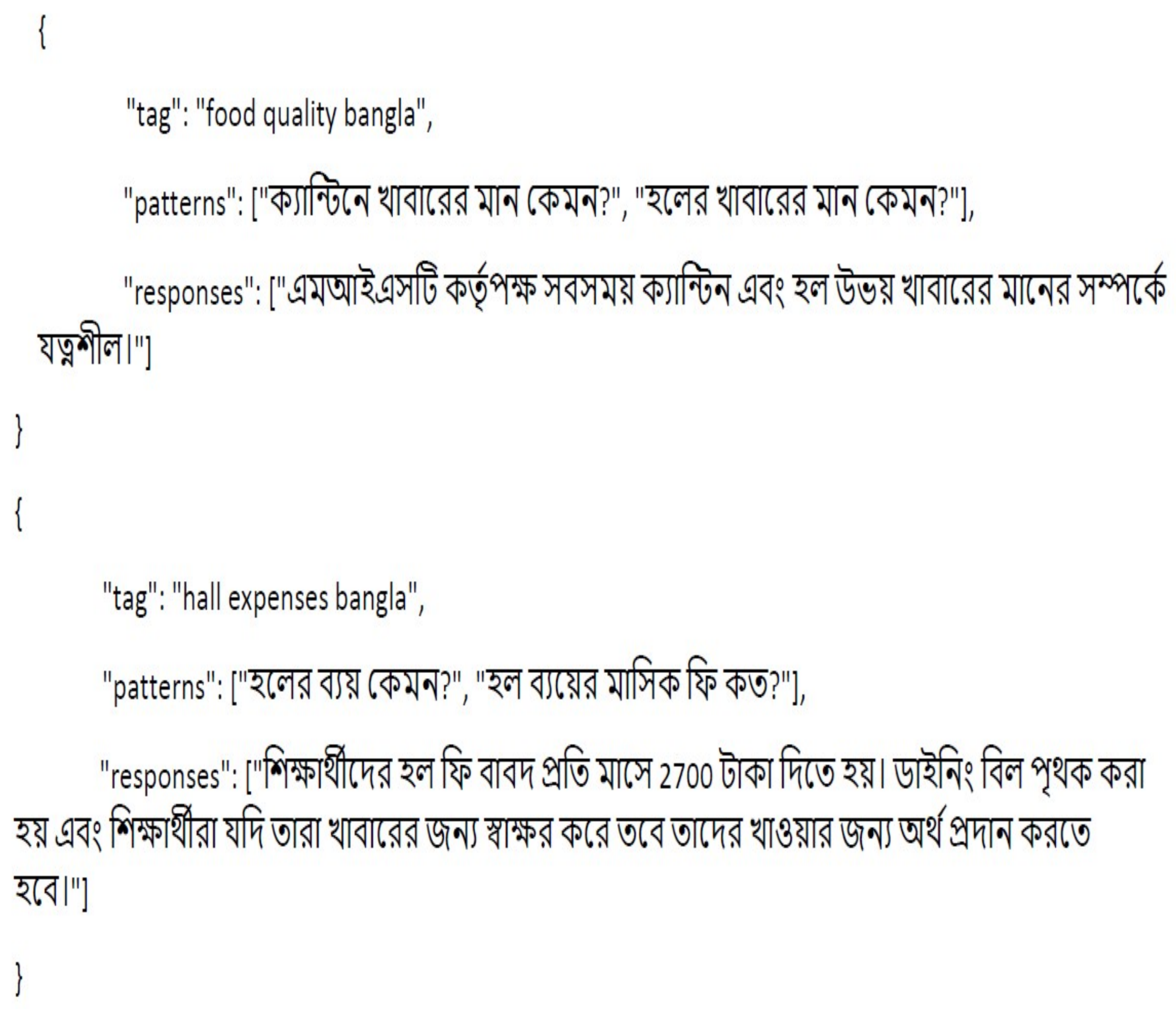
```
{
        "tag": "expenses",
        "patterns": ["how much does it cost to complete BSc in mist?", "how much we have to pay?"],
        "responses": ["If your parents are government jobholders, you will have to pay approximately 3
lacs in total; otherwise, you will have to pay 4 lacs."],
}
{
        "tag": "class duration",
        "patterns": ["what is the class duration?", "when the class starts and ends?"],
        "responses": ["In general classes start at 8:00 hours and end at 14:40 hours. But mostly class
duration depends on the routine. There may be class or lab after 14:40 hours which is quite rare."],


}
```

Figure 4.4: Training Dataset 4

In figure 4.4, there are two tags: ”expenses” and ”class duration”. Under ”expenses” tag, there are some question patterns like, ”how much does it cost to complete B.Sc in mist?”, ”how much we have to pay?” For these questions, the responses are like, ”If your parents are government jobholders, you will have to pay approximately 3 lacs in total; otherwise, you will have to pay 4 lacs.”

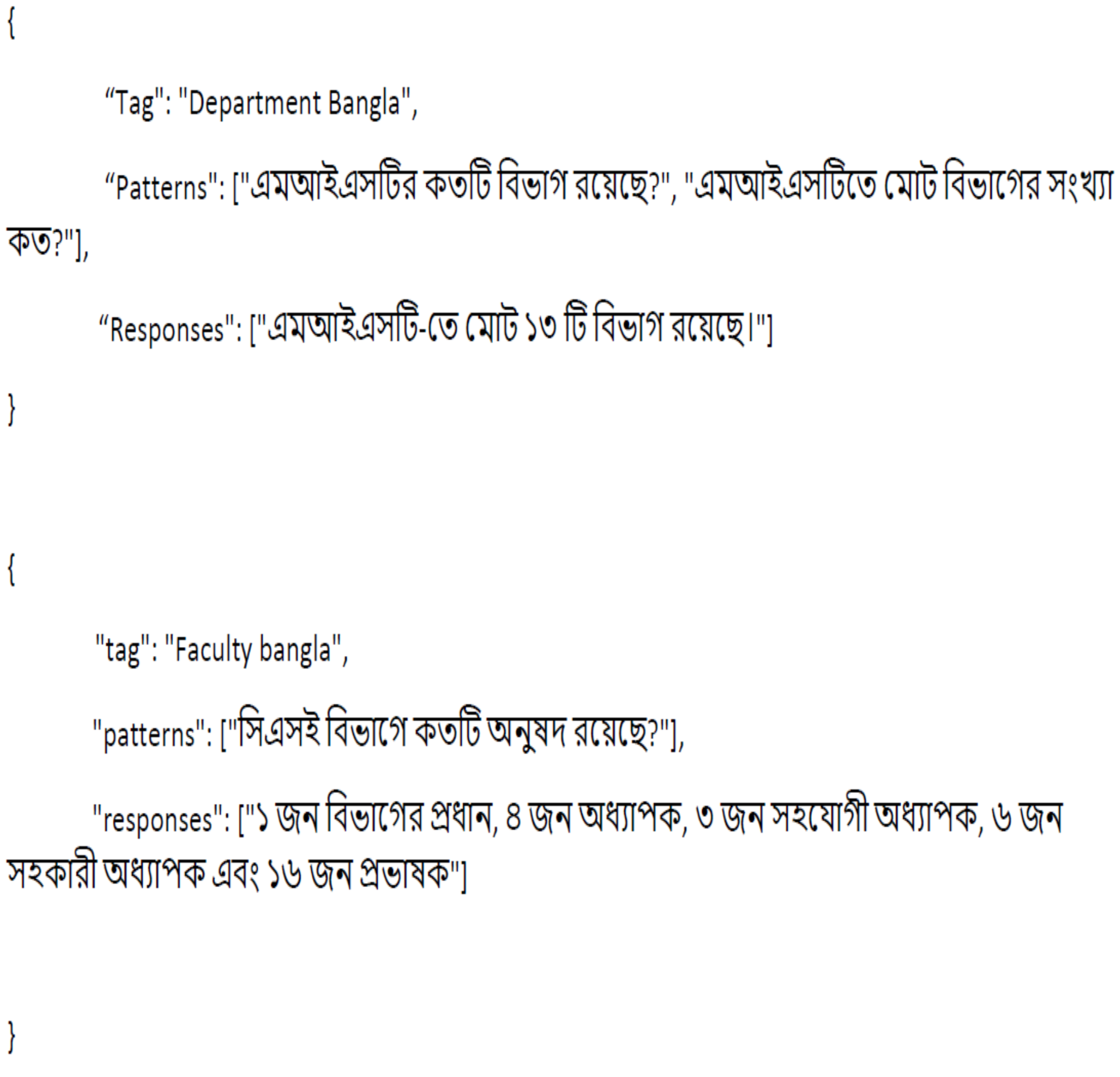
Under ”class duration” tag, there are some question patterns like, ”what is the class duration?”, ”when the class starts and ends?” For these questions, the responses are like, ”In general classes start at 8:00 hours and end at 14:40 hours. But mostly class duration depends on the routine. There may be class or lab after 14:40 hours which is quite rare.” A random response will be chosen from these responses.

There are some dataset given for Bengali language as well. The patterns and responses are same as English language, but they are in Bengali language.



```
  {
          "tag": "food quality bangla",
          "patterns": ["ক্যান্টিনে খাবারের মান কেমন?", "হলের খাবারের মান কেমন?"],
          "responses": ["এমআইএসটি কর্তৃপক্ষ সবসময় ক্যান্টিন এবং হল উভয় খাবারের মানের সম্পর্কে
  যত্নশীল।"]
}
{
        "tag": "hall expenses bangla",
        "patterns": ["হলের ব্যয় কেমন?", "হল ব্যয়ের মাসিক ফি কত?"],
        "responses": ["শিক্ষার্থীদের হল ফি বাবদ প্রতি মাসে 2700 টাকা দিতে হয়। ডাইনিং বিল পৃথক করা
হয় এবং শিক্ষার্থীরা যদি তারা খাবারের জন্য স্বাক্ষর করে তবে তাদের খাওয়ার জন্য অর্থ প্রদান করতে
হবে।"]
}
```

Figure 4.5: Training Dataset 5



```
{
        “Tag": "Department Bangla",
        “Patterns": ["এমআইএসটির কতটি বিভাগ রয়েছে?", "এমআইএসটিতে মোট বিভাগের সংখ্যা
কত?"],
        “Responses": ["এমআইএসটি-তে মোট ১৩ টি বিভাগ রয়েছে।"]
}

{
        "tag": "Faculty bangla",
        "patterns": ["সিএসই বিভাগে কতটি অনুষদ রয়েছে?"],
        "responses": ["১ জন বিভাগের প্রধান, ৪ জন অধ্যাপক, ৩ জন সহযোগী অধ্যাপক, ৬ জন
সহকারী অধ্যাপক এবং ১৬ জন প্রভাষক"]

}
```

Figure 4.6: Training Dataset 6

4.2 Data Preprocessing

For data preprocessing, we’ve used python, Natural Language Tool Kit (NLTK), word_tokenize and Lancaster Stemmer. For each pattern, word tokenizer is used to split the words from the sentences, and word-stemmer is used to reduce the vocabulary by replacing the word with its root form.

For example: [”How many departments does MIST have?”]
Step-1:
Tokenize: [’How’, ’many’, ’departments’, ’does’, ’MIST’, ’have’, ’?’]
Step-2:
Root-form: [’how’, ’many’, ’depart’, ’doe’, ’mist’, ’hav’]

Neural networks and machine learning algorithms require numerical input. Therefore, ‘Bag of Words’ is used to represent the sentences with numbers. Here, each sentence’s length from the pattern is the number of words in our model-vocabulary list. Each position in the list represents a word from the vocabulary-list. If the position in the list is a 1, then that means the word exists in our sentence; if it is a 0, then the word is not present.

Step-3:
Converting the sentences of the pattern into numerical form:
array ([0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1...]

The output is also formatted to make sense to the neural network. Each position in the output-list represents one distinct label/tag; a 1 in any of those positions shows which label/tag is represented.

Step-4:
Converting the corresponding tag of the intent into numerical form
Array ([0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])

4.3 Neural Network Model

We have used Artificial Neural Network (ANN) for the model. For creating the model, We have used Tensor Flow library. The architecture and the model summary are shown in 4.7 & 4.8 respectively:

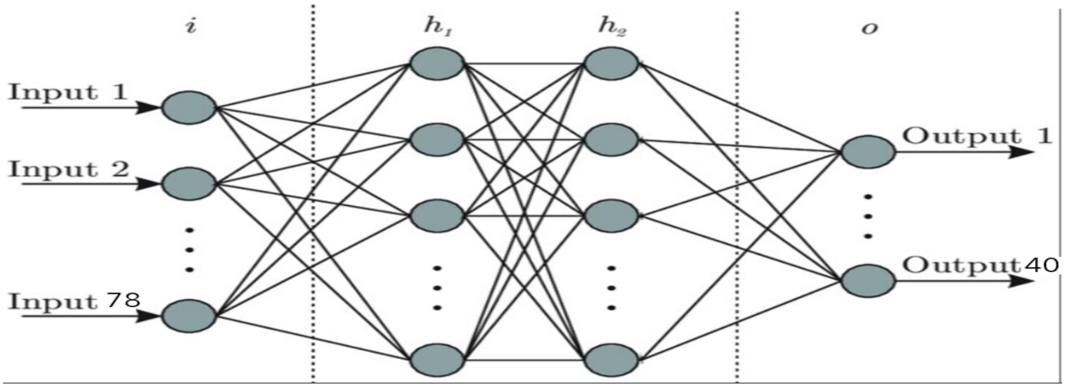
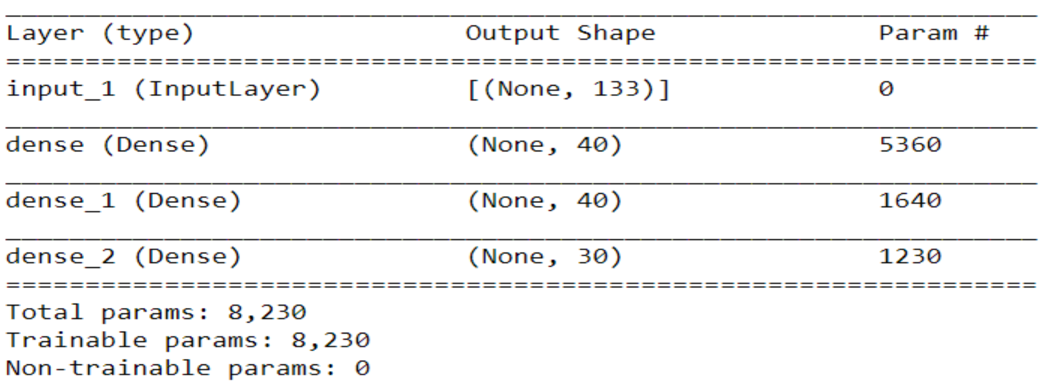


Figure 4.7: Model Architecture



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 133)]             0
_________________________________________________________________
dense (Dense)                (None, 40)                5360
_________________________________________________________________
dense_1 (Dense)              (None, 40)                1640
_________________________________________________________________
dense_2 (Dense)              (None, 30)                1230
=================================================================
Total params: 8,230
Trainable params: 8,230
Non-trainable params: 0
```

Figure 4.8: Model Summary

The size of the input layer is 78 because there are 78 different question-patterns in the dataset. Two ‘dense’ hidden layers are used in the model with 8 neurons each. The output layer has 40 neurons with ‘SoftMax’ activation function. SoftMax activation function will give probability.

4.4 Training & Saving the Model

For fitting the data in the model, 200 epochs are used. In each epoch, the batch-size is 8. After training, the model is saved. Therefore, there is no need to fit the model in the next run. The datasets and the model simply can be loaded. It will save time.

4.5 User Input and Making Prediction

For any input sentence, the model generates a list of probabilities for all of the classes. This makes the process to generate a response look like the following:

Table 4.1: Table of steps and process for user input and making prediction

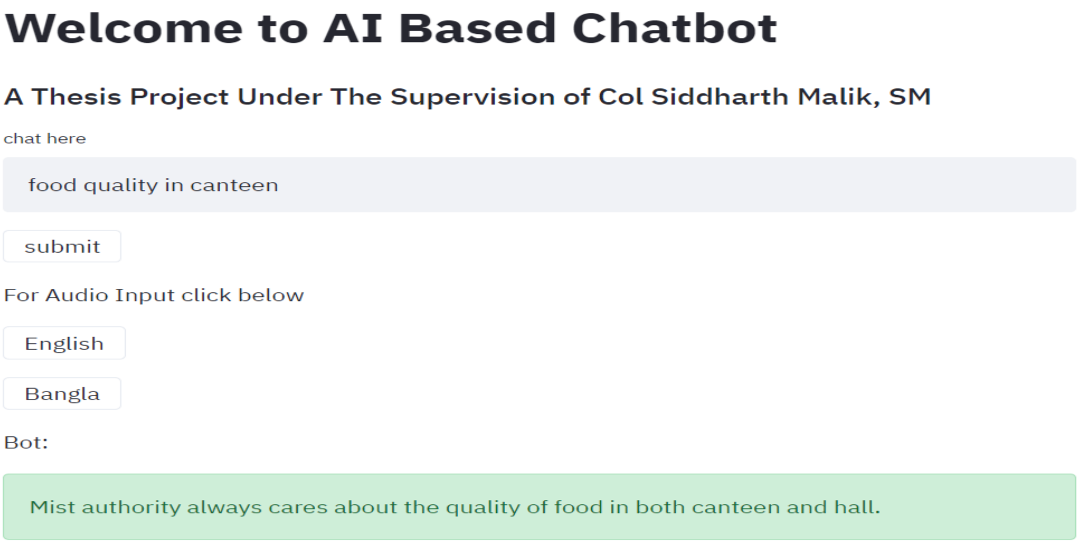
Steps	Process
Input	Audio
	Textual
Language	Bangla
	English
Data Preprocessing	Tokenization
	Stemming
	Numerical Conversion
Prediction	Get a prediction from the model
	Find the most probable class
	Pick a response from that class

If a user wants to give input, then

- First of all, the user may ask query in textual form or in voice form.
- The query maybe in English or in Bangla language.
- When a user finishes asking the query, the query will be preprocessed in the backend. It will be tokenized or converted into separate words. Then, those words will be stemmed, means they will be taken to their root form. After that, those words will be converted into numeric form, because ANN model can't take string as input.
- After the ANN model takes input, it gives a prediction of the possible tag. From the tag, it takes a random response and shows the response on User Interface.

4.6 Graphical User Interface

Streamlit has been used to develop the User Interface and deploy the ANN model. In figure 4.9 and figure 4.10, snapchats of the developed User Interface is shown.
In figure 4.9, there are two options for a user. User can ask query in textual form or in voice form. In the ”chat here” part, user can type in any query and press the submit button. The query will be processed and a reply will be given by the Chatbot. Whether the query is in natural language form or in normal form, the bot can understand the question and give response accordingly.



Welcome to AI Based Chatbot

A Thesis Project Under The Supervision of Col Siddharth Malik, SM

chat here

food quality in canteen

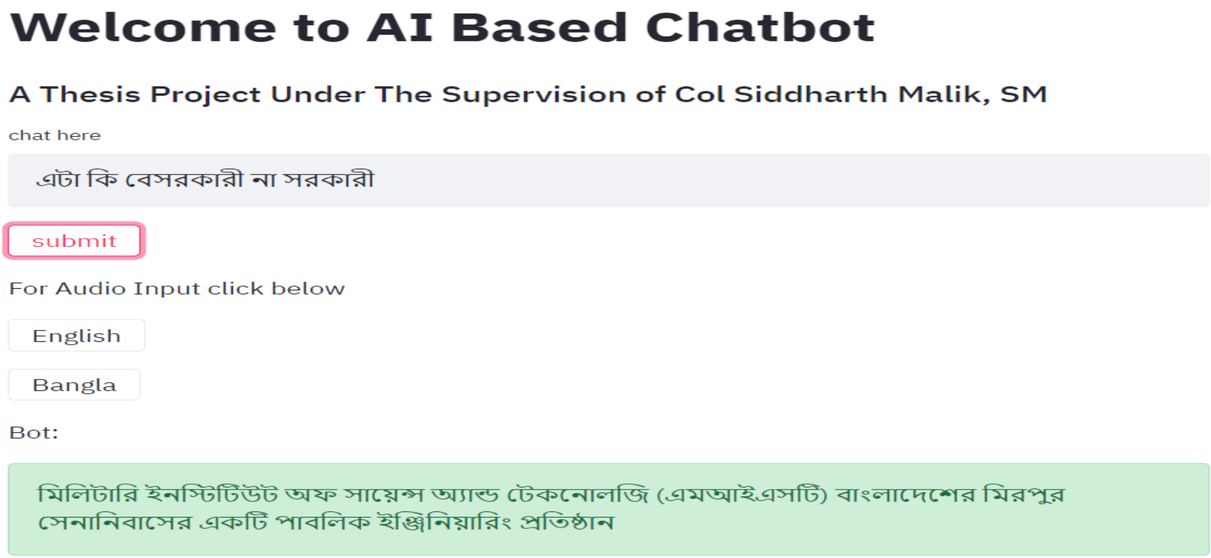
For Audio Input click below

Bot:

Mist authority always cares about the quality of food in both canteen and hall.

Figure 4.9: Graphical User Interface 1

If the user can't type or don't want to type, then, the user can ask the query in voice form. For this, there are two options: English and Bangla. In figure 4.10, if user selects English, then the user have to ask the query in English language and the bot will response in English and if the user selects Bangla, then the user have to ask the query in Banlga language and the bot will response in Bangla.



Welcome to AI Based Chatbot

A Thesis Project Under The Supervision of Col Siddharth Malik, SM

chat here

এটা কি বেসরকারী না সরকারী

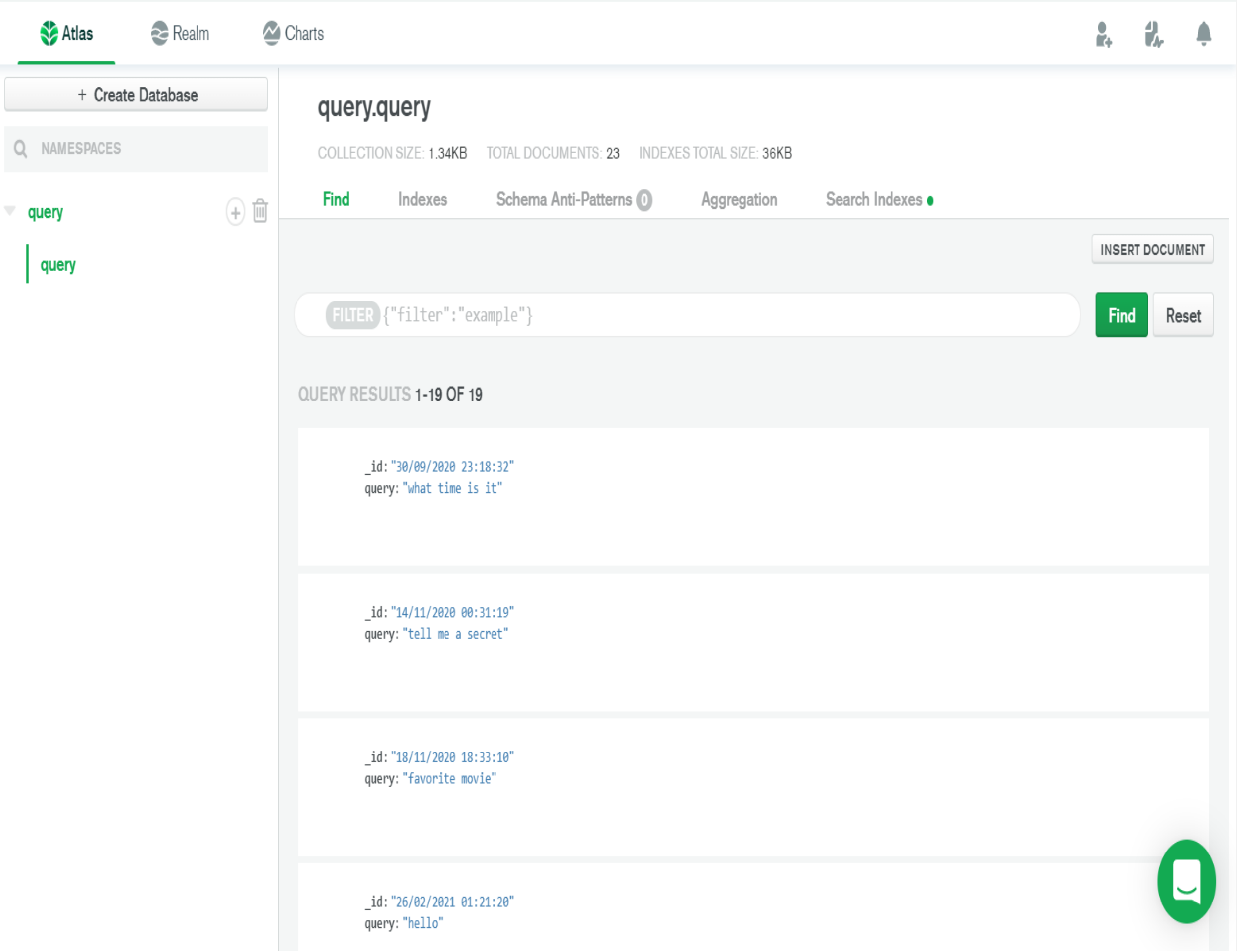
Bot:

মিলিটারি ইনস্টিটিউট অফ সায়েন্স অ্যান্ড টেকনোলজি (এমআইএসটি) বাংলাদেশের মিরপুর সেনানিবাসের একটি পাবলিক ইঞ্জিনিয়ারিং প্রতিষ্ঠান

Figure 4.10: Graphical User Interface 2

4.7 Backend Database

The unsuccessful query will be sent to Realtime database for further development. We have used mongodb for the database, shown in figure 4.11.



{"filter":"example"}

```
_id: "30/09/2020 23:18:32"
query: "what time is it"

_id: "14/11/2020 00:31:19"
query: "tell me a secret"

_id: "18/11/2020 18:33:10"
query: "favorite movie"

_id: "26/02/2021 01:21:20"
query: "hello"
```

Figure 4.11: Backend Database

4.8 Result

The accuracy of the model is approximately 90%. The threshold of matching the input sentences with the patterns is set to minimum 50%. If any query matches less than 50% with the patterns, then that query is sent to the backend database. If new data comes, the model needs to be trained again.

CHAPTER 5
CONCLUSION AND FUTURE WORK

5.1 Conclusion

Bangladesh is rapidly developing in the IT sector and business sector. Renowned companies of Bangladesh like Pureit, Robi, Banglalink, Grameenphone, Daraz etc. have a huge number of customers. To give services to their costomers, they need to maintain a customer-care service center. The chatbots of these companies are really poor and they can only understand scripted questions.

In Bangladesh, university websites doesn't have any chatbot as well, that can provide all the necessary information to the students and faculties. People may have different types of questions regarding a university, but they often don't find all the information or don't know how to find them from the university website.

In this regard, a digital solution AI-based smart assistant chatbot is developed to address several features to overcome all the problems that Bangladeshi people face. The web application can take both lingual query and and textual query. It can understand both Bangla and English language and give response accordingly. It understands Natural Language as well. All the unsuccessful queries are stored in the backend server. These unsuccessful queries can be easily used in future to train the ANN model and increase the accuracy.

5.2 Future Work

One way to improve our Bangla Chatbot's knowledge base in the future is to host it on a crowd-sourcing network. Our chatbot, like Google Translate Bengali, would be able to benefit from user experiences. The greater the amount of contacts, the higher the percentage of appropriate answers to a question that our Chatbot will offer. Crowdsourcing is the method of obtaining needed resources, ideas, or content by enlisting the help of a wide number of people, especially from the online population, rather than conventional workers or suppliers.

In the future, we will use the database from this project to train a chatbot using a neural network model. Also, by incorporating the chatbot into a website, we can try a crowd-sourced model to enrich the chatbot's database.

People keep track of what has been said and what detail has been shared during lengthy talks. Other types of contextual data, such as date/time, location, or information about a person, may be required. Since we lack the necessary data to make it generative, this will have to be achieved in the future.

The AI-based Smart Assistant chatbot is designed only for CSE department of Military Institute of Science and Technology. The dataset is designed for that purpose only. So, in future, it can be expanded and designed for other departments. This can also be customized for any organizations as well.

REFERENCES

[1] J. T. Quah and Y. Chua, “Chatbot assisted marketing in financial service industry,” in *International Conference on Services Computing*, pp. 107–114, Springer, 2019.

[2] “Messenger chatbots: All you need to know about facebook messenger bots.”

[3] A. Lommatzsch, “A next generation chatbot-framework for the public administration,” in *International Conference on Innovations for Community Services*, pp. 127–141, Springer, 2018.

[4] B. R. Ranoliya, N. Raghuwanshi, and S. Singh, “Chatbot for university related faqs,” in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1525–1530, IEEE, 2017.

[5] R. M. von Wolff, J. Nörtemann, S. Hobert, and M. Schumann, “Chatbots for the information acquisition at universities–a student’s view on the application area,” in *International Workshop on Chatbot Research and Design*, pp. 231–244, Springer, 2019.

[6] C. Nischal, T. Sachin, B. Vivek, and K. Taranath, “Developing a chatbot using machine learning,” *International Journal of Research in Engineering, Science and Management*, vol. 3, no. 8, pp. 40–43, 2020.

[7] P. Ponnusamy, A. R. Ghias, C. Guo, and R. Sarikaya, “Feedback-based self-learning in large-scale conversational ai agents,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 13180–13187, 2020.

[8] A. V. Tahami and A. Shakery, “Enriching conversation context in retrieval-based chatbots,” *arXiv preprint arXiv:1911.02290*, 2019.

[9] K. Sugisaki, “Chat-bot-kit: A web-based tool to simulate text-based interactions between humans and with computers,” *arXiv preprint arXiv:1911.00665*, 2019.

[10] T. Bauer, E. Devrim, M. Glazunov, W. L. Jaramillo, B. Mohan, and G. Spanakis, “# metoomaastricht: Building a chatbot to assist survivors of sexual harassment,” in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 503–521, Springer, 2019.

[11] N. Ali, “Chatbot: A conversational agent employed with named entity recognition model using artificial neural network,” *arXiv preprint arXiv:2007.04248*, 2020.

[12] S. Rai, A. Raut, A. Savaliya, and R. Shankarmani, “Darwin: Convolutional neural network based intelligent health assistant,” in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pp. 1367–1371, 2018.

[13] N. Bhartiya, N. Jangid, S. Jannu, P. Shukla, and R. Chapaneri, “Artificial neural network based university chatbot system,” in *2019 IEEE Bombay Section Signature Conference (IBSSC)*, pp. 1–6, 2019.

[14] M. M. Rahman, R. Amin, M. N. K. Liton, and N. Hossain, “Disha: An implementation of machine learning based bangla healthcare chatbot,” in *2019 22nd International Conference on Computer and Information Technology (ICCIT)*, pp. 1–6, IEEE, 2019.

APPENDIX A
CODES

A.1 Data Preprocessing Code

We use this code to preprocess the data…

```
Data Preprocessing

words = []
labels = []
docs_x = []
docs_y = []

for intent in data["intents"]:
    for pattern in intent["patterns"]:
        wrds = nltk.word_tokenize(pattern)
        words.extend(wrds)
        docs_x.append(wrds)
        docs_y.append(intent["tag"])

    if intent["tag"] not in labels:
        labels.append(intent["tag"])

words = [stemmer.stem(w.lower()) for w in words if w != "?"]
words = sorted(list(set(words)))

labels = sorted(labels)

training = []
output = []

out_empty = [0 for _ in range(len(labels))]

for x, doc in enumerate(docs_x):
    bag = []

    wrds = [stemmer.stem(w.lower()) for w in doc]

    for w in words:
        if w in wrds:
            bag.append(1)
        else:
            bag.append(0)

    output_row = out_empty[:]
    output_row[labels.index(docs_y[x])] = 1

    training.append(bag)
    output.append(output_row)


training = numpy.array(training)
output = numpy.array(output)
```

A.2 Model Training Code

We use this code to train and save our model…

```
ANN Model



tensorflow.reset_default_graph()

net = tflearn.input_data(shape=[None, len(training[0])])
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, len(output[0]), activation="softmax")
net = tflearn.regression(net)

model = tflearn.DNN(net)
ANN=model.fit(training, output, n_epoch=200, batch_size=8,
show_metric=True)

Saving The Model

model.save("model.tflearn")
```

A.3 Data Preprocessing of User Input Code

We use this code to preprocess the user input. . .

```
Data Preprocessing of User Input

def bag_of_words(s, words):
    bag = [0 for _ in range(len(words))]

    s_words = nltk.word_tokenize(s)
    s_words = [stemmer.stem(word.lower()) for word in s_words]

    for se in s_words:
        for i, w in enumerate(words):
            if w == se:
                bag[i] = 1
    test = numpy.array(bag)

    return test

def chat():
    print("Start␣talking␣with␣the␣bot␣(type␣quit␣to␣stop)!")
    while True:
        inp = input("You:␣")
        if inp.lower() == "quit":
            break

        results = model.predict([bag_of_words(inp, words)])[0]
        results_index = numpy.argmax(results)
        tag = labels[results_index]

        if results[results_index] > 0.5:

            for tg in data["intents"]:

                if tg['tag'] == tag:
                    responses = tg['responses']


            result = random.choice(responses)
            print(result)
        else:

            print("i␣didn’t␣get␣it.␣Ask␣another␣question")
```

A.4 Audio Input Code

We use this code to take audio input…

```
import streamlit as st
import pyaudio
import speech_recognition as sr
r = sr.Recognizer()


if st.button("submit"):
        inp=string

st.write("For␣Audio␣Input␣click␣below")

if st.button(’English’):
        with sr.Microphone() as source:
        # read the audio data from the default microphone
                st.write("speak")
                audio_data = r.record(source, duration=8)
                st.write("Recognizing...")
                # convert speech to text
                text = r.recognize_google(audio_data)
                st.warning(text)
                inp=text

if st.button(’Bangla’):
        with sr.Microphone() as source:
        # read the audio data from the default microphone
                st.write("speak")
                audio_data = r.record(source, duration=8)
                st.write("Recognizing...")
                # convert speech to text
                text = r.recognize_google(audio_data, language="bn-BD")
                #using bangla
                # text = r.recognize_google(audio_data)
                st.warning(text)
                inp=text
```

A.5 User Interface Code

We use this code to develop the User Interface...

```
import pymongo
from pymongo import MongoClient
import streamlit as st
import nltk
from nltk.stem.lancaster import LancasterStemmer
stemmer = LancasterStemmer()

import numpy
import tflearn
import tensorflow
import random
import json
import pickle
import pyaudio
import speech_recognition as sr
from PIL import Image

from datetime import datetime
now = datetime.now()
dt_string = now.strftime("%d/%m/%Y␣%H:%M:%S")

# cluster= MongoClient("mongodb+srv://chatbot:a4s@cluster0.rqhb3.
mongodb.net/query?retryWrites=true&w=majority")
# db=cluster["query"]
# collection=db["query"]


# initialize the recognizer
r = sr.Recognizer()

import base64
@st.cache(allow_output_mutation=True)
def get_base64_of_bin_file(bin_file):
    with open(bin_file, ’rb’) as f:
        data = f.read()
    return base64.b64encode(data).decode()
```

```

def set_png_as_page_bg(png_file):
    bin_str = get_base64_of_bin_file(png_file)
    page_bg_img = ’’’
<style>
body{
background-image:␣url("data:image/png;base64,%s");
background-size:␣cover;
}
</style>
’’’ % bin_str

    st.markdown(page_bg_img, unsafe_allow_html=True)
    return

# set_png_as_page_bg(’.jpg’)


st.title("Welcome␣to␣AI␣Based␣Chatbot")
st.subheader("A␣Thesis␣Project␣Under␣The␣Supervision␣of␣Col
Siddharth␣Malik,␣SM")

inp="hello"
with open("intents.json", encoding=’UTF-8’) as file:
        data = json.load(file)

with open("data.pickle", "rb") as f:
                words, labels, training, output = pickle.load(f)

tensorflow.reset_default_graph()

net = tflearn.input_data(shape=[None, len(training[0])])
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, len(output[0]),
activation="softmax")
net = tflearn.regression(net)

model = tflearn.DNN(net)

```

```

model.load("model.tflearn")


def bag_of_words(s, words):
        bag = [0 for _ in range(len(words))]

        s_words = nltk.word_tokenize(s)
        s_words = [stemmer.stem(word.lower()) for word in s_words]

        for se in s_words:
                for i, w in enumerate(words):
                        if w == se:
                                bag[i] = 1

        return numpy.array(bag)





string = st.text_input("chat␣here")
if st.button("submit"):
        inp=string

st.write("For␣Audio␣Input␣click␣below")

if st.button('English'):
        with sr.Microphone() as source:
        # read the audio data from the default microphone
                st.write("speak")
                audio_data = r.record(source, duration=8)
                st.write("Recognizing...")
                # convert speech to text
                # text = r.recognize_google(audio_data,
                language="bn-BD") #using bangla
                text = r.recognize_google(audio_data)
                st.warning(text)
                inp=text

```

```
if st.button(’Bangla’):
        with sr.Microphone() as source:
        # read the audio data from the default microphone
                st.write("speak")
                audio_data = r.record(source, duration=8)
                st.write("Recognizing...")
                # convert speech to text
                text = r.recognize_google(audio_data,
                language="bn-BD") #using bangla
                # text = r.recognize_google(audio_data)
                st.warning(text)
                inp=text


# if inp.lower() == "quit":
#       break
# st.write(type(inp))
results = model.predict([bag_of_words(inp, words)])[0]
results_index = numpy.argmax(results)
tag = labels[results_index]

if results[results_index] > 0.5:

    for tg in data["intents"]:

        if tg[’tag’] == tag:
            responses = tg[’responses’]

    st.write("Bot:␣")
    result = random.choice(responses)
    st.success(result)
else:
    st.write("Bot:␣")
    st.error("i␣didn’t␣get␣it.␣Ask␣another␣question")
    #collection.insert_one({"_id": dt_string, "query": inp})


st.sidebar.markdown("<h1␣style=’text-align:␣center;
color:␣black;’>Developed␣By</h1>", unsafe_allow_html=True)
# st.sidebar.title("Developed␣By")
```

```
st.sidebar.subheader("1.Rezwan␣Rownok")


img=Image.open("rownok.jpg")
st.sidebar.image(img,width=300,caption="Rezwan␣Rownok")
st.sidebar.write("Email:␣rownokrezwan@gmail.com")
st.sidebar.write("linkedin:␣https://www.linkedin.com/in
/rezwan-rownok-538978198")

st.sidebar.subheader("2.Shahriar␣Rahman␣Khan")
img=Image.open("shahriar.jpg")
st.sidebar.image(img,width=300,caption="Shahriar␣Rahman")
st.sidebar.write("Email:␣shahriarkhan.ndc@gmail.com")
st.sidebar.write("linkedin:␣https://www.linkedin.com/in
/shahriar-rahman-khan-nehal-322351152")

st.sidebar.subheader("3.Samiha␣Raisa␣Zaman")
img=Image.open("sam.jpg")
st.sidebar.image(img,width=300,caption="Samiha␣Raisa")
st.sidebar.write("Email:␣samiharaisa031@gmail.com")
st.sidebar.write("linkedin:␣https://www.linkedin.com/in
/samiha-raisa-zaman-0b472512b")

st.sidebar.subheader("4.Sharmila␣Rahman␣Prithula")
img=Image.open("prithula.jpg")
st.sidebar.image(img,width=300,caption="Sharmila␣Rahman")
st.sidebar.write("Email:␣sharmilarahmanprithula@gmail.com")
st.sidebar.write("linkedin:␣")
```