



**Thesis Paper**

**Implementation of ALICE chatbot as domain specific  
knowledge bot for BRAC U (FAQ bot)**

**By**

**Johan Rahman (11141002)**

**Supervisor: Dr. Supratip Ghose**

## Contents

1.Introduction .....	4
2.ALICE Overview .....	7
3.Related work.....	9
4.System Architecture of FAQ bot.....	11
5.Experimental Design .....	23
6.Conclusion .....	27
7.Reference .....	28

# Abstract

---

In this paper, a proposal is carried on to explain the design of a chat bot specifically tailored as a undergraduate student information system that helps students in BRAC University with admission and course information's. In particular, the proposal investigates the implementation of ALICE chat bot system as a domain specific chatterbox named FAQ bot, our work will show how a chat bot can work as domain specific information system and experiments on how the system's accuracy could be improved based on a specific domain.

# 1. Introduction

---

A chatbot is software that is used to interact between a computer and a human in natural language. Naturally, it can extend daily life, such as help desk tools, automatic telephone answering systems, to aid in education, business and e-commerce. In our thesis we tried to implement a domain specific knowledge system to deliver answer to frequently asked questions in BRAC U environment, we worked with A.L.I.C.E (Artificial Linguistic Internet Computer Entity) to develop our FAQ chatterbot system.

## 1.1 Why is FAQ chatbot useful for BRAC University?

The implementation of this project on a University environment is particularly useful for students' looking for information regarding admission in BRAC U, and its course curriculum. Even though most of the information is available on the web, students often like to have personal interaction with the advisor. In such an environment, a chat robot could be designed for providing academic advice. The main goal of such a system is to conveniently retrieve information without having to look or browse several web pages to fetch answers to frequently asked questions.

## **1.2 What is Domain-specific knowledge system?**

Domain specific knowledge system is a set of knowledge base consisting of AIML files. In our knowledge system we included CseCourse.aiml, AdmissionInfo.aiml, these files consists of different patterns and templates that relates to FAQ in BRAC U environment and generates responses to them, this along with modified conversational knowledge base in ALICE helps our FAQ chatterbot to stay focused only on chat conversational topic which are related to BRAC U environment thus keeping interaction limited to a specified field of knowledge.

## **1.3 Why ALICE?**

ALICE open source chatbot developed by Dr.Wallace, which is based on natural language understanding and pattern matching; also the architecture of the “chatbot engine” and the “language knowledge model” are clearly separated, which gives us the opportunity to easily inject a newly developed knowledge model such as our FAQ knowledge model.

In this paper we will further give an overview of ALICE and how the knowledge base of ALICE is implemented using AIML files, we will also look into some related work with ALICE and learn how to keep the system focused on a specific domain. We will further explain the architecture of our FAQ chatbot tailored for BRAC U, and find out how the system performance improves.

## 2. ALICE Overview

---

A.L.I.C.E (Artificial Linguistic Internet Computer Entity) is an award winning open source natural language artificial intelligence chat robot which utilizes AIML (Artificial Intelligence Markup Language) to form responses to queries. The main knowledge bases of ALICE are stored on different AIML files; ALICE is available to the public for free under the GNU license.

### 2.1 AIML

The ALICE software implements AIML (Artificial Intelligence Markup Language) a non-standard evolving markup language for creating chat robots. The primary design feature of AIML is minimalism. Compared with other chat robot languages, AIML is perhaps the simplest. The basic unit of knowledge in AIML is called a category. Each category consists of an input or question, and output or answer, and an optional context. The question is called the pattern. The answer or response is called the template. The two types of optional context are called "that" and "topic." The AIML pattern language is simple, consisting only of words, spaces, and the wildcard symbols \_ and \*. The pattern matching

language is also very simple, for example permitting only one wild-card ('\*') match per pattern.

AIML is an XML language, implying that it obeys certain grammatical meta-rules. The choice of XML syntax permits integration with other tools such as XML editors. Another motivation for XML is its familiar look and feel, especially to people with HTML experience. Natural language is extremely complex. A person is presumed to have implicit or meta-knowledge about the world when having conversation. In human to human conversations, implicit and meta-knowledge is always used. This in some way poses as a huge challenge when customizing the FAQ bot for BRACU. The paper will clearly explain some tags that were used to somewhat capture this behavior and produce expected results. An AIML chat robot begins and ends with the <alice> and </alice> tags respectively. For this project, ALICE was renamed FAQ chatbot which stands for frequently asked questions chatbot. So henceforth, this paper will be referring to the chatbot as FAQ chatbot.



## 3. Related work

---

There have been many research studies where ALICE was used. In this section we are going to give an overview of some research studies which are relevant to domain specific knowledge chatterbot system.

First of which is a research based on dialog system. A dialog system can be divided into two major part, High-level dialog system and Low-level dialog system [1]. High-level dialog systems are complex systems that posses learning algorithms and reasoning algorithms to analyze a query and then generate a response based on it. For our project we are going to use Low-level dialog system, low-level dialog systems are referred as systems which tend to reply to question based on simple pattern matching rather than understanding or analyzing the question. One such implementation of this type of system was developed for Chinese students as an English conversational partner [2]. This study mainly focused on ALICE used solely as conversational agent, the study produced unexpected result as majority of the users did not like the response of the chatterbot and left bad comments on the system, the study also showed most users only interacted with the system for very short amount of time. The reason behind this result may be related to the limited amount of knowledge base entries or lack of the system to understand the context of the conversation.

Another similar study on chatterbot system which is based on ALICE was developed to deliver telecomm related definitions [3]. This study showed chatterbot system performed much better when the domain specific knowledge was coupled with conversational knowledge base, rather the domain specific knowledge alone. The studies show that a low-level chatterbot system performed best when domain specific knowledge and conversational knowledge bases were used together, rather than solely using the system as a conversational partner or solely a domain specific knowledge bot.

One of the best working example of a chatterbot is ANNA [4] a virtual assistant at IKEA help centre [5].The chatbot is a life like conversational agent providing an interactive and personal way for users to get answers and assistance on the website. A customer simply chats with an assistant, and the assistant acts as an agent, providing answers, processing data and solving customer problems. The chatbot provides frontline support, so the customer service staff can concentrate on more complex tasks. A chatbot like ANNA on business sites are regarded as shopping bots and upon installation will Greet users on the site and answer FAQ.

## 4. System Architecture of FAQ bot

---

In this project, our responsibilities included reading the user inputs and then respond to the query, while trying to keep the conversation related to BRAC U environment. . The first step in developing the FAQ bot consisted of extensive brainstorming and writing down as many questions as possible. This assisted in allowing FAQ bot to intelligently match pattern (inputs). For doing that we created new AIML files and coupled it with the conversational knowledge base of ALICE bot. We created AdmissionInfo.aiml and CourseInfo.aiml and injected the files into ALICE knowledge base; we also created proj.aiml to change between aiml files to answer questions in context of the topic also and focused on modifying that.aiml and personality.aiml so that our FAQ bot always try to focus on our FAQ domain.

### 4.1 Basic AIML structure in FAQ bot

Each AIML file start with an <aiml> tag that represents the AIML version being used, and it contains the AIML elements which consists of data objects called AIML objects. These objects are made up of units called **topics** and **categories**. The topic is an optional top level element, has a name attribute and a set of categories related to that topic. Each category contains a pattern which represents the user input and a template implies FAQ bot response.

## Example from AdmissionInfo.aiml

```
< aiml version="1.0" >
< topic name= " admission info" >

<category>
<pattern>What is the admission requirements for BRACU undergraduate
programs
</pattern>
<template>
Minimum GPA of 3.0 in SSC and HSC separately.
Minimum GPA of 2.5 in O-Levels in five subjects and A-Levels in two subjects
</template>
</category>
..
..
</topic>
</aiml>
```

## 4.2 Uses of tag

ALICE has a set of three different categories, for our FAQ bot we mostly used recursive category which is a property of template. The template calls the pattern matcher recursively using <srail> and <sr> tags which refers to simply recursive artificial intelligence and symbolic reduction but first we give some example of other categories like the atomic category and default category.

Atomic category are those with patterns that does not have wildcards "\*", example of atomic category

..

```
<category>
```

```
<pattern>Who is your master?
```

```
</pattern>
```

```
<template>
```

Johan is the botmaster.

```
</template>
```

```
</category>
```

..

In the above example:

If the user Input: *Who is your master?*

Then FAQ bot output: *Johan is the botmaster.*

Default Categories are those with patterns has a wildcards, example of default category:

..

```
<category>
```

```
<pattern>* about cse370?
```

```
</pattern>
```

```
<template>
```

Focus of this course is learn the concept and methods of storing and manipulating data.

</template>

<category>

..

In the above example:

If user inputs: *Do you know about cse370?*

FAQ output: *Focus of this course is learn the concept and methods of storing and manipulating data.*

Recursive category <srai> tag is the symbolic reduction tag. This allows minimalism. The following illustration will clearly show the importance of this tag.

..

<category>

<pattern>What do you know about cse370?</pattern>

<template>

<srail>CSE370?</srail>

</template>

<category>

<category>

<pattern>\* do you know about cse370?</pattern>

<template>

<srail>CSE370?</srail>

</template>

<category>

<category>

<pattern>\*what informations do you know about CSE370?</pattern>

<template>

<srail>CSE370?</srail>

</template>

<category>

<category>

<pattern>tell me more \*about CSE370?</pattern>

<template>

<srail>CSE370?</srail>

</template>

<category>

```
<category>
<pattern>*who takes CSE370?</pattern>
<template>
<srail>CSE370?</srail>
</template>
<category>
<category>
<pattern>information*CSE370?</pattern>
<template>
<srail>CSE370?</srail>
</template>
<category>
<category>
<pattern>*who is the instructor for CSE370?</pattern>
<template>
<srail>CSE370?</srail>
</template>
<category>
<pattern>*course detail CSE370?</pattern>
<template>
<srail>CSE370?</srail>
```



</template>

<category>

<category>

<pattern>CSE370?</pattern>

<template>

Focus of this course is learn the concept and methods of storing and manipulating data.

The instructor for Database System is Hossain Arif .

</template>

</category>

..

In the above example:

If the user inputs: *course detail CSE370?*

Or *tell me more about CSE370?*

Or *What information do you know about CSE370?*

Or *What do you know about cse370?*

...

The FAQ bot output: *Focus of this course is learn the concept and methods of storing and manipulating data.*

*The instructor for Database System is Hossain Arif.*

We used these tags in creating our AdmissionInfo.aiml and CourseInfo.aiml files. The advantage of using this tag as noted before is that it allows minimalism. One is able to program numerous possible responses and let it point to just one answer. Another advantage of using <srai> tag would be that if for example in the previous example, if the instructor for Database was changed from X to Y, All one is required to do is to change the one answer as opposed to changing all possible answers.

#### **4.3 Conversational agent modification**

FAQ bot is able to understand the flow of conversations by capturing the questions and going into a particular topic, <topic> allows FAQ bot to prefer responses that deal with the topic currently being discussed. This creates topic based conversation, yet still has the ability to move from one subject to another.

For example, if the questions were asked in relation to Admission Information, the file AdmissionInfo.aiml will be put into use.

Example:

<USER> Hi !

<FAQ Bot> Hi [Still in general topic ]

<USER> How are you?

<FAQ bot> Fine thank you [general topic]

<USER> Can you please tell me more about Admission?

<FAQ bot> Let's move on the Admission Information [ AdmissonInfo topic ]

<USER> Does everyone have to take the admission test?

<FAQ Bot> Yes

<USER> How to complete an application for admission?

<FAQ Bot> To complete an application, you have to fill up the application form and sign it and return it to admission office with two photograph and 1000tk fee.

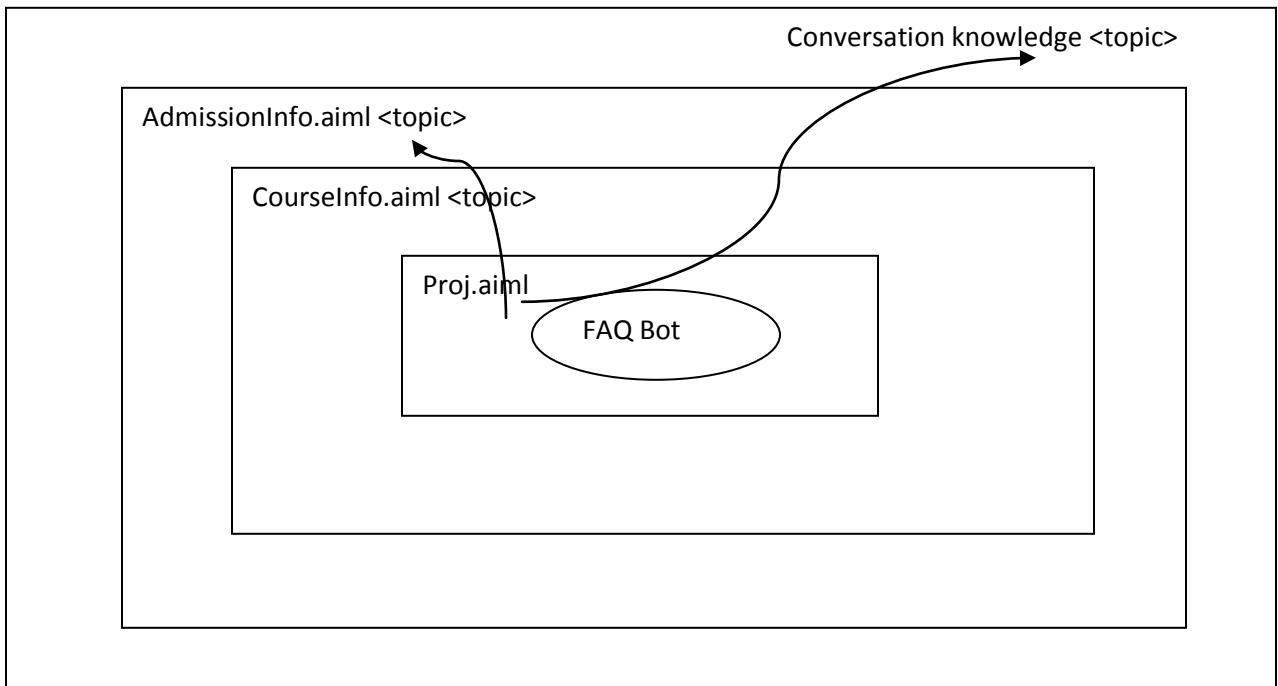


Fig 1.0

This diagram shows the ease of changing topic, proj.aiml uses pattern matching by looking at <topic> to move from one aiml file to another

FAQ bot can also remember its previous output and see if the user is asking further questions based on previous output, for this we used the **that.aiml** file from the ALICE conversational basebot.

Example of <that> tags

..

<category>

<pattern>\*Faculty</pattern>

<template>Who's your favorite faculty?</template>

</category>

<category>

<pattern>\*</pattern>

<that> Who's your favorite faculty?</that>

<template><person/> Oh you know what courses he takes?</template>

</category>

..

User: *Dr.Mumit is a great faculty*

FAQ bot: *Who's your favorite faculty?*

User: *Dr.Mumit*

FAQ bot: *Dr.Mumit, Oh you know what courses he takes?*

FAQ bot can also throw random responses. This in a way makes a conversation more natural because human responses to a particular question may differ from time to time, but providing the user with a proper response in context of the question.

<category>

<pattern>tell me something about TARC semester</pattern>

<template>

<random>

<li>TARC is the best experience ever.</li>

<li>Only BRAC U offers a residential semester at TARC.</li>

<li>TARC is the best place to develop your leadership skills.</li>

<li>TARC semester is compulsory for all students of BRAC U</li>

<li>Most students love TARC</li>

</random>

</template>

</category>

## 5. Experimental Design

---

The main objective of the paper is to analyze the accuracy of a domain specific knowledge bot, to do this we created our FAQ knowledge base of 150 entries and classified our knowledge bases into three categories.

- Admission Information
- Course Information
- Faculty Information

With this knowledge base we have addition approximately 23000 conversation entries from the basebot, to analyze the accuracy we made two different chatterbox system and named them FAQ (limited) which consisted of the 150 entries from FAQ knowledge base and addition 5000 conversational entries from the base bot, the second chatterbox system is FAQ (full) consisted of the 150 FAQ knowledge base entries and all of the 23000 conversation entries from the base bot.

System name	FAQ entries	Basebot entries	Total entries
FAQ(limited)	150	5000	5150
FAQ(full)	150	23000	23150

FAQ (limited) is composed with approximately 5000 conversation entries from the ALICE basebot to avoid the potential risk of not returning any responses to participants, at the same time we wanted to test if limiting the conversation entries in basebot of ALICE but modifying topic changing and personality related files such “that.aiml, personality.aiml and proj.aiml” to the context of the domain improves the accuracy of response relating to the domain of the FAQ knowledge bot. The FAQ (limited) bot also consists of all the FAQ knowledge base entries in addition with the conversational entries.

FAQ (full) is composed with all the conversational entries from the basebot in ALICE but no files are modified to the context of the domain knowledge. It is basically the conversation agent of ALICE with addition to our FAQ bot knowledge base entries. This system has a lot of conversational knowledge base entries but the modified personality and topic changing AIML files are not used.

To measure the two chatterbox system’s performance we introduced two evaluation variables; Satisfactory and Unsatisfactory.

Satisfactory responses are based on how appropriate the system responded in context to the question asked by the user; this value is either true or false determined by the user, depending on how accurately the system responded to the query.



Unsatisfactory responses are just the opposite of satisfactory, if the response by the system is out of the context and the user determines the response as inaccurate or irrelevant to the query, the users are asked to determine the response as unsatisfactory response.

We conducted a survey among a group of final year student from BRAC U. We asked students to interact with the chatterbox FAQ (limited) and FAQ (full) and told them to stay focused on the context, in our case Admission Info, Course Info, Faculty Info. They were told to ask equal number of questions on each sub-topic and then mark each of the systems response as either satisfactory or unsatisfactory, at the end of the survey the sum of all satisfactory responses are taken into accounts with respect to Admission Info, Course Info and Faculty Info separately.

Context of conversation	Satisfactory	Unsatisfactory
Admission Info	70%	30%
Course Info	80%	20%
Faculty Info	60%	40%

Table 1.1

Table 1.1 shows the statistics of data collected from the survey when students interacted with the FAQ(limited) chatterbox.

Context of conversation	Satisfactory	Unsatisfactory
Admission Info	60%	40%
Course Info	60%	40%
Faculty Info	50%	50%

Table 1.2

Table 1.2 shows the statistics of data collected from the survey when students interacted with the FAQ(full) chatterbox.

It was seen that FAQ(limited) performed better in all the three sub-categories, this may be because of the proj.aiml file used in the limited set of conversational knowledge base along with the modified that.aiml file and personality.aiml file, further observations reveal that the lowest amount of satisfactory percentage was found on the responses related to faculty information, one of the reason behind this may be due to the use of single CourseInfo.aiml to include pattern related to faculties as well, thus the proj.aiml could not point to a specific topic for questions related to faculty, thus responding with templates which are not in proper context of conversation.

## 6.Conclusion

---

A chatbot is one of the easiest way to fetch information from a system without having to think for proper keywords to look up in a search engine or browse several web pages to collect information, users can easily type their query in natural language and retrieve information, in this project we looked into how ALICE can be tailored to be used as a domain specific chatterbot, and also we looked into our FAQ bot architecture and some ways by which the accuracy of a domain specific bot can be improved, certainly if we can collect more data and broaden our knowledge base, the accuracy of the system can further improve and this FAQ bot could be used on University website to serve future students.

## 7.Reference

---

- [1] R.S. Russell, Language Use, Personality and True Conversational Interfaces. Project Report, AI and CS, University of Edinburgh,Edinburgh, 2002.
- [2]. Jia, "The Study of the Application of a Keywords-based Chatbot System on the Teaching of Foreign Languages," University of Augsburg, Augsburg, Germany 2002.
- [3]R. P. Schumaker and H. Chen, "Leveraging Question Answer Technology to Address Terrorism Inquiry," *Decision Support Systems*, vol. 43, pp. 1419-1430, 2007.
- [4] [http://www.chatbots.org/virtual\\_assistant/](http://www.chatbots.org/virtual_assistant/)
- [5] <http://www.ikea.com/us/en/customerservices/faq>
- [6] <http://www.alicebot.org/documentation/>
- [7] S. Han and Y. Kim, "Intelligent Dialogue System for Plane Euclidean Geometry Learning," presented at International Conference on Computers in Education, Seoul, Korea, 2001.
- [8]A. De Angeli, G.I. Johnson, L. Coventry, The unfriendly user: exploring social reactions to chatterbots, Proceedings of The International Conference on Affective Human Factors Design, Asean Academic Press, London, 2001
- [9] A. Flycht-Eriksson, A. Jönsson, Dialogue and domain knowledge management in dialogue systems, Proceedings of the First SIGdial Workshop on Discourse and Dialogue, 2000.
- [10] M. Awad and L. Khan, "Web Navigation Prediction Using Multiple Evidence Combination and Domain Knowledge," *IEEE Transactions on Systems, Man and Cybernetics, Part A*, vol. 37, pp. 1054-1062, 2007.
- [11] M. A. Pasca and S. M. Harabagiu, "High Performance Question/Answering," presented at Annual ACM Conference on Research and Development in Information Retrieval, New Orleans, LA, pp. 366-374, 2001.

- [12] J. Kupiec, "MURAX: A Robust Linguistic Approach for Question Answering Using an On-Line Encyclopedia," presented at ACM-SIGIR, Pittsburgh, PA, pp. 181-190, 1993.
- [13] E. M. Voorhees, "Overview of the TREC 2003 Question Answering Track," presented at Text REtrieval Conference, pp. 2003.
- [14] S. Potter, "A Survey of Knowledge Acquisition from Natural Language," in *TMA of Knowledge Acquisition from Natural Language*, vol. 2003. Edinburgh, 2001