

dBot: AI Based Conversational Agent

Gayathri Ravichandran¹, N. Maithili Reddy², Shriya G. S.³

Department of Computer Science and Engineering, M.S Ramaiah Institute of Technology, Bangalore, India

Abstract: *An Artificial Intelligence Chat Bot is a service that simulates intelligent conversation with humans via textual or auditory methods. This paper discusses how natural language processing helps computer applications hold and maintain human conversations. We examine in particular the design and implementation details of our newly proposed chat bot: the dBot- Desktop Bot.*

Keywords: Chat bot, Natural Language Processing, Artificial Intelligence, Desktop Bot, dBot

1. Introduction

An AI chat bot is a conversational program that initiates an intelligent conversation with a human user through textual and auditory methods. As the complexity of computer systems are increasing constantly with the number of applications, it is useful to have a personal digital assistant that eases the burden on the users. Machine speech humanization thus plays an important role. Since dialogue systems are more natural than graphic based interfaces, the recent past has witnessed more emphasis on the former. Machine interaction splits into many areas including Natural Language Processing (NLP), Artificial Intelligence, speech parsing, keyword identification, etc.

Chat bots fall into two broad categories- bots built on a predefined set of rules and bots built on an artificial Intelligence algorithm. The former is limited in functionality as it only produces a response when it identifies a known keyword or input. The latter, however, simulates a human conversation and continuously gets smarter through its interactions with the user. It has the ability to hold a conversation with a human.

In this paper, we present our recent development: “dBot”, or Desktop Bot. We will look into some related work with dBot and present the design features and system architecture. We will further explain the algorithm that serves as a backbone to our project and also compare it with existing bots in the market. Finally, we explain the scope and future developments of dBot.

2. Related Work

Chat bots have been implemented by many researchers in the past. Notably, there has been work on Natural Language Processing and domain specific knowledge extraction. We examined the following two chat bot systems:

[1] Johan Rahman, Implementation of ALICE chat bot as Domain Specific Knowledge Bot for BRAC U, Brac University- Thesis Paper

This paper discusses the implementation details of ALICE chatbot as a domain specific knowledge bot, that helps the students of BRAC University with admission and course

information. It works on Natural Language Processing and Pattern Matching. The chat bot architecture and the language model architecture are distinct. It utilizes Artificial Intelligence Mark-up Language.

[2] Sameera A. Abdul-Kader et-al [2015], Survey on Chatbot Design Techniques in Speech Conversation Systems, International Journal of Advanced Computer Science and Applications, Vol.6, No.7, 2015

The proposed chat bot used Natural Language Processing Toolkit [NLTk]. The fundamental design techniques discussed were Parsing, Pattern Matching, Chat Script and Language tricks. The paper also discussed voice recognition and conversion, text processing and response generation.

3. Design

3.1 System Flow Architecture

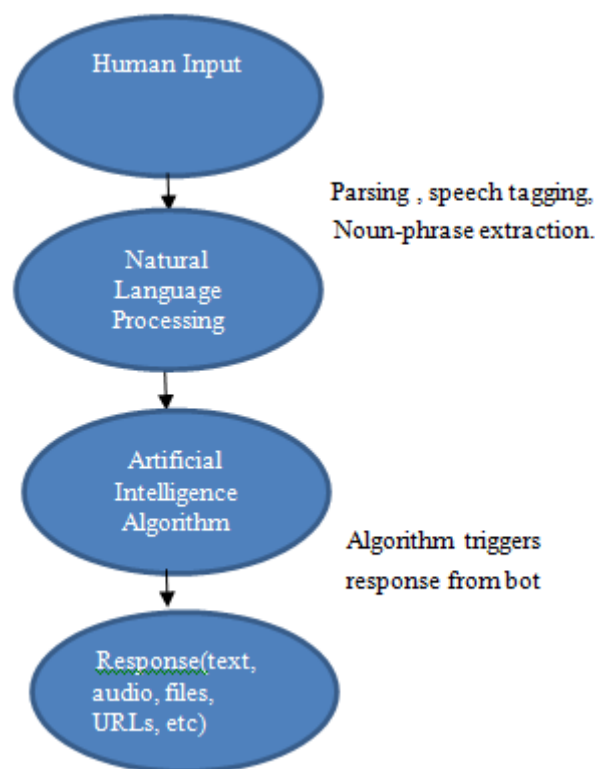


Figure 1: System Flow Architecture.

3.2 Software Specifications

The textual input is processed using TextBlob . TextBlob is a Python library for processing textual data. It provides a simple application programming interface (API) for common natural language processing (NLP) tasks such as part of speech tagging, noun-phrase extraction, sentiment analysis, classification and more.

The following are the software requirement specifications:

- a) Tkinter: Used to develop user Interface.
- b) TextBlob: Performs NLP tasks such as part of speech tagging and noun-phrase extraction.
- c) eSpeak: A compact open source software speech synthesizer for English.
- d) Google API

4. Implementation

4.1 Algorithm

Let 'S' denote the system.

- (1) Initially S= {..}
- (2) Let the system contain 'M' modules.

S = {M,..}

M= { K,F,C,O}

where, K <- Keywords/ Textual Input

F <- Files; local files in the user's system.

C <- Numeric input

O <- Objects

- (3) The user input can take the following forms :-

- (i) Keywords <- { pronoun, noun, verb, adjective, greeting_keyword, new_phrase} where new_phrase is any keyword that the chat bot has not encountered so far.
- (ii) Files <- { File name, File location}
- (iii) C <- { operators, operands}
- (iv) Objects <- { search queries}

If input is of type(i)

goto (4)

else if input is of type(ii)

goto (5)

else if input is of type (iii)

goto(6)

else

goto (7)

- (4) Return an appropriate audio response.

if (greeting_keyword)

return greeting_response

if (pronoun=="you" || pronoun=="u")

return phrase with "me"||"myself"||"I"

else if (pronoun=="I" || "myself"|| "I")

return phrase with "You"|| "u"

if(new_phrase)

invoke TextBlob library to provide intelligent response.

- (5) Open the file from the desired location and return its contents.

- (6) Open calculator application. Perform the required operation on the operands and return the result

- (7) Return links related to the search query.
- (8) Repeat till user continues to provide input.

5. Testing and Comparison

5.1 Functional Requirements 1.1 – Textual Input

Table 1: Test case for greeting message

Input	Greeting messages from user.
Algorithms and Libraries used	Natural Language Processing, TextBlob, eSpeak
Expected Output	Audio greeting Response

Table 2: Test case for specific keyword

Input	Message with keywords-"you" or "u"
Algorithm and Libraries used	Natural Language Processing, TextBlob, eSpeak
Expected Output	Audio and text response with words "I", "me", "Myself"

Table 3: Test case for specific keyword

Input	Message with keywords-"I" or "me" or "myself"
Algorithm and Libraries used	Natural Language Processing, TextBlob, eSpeak
Expected Output	Audio and text response with words "you", "u".

5.2 Functional Requirements- Arithmetic Operations

Table 4: Test case for numbers in input

Input	Operators(+, -, *, /) and operands (numerals)
Module Used:	OS Module
Expected Output	Returns result of operation.

5.3 Functional Requirements 1.3 – Files and Applications

Table 5: Test case for file operations

Input	File name and location in local host
Algorithm and Library Used:	Natural Language Processing, TextBlob
Expected Output	Opens file and returns the contents

5.4 Functional Requirements 1.4 – Objects

Table 6: Test case for search queries

Input	Search Query
Library used:	Google API
Expected Output	Returns related URL.

5.5 Comparison with existing applications

Most of the existing bot applications are for mobile environments. We propose an application that runs on desktop applications. Many chat bots available in the market run on a remote server, whereas the proposed system runs on a local host, thus providing easy access to various files, folders and applications in the user's computer.

6. Results

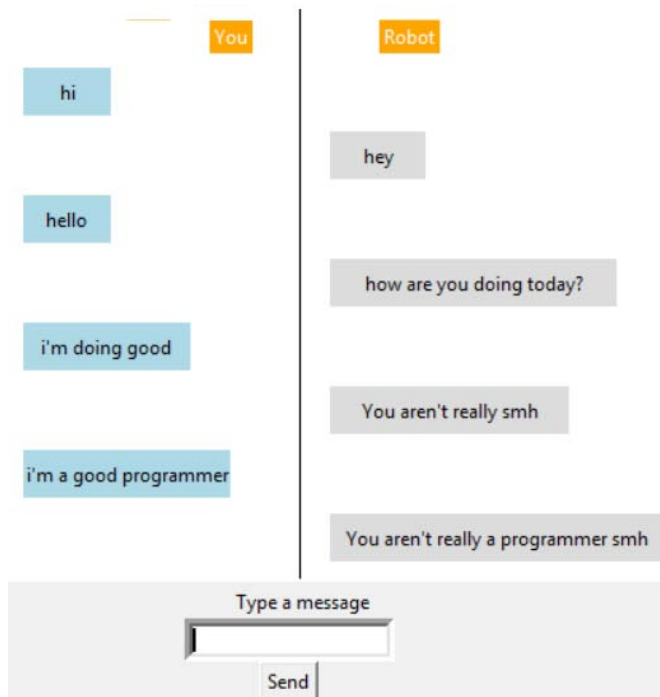


Figure 2: Snapshot of an ongoing Conversation

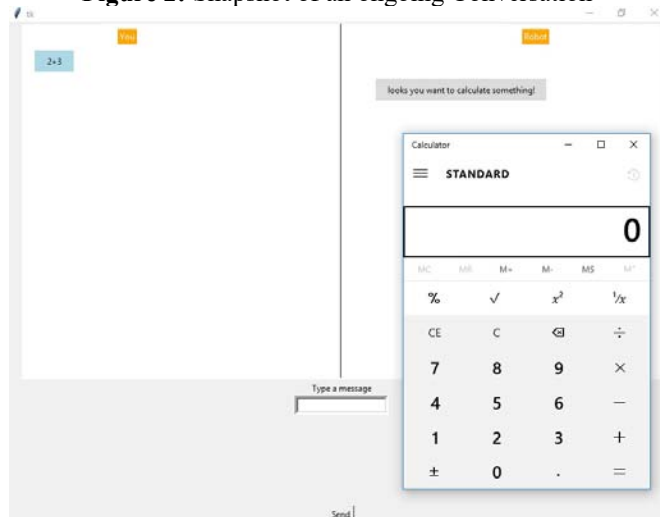


Figure 3: Illustration of Opening of Applications in the user's local system

7. Conclusion

We have thus examined the design and implementation specifications of our new chat bot: the dBot. Chat bots use Natural Language Processing that helps them understand user's queries better. As the complexity of applications are increasing, chat bots serve as useful mediators and assistants by reducing human effort.

8. Scope and Future Work

Our desktop bot can be extended to perform any of the following tasks:

1) Online business or e-commerce chat bots that can serve as personal assistants to customers and help guiding them

with their queries regarding the company's products, services, etc.

2) Chat bots for food and grocery delivery.

3) Chat bots that schedules meetings and manages timetables.

4) Chat bots programmed to provide banking solutions and financial advice.

References

- [1] Sameera A. Abdul-Kader and Dr. John Woods, Survey on ChatBot Design techniques in speech conversation systems, International Journal of Advanced Computer Science and Applications, Vol.6, No.7, 2015
- [2] Om Komawar et al. An Internet Relay Chat Bot using AIML, International Journal of Science and Research, Volume 4 Issue 10, October 2015
- [3] Jizhou Huang, Ming Zhou, Dan Yang, Extracting Chatbot Knowledge from Online Discussion Forums, International Joint Conference on Artificial Intelligence
- [4] K.M. Colby, Simulation of Belief Systems, Computer Models of Thought and Processing, 1973.
- [5] J. Weizenbaum, ELIZA- A Computer Program for the Study of Natural Language Communication between Man and Machine. Communications of the ICM, 1966
- [6] Jia, The study of the applications of a Keyword based Chatbot System on the teaching of Foreign Languages, University of Augsburg, 2002
- [7] S.Potter, A Survey of Knowledge Acquisition from Natural Language, vol.2003, Edinburgh, 2001
- [8] Johan Rahman, Implementation of ALICE chatbot as domain specific knowledge bot from BRAC U, Thesis Paper, BRAC University
- [9] D. Vrajitoru, Evolutionary sentence building for chatterbots, pp.315-321, 2003.
- [10] P.Hingston, A Turing Test for computer game bots, IEEE Transactions on vol. 1, no.3, 2009