Geek-AI-Mania‘18 Competition

[Mailam Engineering College]

[DIVIYA K] – [Team Leader]

[ABARNA A] – [Team Member]

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Document Revision | Date | Comments |
| DIVIYA K |  | 05/10/2018 | Chatbot |
| ABARNA A |  | 05/10/2018 | Chatbot |
|  |  |  |  |

Contents

1.Abstract 4

2.Problem Statement 4

3.Background Search 5

4.Approach to the Solution 5

5.Solution Description

6.Technology Stack & Architecture (Software and Hardware)

7.Experimentations & Results

8.Working Model & Accuracy Test Result

9.Future Scope of Solution

10.Conclusion

11.Appendix (as applicable)

# Abstract

There are lot of treatments that are available for various diseases. No human can possibly know about all the medicines and the diseases. So, the problem is that there isn’t any place where anyone can have the details of the diseases or the medicines. What if there is a place where you can find your health problem just by entering symptoms or just scanning an ECG or you can check whether the prescribed medicine is supposed to be used the way you are told to. Then it will help us to deduce the problem and to verify the solution. The proposed idea is to create a system with artificial intelligence that can meet the requirements. The AI can predict the diseases based on the symptoms and give the list of available treatments. The System can also give the composition of the medicines and their prescribed uses. It helps them to take the correct treatment. Hence the people can have an idea about their health and can have the right protection.

# Problem Statement

Chabot, is an Artificial Intelligence (AI) feature that can be embedded and used through any major messaging applications. Companies are looking for a way to efficiently connect with customers on a more human level, and chatbots been advanced into new level that answer customer queries from the documents or FAQ. Our Aim is to create intelligent chatbot which can answer user query by looking over the documents. Requirement is to create Automatic factual creation of question and answer knowledge based from Documents which can be feed to Question and Answer bot. User interface should be there to upload new documents (pdf/doc/txt) and model should update knowledge base for the same. Model should be flexible to handle domain specific documents (healthcare/manufacturing/retail) as well. Currently Documents consists of 2225 documents from the BBC news website corresponding to stories in five topical areas from 2004-2005. Class Labels: 5 (business, entertainment, politics, sport, tech). Identify entity and entity Linking detection and create of Knowledge Graph from the corpus. The team is also free to choose a mode for the solution demonstration. A sample proposed set of steps for demonstration.  We would like to understand in detail the participants’ approach to the problem. Hence we would like to know the technology in use, use of any language/library for algorithms you may develop, how you code has been developed, test/experiment results and any other details you may provide. We are expecting to see working model during the live demo by uploading new documents model should generate Question and answer for those documents.

# Background Search

*Build a simple and interactive real time chat system.*

# Approach to the Solution



# Solution Description

Pseudocode: med\_disc (business)

{

Read the message.

Identify type of message.

If bbc news, then

Search for json object.

If found, then

Print values.

Else

Search online for result and load it to json.

Print values.

Else

Print suggestions.

}

------------------------------------------------------------------------

Pseudocode: chat\_bot (business)

{

Read message.

Identify keywords.

Search for matching conversations in previous chats.

If found, then

Print reply.

Else

Search for matching keywords.

Call function choose\_chat (business)

}

Pseudocode: chat\_bot (entertainment)

{

Read message.

Identify keywords.

Search for matching conversations in previous chats.

If found, then

Print reply.

Else

Search for matching keywords.

Call function choose\_chat (entertainment)

}

Pseudocode: chat\_bot (politics)

{

Read message.

Identify keywords.

Search for matching conversations in previous chats.

If found, then

Print reply.

Else

Search for matching keywords.

Call function choose\_chat (politics)

}

Pseudocode: chat\_bot (sports)

{

Read message.

Identify keywords.

Search for matching conversations in previous chats.

If found, then

Print reply.

Else

Search for matching keywords.

Call function choose\_chat (sports)

}

Pseudocode: chat\_bot (tech)

{

Read message.

Identify keywords.

Search for matching conversations in previous chats.

If found, then

Print reply.

Else

Search for matching keywords.

Call function choose\_chat (tech)

}

# Technology Stack & Architecture (Software and Hardware)

This is the actual knowledge and brain of the chatbot. This is an organized set of large number of input patterns and their corresponding replies covering varied types of situations that may arise during a conversation. Obviously, a vast range of questions has to be included in the database of an efficient chatbot for providing accurate and error-free experience. There may be more than one answer to the same question to avoid redundancy of reply as well as one reply to more than one question to avoid unnecessary lines of code. While designing the database, we had to take care regarding the following points. Ambiguity handling for those questions for which the chatbot does not have pin-point answers to. A “default” case is maintained, which gives generic answers called “pickup lines” when no pattern match is available. These lines fit into wide range of questions and hence, user is satisfied even for a difficult question. File handling can be used to store key data about the user for future purposes.

# Experimentations & Results



# Working Model & Accuracy Test Result



# Future Scope of Solution

# Conclusion

There are a lot of intelligent chatbots in the world. Many have won the Loebner prize and are really clever programs. But even after all these advance scripts, the core problems with conversational entities (and other AI systems) remain more or less identical. None of such systems have the power to learn, or to hold memories of incidents for longer time. They lack creativity, one of the most important aspects of intelligence. Today’s AI systems just keep following a fixed set of instructions. They do not actually show intelligence and don’t understand what is happening around. The most popular methodology is just searching and finding matches from a database, called the rule method [7]. But that is not intelligence. To be more concrete, this is to be noted that chatbot programs still lack at a 0.5 score of Performance Factor according to the Theory of Machine Intelligence [5]. So do other such systems. Hence there is still a vast need and scope for future improvements in this niche.

# Appendix (as applicable)

[This section include references and standards]