Journey to the Realm of Chatbots

Aashutosh Kumar Jha

B.E. Student, Department of Computer Science and Engineering, SSIT, Tumkur, India

Abstract: **The growth in the field of Artificial intelligence is at its peak, so does its sub-domains. One of the most influential and popular sub-domain of artificial intelligence is "Human-computer Interaction". Human-computer interaction is done with the help of "conversational agents" or "Chatbots", and it is impacting human life on a daily basis in almost every aspect, at a larger scale at present time compared to a decade ago. There is a huge demand of chatbots in the market today from all domains of work, it may be business, education or medical. Present era is considered as the era of chatbots, so here in this survey paper we will see the technology shift in the field of "evolution of human-computer interaction" from its origin to current models. We will also take a look at how and why a specific chatbot or conversational agent should be considered at a given moment of time and not the other one, along with a high level approach toward building a "Rule-based" chatbot.**

Keywords: **chatbot, Turing test, Natural language understanding, Parsing, Rule-based**

1. Introduction

Chatbots are gaining exponential popularity, so there is no doubt that the year 2019-2020 is going to be the year of chatbots. Chatbot is considered as an artificial intelligent programs used to make conversations between human and computers. This AI technology at the present time is putting a huge impact on the way human and computers have ever conversated, all over the world. So, the first and major question comes in mind is that, how this technology has made a huge impact, what are the major reasons for its success. The previous question can be followed by several other, such as is this a new technology or how it has evolved over the time?, where this technology can be used precisely?, what are the forms or types of chatbot are available today? , how and why someone should prefer one of its type over other?, and what can be the approach toward building one of its model?. The goal of this paper is to answer all the above questions. First we will see origin of this technology, then a list of most popular chatbots along with technology used in them . Further we will look a bit deeper at the major technologies that can be used for making a chatbot. This is followed by domain analysis and architecture of various types of chatbots. Next, we will see what can be a possible approach toward building a simple chatbot using python language and its libraries.

The Cause:

Chatbots or the conversational-agents which are considered to be as a successful model , always proved its capabilities and efficiency at "Loebner Prize Competition". The Loebner prize competition is an annual competition for chatbots, where chatbots are tested in their capabilities by passing Turing test and also it is known to be the first formal instantiation of "Turing test".
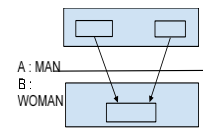

Fig. 1. Turing test model

Question is, how Turing test serve as a basis for analyzing the capabilities of a chatbot? The answer can be understood by understanding the model of Turing test or "The Imitation game": Turing test consist of three players A: a man, B: a woman, C:an interrogator. The first requirement is to put interrogator into a seperate room and man, woman in seperate room. Communication will be done in a written manner or preferably by a typewriter.

Let's see the part played by each character:

- Role of Interrogator('C'): Interrogator has to ask interrogate A(man), without knowing that he is interrogating a man, in a written manner and has to identify that who is being interrogated a man or a woman.
- Role of man ('A'): man has to provide false answer about his identity to confuse the interrogator.
- Role of a woman ('B'): woman has to help the interrogator by providing true answers.

Next, "What will happen when a machine takes the place of A?", will the interrogator be able to decide that he/she is interrogating a machine or a woman. Clearly this imitation game model indicate that term "machine" and "think" are not in generalized form. The boundary of "thinking machine" for imitation game indicated toward a particular machine," Digital Computer". The further exploration of digital computer indicated that, digital computer has special ability to mimic any discrete state machine and were described by saying them as "Universal machine".

This scenario clearly provides the foundation of first ever human-computer interaction(theoretically), and using this test model most of the chatbots are evaluated in term of their capabilities by passing this test.

2. Chatbot evolution and technology involved

Chatbots are quite big in number present in world today, in

International Journal of Research in Engineering, Science and Management
Volume-2, Issue-4, April-2019
www.ijresm.com | ISSN (Online): 2581-5792

46

order to analyze the evolution of chatbots over the time we will look some of the most popular chatbots of all time along with the techniques used in them from basic model to present models.

ELIZA [1966]: The first conversational agent, developed by J. Weizenbaum at MIT. It was written with MAD-SLIP script, and designed in such a manner so that its ability could be improved by its user. It was capable to process natural-language with general syntax and punctuations. A double carriage return in input statement was used to transfer control from user to ELIZA.It made its response on the basis of keywords present in input, which was also a challenge for it.

SHRDLU [1970]: Natural language understanding program developed by T. Winograd at MIT. It was made with LISP programming with micro-planner. It was used for describing object and its location and had a memory supply too. It was capable to remember names of object and searching back through interactions to find correct context.

PARRY [1972]: A similar program like ELIZA, developed by K.colby. It was capable to implement a crude model of behaviour of a person with schizophrenia. This model of conversational agent is considered one of the very first model in medical implementation of conversational agent.

LADDER [1978]: This was one of its kind, a program with deductive algorithm and semantics rules with subset of English language. It had a sufficient amount of vocabulary and also capable to learn new concepts and can apply its learning in a problem-solving situation using effective interaction.

JABBERWACKY [1982]: A conversational agent aimed to simulate human computer interaction in natural language by moving a text based system to voice operated system. Developed by Rollo carpenter, its evolution leads to another agent "cleverbot" and here for the first time the word "chatterbot" used. Jabberwacky used a contextual pattern matching technique to produce response and was capable to take take sensory input such as sound.

ALBERT ONE [1998]: A chatterbot with multifaceted approach in natural language to mimic human conversation. Developed by R. Garner, and used multiple response method for output. System was composed with several subsystem working in a hierarchical approach and its controller had built-in- stimulus response database of human statements.

A.L.I.C.E [2001]: Artificial Linguistic Internet Computer Entity a natural language processing chatbot, developed by R.wallace.Storage of conversation pattern was done in artificial intelligence markup language(AIML) which used "topic"," category" object to store data on a specific topic of a particular category. AIML interpreter was used with pattern matching technique to generate response.

WATSON [2006]: WATSON is considered as "cognitive system" with the capabilities of natural language processing, hypothesis generation and dynamic learning. Developed by IBM and it uses Deep Question-Answering technique to answer a question. At present watson is capable for making a reverse inference that is generating a question from answer.

SIRI [2011]: Developed by Apple, it uses natural language processing algorithm and training of machine learning model on large dataset for speech recognition in order to generate a response. First it uses a discretization algorithm to turn voice into digital data then uses Dynamic Time Warping (DTW) to understand the input.

ALEXA [2015]: A cloud-based chatbot by Amazon with natural language understanding capabilities. It uses a shortlisting-re ranking approach for understanding input. It has a neural model with an efficient stored encoder and skill-specific classification technique for its responses. The voice of alexa is generated by a long-short term memory artificial neural network. As we have now seen some of the most popular chatbots we should have a look at some core techniques used in making a chatbot:

One or more of the following technologies can be used for building a chatbot model:

- *Natural language understanding*: This a component of natural language processing and used for human-computer interaction. This technique helps chatbot to understand user natural language input without a formal syntax of computer language and then return the response back to user.
- *Speech recognition:* This technique enables a chatbot to recognize audio input and then convert this input to text form in order to understand the context of input. The major models which uses this technique can be a "speaker dependent" or "speaker independent" model.
- *Pattern matching:* This is an all-time hit technique in chatbot universe, where it tries to find a match for given sequence of tokens. The sequential patterns are generally represented using regular expression. Chatbot uses a programming language such as LISP for pattern matching and finding a required keyword in order to generate output.
- *Parsing:* This technique is used to analyze the natural language user input in the form of syntax analysis of input. This technique uses rules of formal grammar.
- *Semantic-net:* In order to represent the knowledge of dataset in the graphical manner with a pattern of interconnected nodes this technique is used. The major reason for its validation for chatbots is that it is more natural than a logical representation of dataset and also permits the effective use of inference algorithm to find correct response for input statement.

Other technologies that are widely used in the modern systems are -Deep learning, Machine learning, markov-chain model along with languages like AIML, ChatScript, Rive Script, Python, Java script etc.

3. Domain and architecture

Next question is, what are the major domains or areas according to which we can approach toward building a

International Journal of Research in Engineering, Science and Management
Volume-2, Issue-4, April-2019
www.ijresm.com | ISSN (Online): 2581-5792

47

chatbot?. The answer depends on the length of conversation and scope of conversation. If the length of conversation is simple, specific to the context and short, then it is a closed domain chatbot otherwise an open domain. Similarly, if a chatbot is using a predefined set of responses it is called Rule-based chatbot or retrieval-based model, otherwise a generative based model where a chatbot will use machine learning algorithms to generate a response from scratch for the unknown dataset. Hence the possible types of chatbot can be built according to the combination of each of above (closed, open, retrieval-based, generative based) concepts. This can be represented by the following figure:
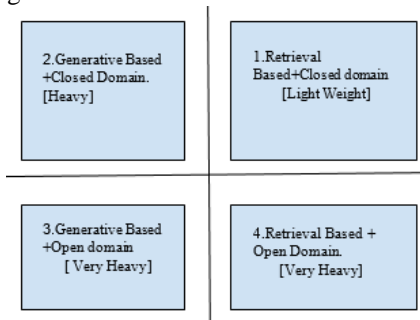

Fig. 2. Chatbot domain

- A retrieval based closed domain chatbot has a predefined set of responses or a corpus and conversation is within the boundaries of context of chatbot. This model is easier to achieve as compared to other counterparts.
- If in a closed domain chatbot there is no predefined responses then chatbot has to generate responses by using machine learning algorithms, and a resulting chatbot is a retrieval based open domain chatbot.
- Further, when the conversation of the chatbot is in closed domain and also if it is generating response by its own, it falls under the category of general ai chatbots.
- The most difficult type of chatbot is an open domain generative based chatbot where the conversation scope and length along with responses are not defined. It requires a very huge amount of data to train these models and output responses may vary also in this case.

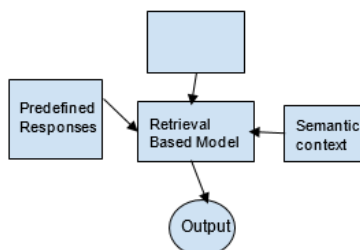How Retrieval based architecture works?

Fig. 3. Retrieval-based architecture

Chatbots here will have a predefined set of responses and it works as follows: A predefined dataset is used to train the model and when input message is passed to chatbot here, with the help of semantic and syntax analysis it selects the most appropriate response and return it back as output response.

How Generative based architecture works?

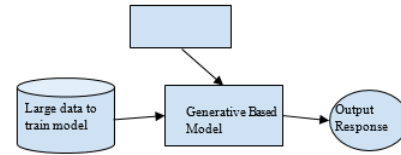Chatbot here will not have any predefined set of responses and it works as follows:

Fig. 4. Generative-based architecture

First, this model is trained with a large amount of data and no knowledge base is predefined in order to avoid the selection of a particular response which is already present in advance because here goal is to generate a response for input regarding which the chatbot model is not trained. when user provides an input to chatbot then with the help of machine learning algorithm it generates a response according to the data which is used to train it and then finally chatbot returns an output response. This type of model plays an important role where questions are asked on unknown data or out of the scope of chatbot.

4. Rule based chatbot

At this stage we have analyzed major chatbot technologies, domain in which a chatbot may fall under and architecture of chatbots, now here i will present an approach to make a simple rule-based chatbot. Rule based chatbot or a retrieval based chatbot model will require a predefined set of responses (or a dataset or a corpus), and use of a programming language. The options of programming language are developer dependent, for this high level-approach of making a chatbot we will use Python on a windows platform. Following are the step by step approach toward making of a rule based chatbot: We will call our bot "shazz", and here is the list of requirements for this chatbot
Operating system: Windows 10,
 python packages: NLTK, pip, and
 an editor (example:atom),
Python NLP engine: Chatterbot.

- *Step 1:* Make a file with extension "yml" for a set of responses using editor. This file will serve as a knowledge base for our bot. Yml is a data serialization language and the responses can be made as follows:
  (question):
  - - hello bot,
  (response):
   -hello man.
- *Step 2:* write a python program by importing two chatterbot engine packages: chatbot and listtrainer, and provide a name for your bot in this function:

International Journal of Research in Engineering, Science and Management
Volume-2, Issue-4, April-2019
www.ijresm.com | ISSN (Online): 2581-5792

48

chatbot('shazz') .
-->import chatbot
-->import list_trainer
-->Bot = chatbot('shazz')

- *Step 3:* Provide a path for your yml file to the python file using open function in python in function listdir():
  -->listdir('path of yml file ').
- *Step 4:* pass data file to train('filename'), for training of your bot as "train(data)".
- *Step 5:* open command prompt and navigate to the folder where you made python file and run" python filename.py ", and training of your bot will start and after completion of training you can conversate with your bot.

Now here are some important questions that may come in your mind:

Why to use python chatterbot engine?

I have used chatterbot engine because it support multiple natural languages and once trained there is no need to train over the same dataset.

Next, how training of chatbot takes place?

Chatterbot engine uses built-in adapters to train data and its storage adapter helps to remember the data over which it is trained.

Further, how it select a response?

chatterbot engine use a graph data structure and a confidence value in order to find out the most efficient response for user input. And Logic adapters of chatterbot helps to select a particular response for a corresponding input statement. Further chatterbot uses a Jaccard's algorithm (or a similar one is also valid) which first tokenize the input and then selects a most suitable output for an input.

5. Application and advantages

Major applications of chatbot:
- e-commerce websites.
- Online banking websites.
- Restaurant bots.
- Smart device personal assistant.
- Online educational websites.

A. Advantages of using chatbots

- Faster query processing
- After hour services
- Handles more queries at a time than a human
- Helps in retaining more span of a person over a website and,
- More information access without too much physical effort.

6. Conclusion and future work

In this paper our major focus was to illustrate technology shift in chatbots along with domain and architecture of various types of chatbots followed by a method to make a simple rule based chatbot using python language. I have also represented a Turing test model which is the base for evaluating the capabilities and efficiency of chatbots. The rule-based chatbot "shazz" which is presented above is a simple chatbot design but with good capabilities of remembering the trained data and multiple language support. Its working capacity can be enhanced by adding "speech -recognition" which is supported by chatterbot engine. Further a confidence value can be set by designer (example 0.7) after which bot select a response with accuracy probability of given confidence value X 100.

References

[1] A. Turing "computing machinery and Intelligence", vol. 59, no. 236, October 1950.
[2] J. Weizenbaum "Eliza-A computer program for the study of natural language communication between man and machine," Mass Communications of the ACM, Volume 9, Number 1, pp. 36-35, January 1966.
[3] Rob high "The era of cognitive system a look at ibm watson" IBM corporation, Redbook 2012
[4] Wallace, R.S., 2009. The anatomy of ALICE. Parsing the Turing Test, pp.181-210.
[5] F. Catania, M. Spitale, D Fisicaro, F. Garzotto - 2019 - ceur-ws.org
[6] Workshops of the International Conference on Advanced Information Networking and Applications
[7] Waina, Web, Artificial Intelligence and Network Applications, pp. 946-956, 2019.