Proceedings of the Fourth International Conference on Smart Systems and Inventive Technology (ICSSIT-2022)
IEEE Xplore Part Number: CFP22P17-ART; ISBN: 978-1-6654-0118-0

Design and Implementation of Multipurpose Chatbot

Pavitha N
Dept. of Artificial Intelligence and Data Science
Vishwakarma Institute of Technology
Pune, Maharashtra, India
pavitha.n@vit.edu

Priyanka Bhatele
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, Maharashtra, India
priyanka.bhatele@vit.edu

Sharmishta Desai
School of Computer Engineering and Technology MIT World Peace University, Pune, Maharashtra
sharmishta.desai@mitwpu.edu.in

Himangi Pande
School of Computer Engineering and Technology MIT World Peace University, Pune, Maharashtra
himangi.pande@mitwpu.edu.in

2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT) | 978-1-6654-0118-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICSSIT53264.2022.9716306

***Abstract*— Chatbots, also called chatterbots are applications that simulate human conversation either taking voice or text commands or both as input. They are automated programs that utilize Natural language processing techniques to understand process and responds accordingly to the users. Chatbots are cost effective and require a very little investment cost in the beginning stage of development. The software program can be built using AIML that use Extensible Mark-up Language as a base. In python, libraries like NLTK and JSON package can be used for the efficient designing of a Chatbot model. Chatbots use Corpus which is the vocabulary or the huge amount of data from different human dataset to the model. This paper elaborates on a Chatbot model that uses the functions of nltk library in python to pre-process data and responds accurately to the user using the cosine similarity approach. This chatbot can be implemented for any specific domain by changing the corpus. The corpus used in this design is about chatbots itself. The chatbot is capable of accurately responding to the queries related to chatbots. Accuracy of the proposed chatbot is 93% that is out of 150 queries it answered 140 queries correctly.**

Keywords— Chatbots, Cosine Similarity, Stemming and Tokenization, Lemmatization, TF-IDF

I. INTRODUCTION

Chatbot is a software program that interacts to human imitating itself to be a human as much as possible. Capable of making a conversation via text or speech, it is a computer program which is a subset of Artificial Intelligence with Natural Language Processing. The software application has been designed to simulate convincingly in a manner as a real-world conversational agent. Building of such programs requires continuous tuning and has to achieve certain defined metrics in a quantifiable measure for evaluation purpose [17].

Almost 80% of all the service providing companies, having employee strength of more than 500, have deployed Chatbots. Most of them are capable of handling first level queries from the customers including gathering their insights. Being application software, it can provide an availability of 24*7 for the users. It has not only provided better customer satisfaction but also improved the sales structure by 67% with the increase in customer engagements [4]. Despite this, it has also saved on the expenses part of the company by at least replacing the place of level 1 layer of human support teams.

Chatbots can be classified as task-oriented bots and domain based chatbots [2]. Task oriented bots are usually trained on some specific actions. On the other hand, domain-based bots are designed to make any conversations on behalf of humans. Some of the very famous examples are Siri application by Apple, Alexa by Amazon, Cortana by Microsoft and Google assistant by Google.

This paper proposes to implement a Chatbot for any domain by training it with the corpus containing relevant data about the field. Corpus is that plethora of data that is used to train the machine and make it understand about human language [18]. The concept of cosine similarity, using python, has been used to generate response which enables the system to answer the queries by matching the significant keywords within it. Latent Dirichlet Allocation (LDA) and Latent Semantic Analysis (LSA) are some other algorithms that are used to check the similarity between the words. Cosine similarity

approach is found to be 93% accurate as reported in [11,12,14]. When the chatbot is queried, it matches the words in the query with the corpus using cosine similarity. The chatbot answers the user query accurately as per the algorithm. The proposed implementation of Chatbot does not pertain to any specific field. Cosine Similarity is the mathematical formulation that matches a query with corpus and hence corpus changed will enable the chatbot to work for any field.

Section 2 of the paper indicates on the literature studied about the related work. Section 3 details the methodology used to implement the design model. It covers the pre-processing, keyword matching and the approach used to generate the responses. Section 4 discusses the results and the output examples of the Chatbot application. Section 5 concludes the paper with the future work that can be done in order to improvise the design. Section 6 quotes the references used.

II. LITERATURE REVIEW

Chatbots can be implemented in python using nltk (Natural Language Toolkit) library [3,4, 5]. It is one of the foremost used libraries in python that is used to accomplish pre-processing tasks like word tokenization, sentence tokenization, removal of stop words, stemming, entity recognition, PoS Tagging [3,14]. Automatic text summarization, Translation, named entity recognition, relationship extraction, and sentiment analysis are some of the vital tasks that can be done using NLTK [3,4]. Riverscript is a plain text, line-based scripting language used mainly for the building of chatbot application. It's an open-source language that can be interoperated with Java, JavaScript, Perl and Python language. [14]

Punkt tokenizer is the pre trained data package in nltk library that is used for word tokenization and sentence tokenization. There is a lexical database named WordNet that can be used as dictionary for English Language designed for NLP [19].

Python package JSON also has certain tags defined that can be used to query data set to be parsed by python code. These tags have set of patterns and their responses. Graphical User Interface can be developed using Tkinter package in python. It is a standard interface for GUI creations [4].

Chatbot can also be designed using Artificial Intelligence Markup Language (AIML). The base to derive it is Extensible Markup Language (XML). It has tags or units named as topic and categories [1,5,14]. These categories have a predefined pattern that is used as a basic unit of knowledge input to the model. Bots respond accurately using the templates defined within the matching pattern of categories [1].

AIML flies can also be used to train the Bot models using question-answer pairs. Files also have a defined pattern which when matched to the input query is used to return the response [5].

With the purpose of the evaluation and assessment of the performance of Chatbots quantifiably, certain metrics has been defined. Some metrics that help in effective Chatbot designing are Scalability, Interoperability and Speed. Any bot application passing test like Turing Test and BLEU Score (BiLingual Evaluation Understudy) with high score is considered to be an intelligent machine [2].

MedBot was created for the medical field that assists patients on the basis of the symptoms they have and help them to diagnose what medical illness they may have. The architecture created a query tree from the sentences input by the user. After parsing the tree cosine similarity can be used to output the results. [6]

Chatbots can be based on Speech Recognition where the user provides audio input. Automatic Speech Recognition (ASR) is used mostly for the conversion of the speech to text. Which in turn can be used for further conversation. Some of the techniques like Chat Scripts, Markov Chain, SQL and Relational Database and Language Trick can also be used to design chatbots.[7]

Based on goals, bot application can be classified as task oriented and non-task oriented. They can also be classified based on the approaches applied as rule-based, retrieval-based and generative based approaches. [13,14]

Chat.io is a chatbot that is integrated inside an already existing system to carry over the business and answer the queries related to it. Such systems are used in Facebook Messenger to interact with the Facebook users to help interact with the admin. Collect.chat is a chatbot that operates based on the widget interaction to get the user enquiries. Cleverbot learns from the human interaction process.[8]

Uchaguzhi, a chatbot was designed for the crises that happened at the time of Kenyan elections in 2007. It was deployed within Facebook Messenger for the users to report about the Polling Booths experiencing violence or misconduct. The system was also used to understand and study the human behaviour when they are interacting with a bot application. Mostly 50% of the chats ended after the greeting message from the chatbot.[9]

SARANG Bot was designed and tested by Turing test and found to be more than 80% accurate. The bot application was based on AIML technique.[10] Chatbots for the security training purpose were studied and the positive effect it had is off the knowledge base created by the bot applications and be used for future development of the organization. [15].

A centralized grievance redressal application was developed by Viral Patel et. al. for India. The application can register complaints related to daily life problems within the country. Application was developed using Java and XML. Users can online check the status of the complaints using the complaint id. One of the vital functions of the application is to provide an estimate about the material and time required to repair the pothole on the basis of the picture uploaded with the complaint by the user. The estimation algorithm was implemented in Matlab.[21]

Naive Bayes and entropy classification algorithms were implemented to recognize whether a chat application is enabled using a chatbot or a human. Naive bayes were concluded to be faster than the entropy classifier.[22]

This study proposes a chatbot design that inherits functionalities of NLTK library for pre- processing the text. It uses cosine similarity approach in python to generate the response.

Paper Name	Authors	Research Gap
Chatbot in Python	Akshay Kumar, Pankaj Kumar Meena, Debiprasanna Panda, Ms. Sangeetha	Chatbot program is limited to answer some specific fields
Human Chatbot Interaction Using NLTK	N.Naveenkumar, M.Hemanth Reddy, S.Sai Nikitha, T.SaiRam Reddy	Work has majorly foccussed on the pre-processing parts.
Development of Chatterbot using Python	P. Shanmugapriya, Harini M	Limited dataset due tagging done for queries.
Implementation of a Chat Bot System using AI and NLP	Lalwani, Tarun & Bhalotia, Shashank & Pal, Ashish & Bisen, Shreya & Rathod, Vasundhara	The question-answer pairs have implemented. The responses are generated based on the pattern matching
Chatbot for Medical Treatment using NLTK Lib	Kalla, Dinesh & Samiuddin, Vatsalya	Chatbot program is limited to medical field.
Designing Chatbots for Crises: A Case Study Contrasting Potential and Reality	Piccolo, Lara & Roberts, Shadrock & Iosif, Anna & Alani, Harith	Chatbot program is designed for a specific field

Fig 1: Summarized Literature Survey

III. METHODOLOGY AND IMPLEMENTATION

The paper uses Natural Language Toolkit (NLTK) library for implementation by using opensource Jupyter platform. The first step is to download and install nltk for this purpose pip command can be used. Once the installation is done nltk can be imported and can be used for the subsequent steps. Next step is to read the corpus, we have stored corpus as a text file. The principal issue with text data is arrangement of the content. In any case, the Machine learning calculations need a type of mathematical element vector to play out the undertaking.

Fig. 2 shows the overall architecture of the proposed model. The chatbot start with the User Greetings and a message. The query from the user is then pre-processed and matched using the cosine similarity algorithm. Chatbot finds the accurate answer and provides the same to the user. The query-answer can be looped until the user wants to continue the enquiry. Once done chatbot greets and ends the conversation.

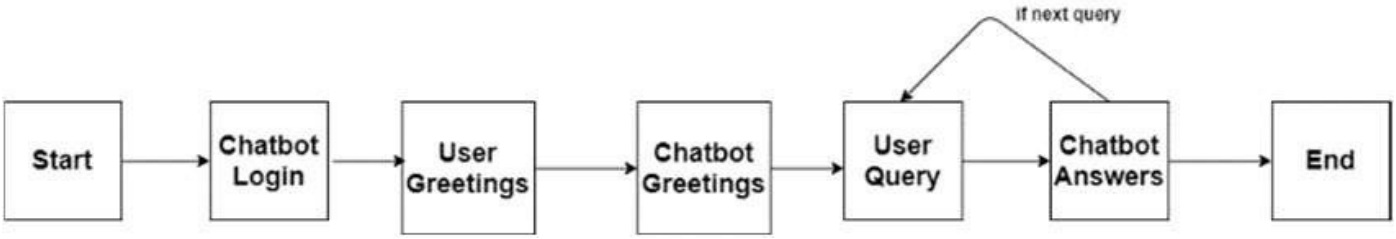

Fig 2: Overall Architecture

NLP projects requires to pre-measure it, to make it ideal for working.

A. Pre-processing

Various pre-processing operations performed are

(1) Convert text to lower or uppercase: Changing over the whole content into single case, with the goal that the calculation doesn't treat similar words in various cases as different.

(2) Tokenization: Sentence tokenizer and Word tokenizer is utilized to discover the rundown of words in strings. Punkt tokenizer is available in nltk for English language text.

(3) Noise removal: Noise elimination is tied in with eliminating characters digits and bits of text that can meddle with your content investigation. This is quite possibly the most fundamental content pre-processing steps.s

(4) Stop word removal: Once in a while, some amazingly regular words which gives an impression of being of little encouragement are prohibited from the jargon totally.

(5) Stemming: Inflected word removal process is called stemming.

(6) Lemmatization: Lemmatization typically alludes to doing things appropriately which is the last step in this chatbot design pre-processing.

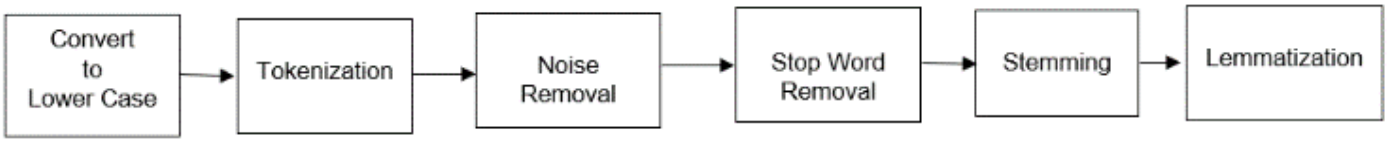

Fig 3: Pre-processing steps

A. Keyword matching

The paper uses the concept used by ELIZA for keyword matching. Watchword extraction otherwise called catchphrase discovery or watchword investigation is a book examination procedure that naturally extricates the most utilized and most significant words and articulations from a text. It sums up the substance of writings and perceive the fundamental points examined.

B. Generating Response

There are various techniques for generating response. This section discusses these techniques in detail.

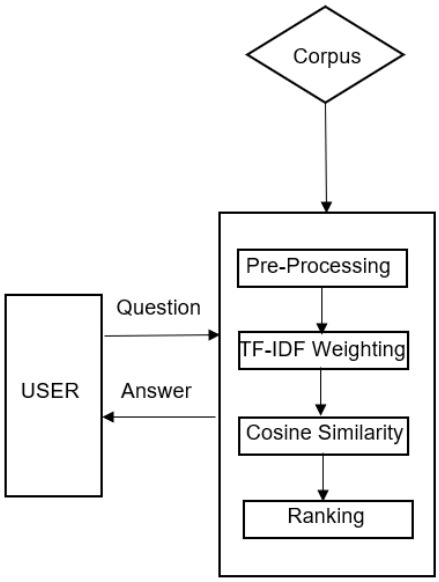

Fig 4: Chatbot Response Generation Steps

(1) Bag of words: After all the pre-processing operations the text is changed into a vector of numbers. The instinct behind the Bag of Words is that archives are comparable in the event that they have comparative substance. Likewise, we can learn something about the importance of the record from its substance alone.

(2) TF-IDF Approach: To rescale the recurrence of words by how regularly they show up in all records an approach of TF-IDF is utilized.

TF = (Number of occurrences of term t)/(Total no. of terms)

(3) Cosines Similarity: Cosine similarity metric is used for matching the query content with the corpus.

Cosine Similarity (d1, d2) = Dot product (d1, d2) /

‖d1‖ * ‖d2‖ Here d1, d2 are two non - zero vectors.

To produce a reaction from our bot for input questions, the idea of report comparability will be utilized. Chatbot program computes the similarity value of each word in the user query and the corpus. The maximum value is selected as the watchword. The program responds with the content associated with the watchword. On the off chance that it doesn't discover the info coordinating any of the watchwords, it restores a reaction: "I didn't get you".

IV. RESULT AND DISSCUSSION

This implementation can be applied to any of the domain specific chatbots. Some examples of domain specific chatbots are chatbots used in medical domain, agricultural domain, banking and financial sector, tourism, education etc. Based on the domain needs only the corpus needs to be changed. Keywords pertaining to a particular domain can be incorporated in the text file which is created for storing the corpus. However, for our demonstration purpose we have created a corpus containing details on chatbots. This chatbot can answer any queries related to chatbots.

Fig. 5 shows the output of the working of the chatbot. When a user is logged in the session, the chatbot welcomes user with a greeting message. User queries about chatbot design, the bot program identifies the word 'design' per the cosine similarity algorithm. The detailed description associated with the identified word is then given as the output to the user. User may logout the session using the word "bye'.

```
BOT: My name is NLP_BOT. I will answer your queries about Chatbots. If you want to exit, type Bye!
hi
BOT: hi

BOT: My name is NLP_BOT. I will answer your queries about Chatbots. If you want to exit, type Bye!
Tell me about chatbot design
BOT: design
the chatbot design is the process that defines the interaction between the user and the chatbot.the chatbot designer will defin
e the chatbot personality, the questions that will be asked to the users, and the overall interaction.it can be viewed as a sub
set of the conversational design.

bye
BOT: Bye! take care..
```

Fig 5: Output Snapshots

The pie chart for the response accuracy is shown in the figure below. The proportion of correctly responded, partially correct responses and correct responses are compared.

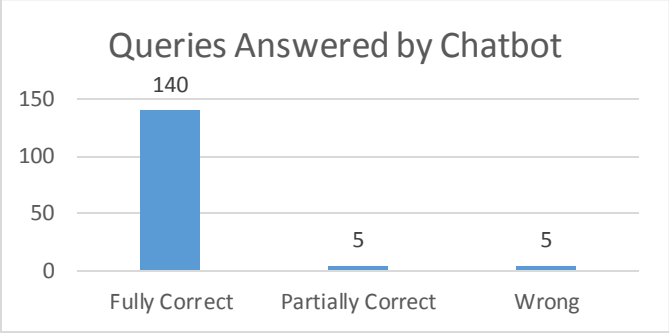

Fig 6: Chatbot response comparison

V. CONCLUSION AND FUTURE WORK

This paper proposes chatbot based on cosine similarity matching and this can be scaled to any domain. Based on domain specific requirements we have to change the corpus. We have analyzed our chatbot with different input sentences like complete sentence, incomplete sentence, related sentences etc. Based on the corpus it can respond to any such statements. This chatbot can be embedded in any of the company websites or even personal websites. This chatbot is tested with 150 questions and was able to correctly answer 140 questions so accuracy of this design is 93%. The proposed design can be further extended to accept voice commands as the inputs as future work.

REFERENCES

[1] Akshay Kumar, Pankaj Kumar Meena, Debiprasanna Panda, Ms. Sangeetha, "CHATBOT IN PYTHON" in International Research Journal of Engineering and Technology (IRJET) , Volume: 06 Issue: 11 | Nov 2019

[2] Jwala, K., Sirisha, G. N. V. G., & Raju, G. P. (2019). Developing a Chatbot using Machine Learning. International Journal of Recent Technology and Engineering (IJRTE), 8(1S3), 89-92.

[3] N.Naveenkumar, M.Hemanth Reddy, S.Sai Nikitha, T.SaiRam Reddy, "HUMAN-CHATBOT INTERACTION USING NLTK", International Journal of Creative Research Thoughts (IJCRT), Volume: 08 Issue: 2 | Feb 2020.

[4] P. Shanmugapriya, Harini M, "Development of Chatterbot using Python" International Journal of Computer Applications (0975 – 8887) Volume 176 – No. 21, May 2020.

[5] Lalwani, Tarun & Bhalotia, Shashank & Pal, Ashish & Bisen, Shreya & Rathod, Vasundhara. (2018). Implementation of a Chat Bot System using AI and NLP. 10.21276/ijircst.

[6] Kalla, Dinesh & Samiuddin, Vatsalya. (2020). Chatbot for Medical Treatment using NLTK Lib. 10.9790/0661-2201035056.

[7] Abdul-Kader, SA and Woods, JC (2015) 'Survey on Chatbot Design Techniques in Speech Conversation Systems.' International Journal of Advanced Computer Science and Applications, 6 (7). ISSN 2156-5570

[8] aza, & muha, & zura, & Ahmad, Nahdatul Akma. (2018). Review of Chatbots Design Techniques. International Journal of Computer Applications. 181. 7-10.

[9] Piccolo, Lara & Roberts, Shadrock & Iosif, Anna & Alani, Harith. (2018). Designing Chatbots for Crises: A Case Study Contrasting Potential and Reality. 1-10. 10.14236/ewic/HCI2018.56.

[10] Khanna, Anirudh & Pandey, Bishwajeet & Vashishta, Kushagra & Kalia, Kartik & Bhale, Pradeepkumar & Das, Teerath. (2015). A Study of Today's A.I. through Chatbots and Rediscovery of Machine Intelligence. International Journal of u- and e-Service, Science and Technology. 8. 277-284. 10.14257/ijunesst.2015.8.7.28.

[11] Colace, F. & De Santo, Massimo & Lombardi, Marco & Pascale, Francesco & Pietrosanto, Antonio & Lemma, Saverio. (2018). Chatbot for E-Learning: A Case of Study. International Journal of Mechanical Engineering and Robotics Research. 7. 528-533. 10.18178/ijmerr.7.5.528-533.

[12] Tiedan Zhu, Kan Li,The Similarity Measure Based on LDA for Automatic Summarization,Procedia Engineering, Vsolume 29, 2012, Pages 2944-2949, ISSN 1877-7058

[13] Hussain, Shafquat & Sianaki, Omid & Ababneh, Nedal. (2019). A Survey on Conversational Agents/Chatbots Classification and Design Techniques. 10.1007/978-3- 030-15035-8_93.

[14] Nimavat, Ketakee & Champaneria, Tushar. (2017). Chatbots: An overview. Types, Architecture, Tools and Future Possibilities.

[15] Kowalski S., Pavlovska K., Goldstein M. (2013) Two Case Studies in Using Chatbots for Security Training. In: Dodge e-words-works/.

[16] R.C., Futcher L. (eds) Information Assurance and Security Education and Training. WISE 2013, WISE 2011, WISE 2009. IFIP Advances in Information and Communication Technology, vol 406. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-39377-8_31

[17] https://www.machinelearningplus.com/nlp/cosine- similarity/#4howtocomputecosinesimilarityinpython

[18] https://en.wikipedia.org/wiki/Chatbot

[19] https://www.mygrseatlearning.com/blog/basics-of- building-an-artificial-intelligence-chatbot/#corpus

[20] https://www.geeksforgeeks.org/nlp-how-tokenizing-text- sentenc

[21] Patel, Viral & Kapadia, Daanyaal & Ghevariya, Deval & Pappu, Shiburaj. (2020). All India Grievance Redressal App. Journal of Information Technology and Digital World. 2. 91-99. 10.36548/jitdw.2020.2.002.

[22] Smys, S., and Wang Haoxiang. "Naïve Bayes and Entropy based Analysis and Classification of Humans and Chat Bots." Journal of ISMAC 3, no. 01 (2021): 40-49.