

Skillbot: A Conversational Chatbot based Data Mining and Sentiment Analysis

Babar Islam

Department of School of Engineering
London South Bank University
London, the United Kingdom.
babar.islam007@gmail.com

Muddesar Iqbal

Department of School of Engineering
London South Bank University
London, the United Kingdom
m.iqbal@lsbu.ac.uk

George Ubakanma

Department of School of Engineering
London South Bank University
London, the United Kingdom
george.ubakanma@lsbu.ac.uk

Spike van der Vliet-Firth

Jobs and Skills Programme Economy,
Jobs & Partnerships Housing,
Regeneration and Public Realm
Lewisham Council.
Spike.vanderVliet-
Firth@lewisham.gov.uk

Abstract—With the advent of technology, Artificial Intelligence is emerging exponentially. Using this advancement, chatbots are widely used in various sectors to accommodate users with their queries without waiting. In this study, work in the development, training, improvement, and chat sentiment analysis of Skillbot Chatbot is performed. First, the data was scrapped using tools like the GPT2 model from the Gov. UK website, and that data was used to train intents for the Skillbot model. After successful training, testing, and evaluation of Skillbot for better performance, conversations of users were analyzed deeply. Sentiment analysis was also performed as it is important to train the Skillbot to efficiently respond to users. Then, this project was deployed on Streamlit named Conversation Analyzer. Analysis was performed using different technologies like Natural Languages processing, Vader model for sentiment analysis, TextBlob for topic modeling of conversations, Streamlit for visualization, Rasa, Artificial Intelligence, and machine learning. Chatbot training with cleaned data and conversation analysis would be very beneficial for Skillbot to give users better services. The findings with massive data wrangling, model training for Skillbot, and chat analysis would provide results' evaluations with successful and unsuccessful dialogues with insights to help warrant future research and Skillbot improvement.

Keywords—Chatbot, Rasa, Sentiment Analysis Data Collection and Wrangling, Streamlit Deployment

I. INTRODUCTION

Chatbots are artificial intelligence interfaces that are used to interact between computers and human beings. Chatbots can be used by users to get answers to their queries. For instance, it can interact with the user to provide useful information, book a flight, place an order and the list goes on. Chatbots started in the 1960s with ELIZA till now with more efficient chatbots like Alexa, Google Assistant, and Cortana [1]. These chatbots are used for different purposes and getting more and more advanced.

Similarly, Skillbot chatbot is built using Rasa, Artificial Intelligence (AI), and Natural Language Processing (NLP) to assist job seekers to find jobs. Users can come on the chatbot to look for desired jobs and the Skillbot will provide relevant useful information and links to help the user find a suitable job. So, in this study, two areas are covered: The first one is to get the data, clean it, and impute it to the model for better training and compare the older model's version and the latest version of the model built with cleaned data. Secondly, after

the deployment of Skillbot, chat analysis is necessary as it can give us a lot of information about the chat that happened and analyze the sentiment of the users to give users better services. This way Skillbot can perform more efficiently in recognizing the user's mood and can serve the user in an appropriate way further training of Skillbot will also be done.

A. Skillbot Model Training

The first part of this study deals with training the model for Skillbot with the cleaned data which will be discussed in detail in later sections. The Skillbot was built in Rasa.

1) Rasa – Open-Source Conversation AI

Rasa is a machine learning framework that is open source and used to build Conversational chatbots. It is an amazing tool to create chatbots and have conversations between users and chatbots for business needs. Moreover, it is built in such a way that it can train itself based on the conversation performed with the users [2]. Rasa has two modules:

- Rasa NLU: It recognizes the user messages, entities, and intents using tokenization
- Rasa Core: It is used to hold users' conversations and provide a response based on user query identification with the entities. Simply, it gives out the output from the chatbot.

2) Architecture of Rasa

The architecture of Rasa can be divided into 6 steps shown in Figure 1 below. First, the user comes to the chatbot and writes the queries. The message is passed to the interpreter who extracts the intent of the message and passes the relevant entity. Tracker, which is clear from the name that it tracks the conversation between the chatbot and the user, and then policy holds the state of the conversation and decides which response should be given back to the user. Then, the action is passed to the user, and the action is also tracked by the tracker.

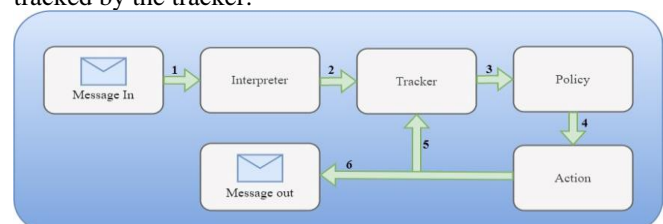


Figure 1: Architecture of Rasa

3) *Need of training the Model again*

The Skillbot model was already trained using some of the self-made data but the chatbot was not performing efficiently because the data was not enough. More data was obtained by scraping using the GPT-2 model. After scraping the data, it was put into the intent format to train the model. Then, the training was done again and still Skillbot was not performing well. After analysis, it was observed that the data need a lot of cleaning. There were issues with the context of sentences, formatting, and unnecessary elements like commas and colons, etc. So, bulk data cleaning was performed on the data.

B. *Conversation Analysis and Sentiment Analysis*

Secondly, work is done on the conversation which is created after the implementation of the Skillbot. The conversation between the user and Skillbot is then analyzed deeply, and this project was deployed on the Streamlit named Conversation Analyzer. The chat was taken from MongoDB manually and chat analysis was performed.

The following steps were taken to do the conversation analysis and sentiment analysis.

1) *Sentiment Analysis*

Sentiment analysis is very important as it tells us about the user's mood and makes the chatbot more empathetic. Secondly, it is also used to analyze the chat after the conversation to train our model for a better understanding of the user utterances and entertain the user accurately.

Sentiment analysis of the conversation was performed using the Valence Aware Dictionary for sentiment Reasoning (Vader) [3] which uses the Natural Language Toolkit (NLTK). It takes the single sentence and gives out the polarity for negative, positive, neutral, and compound sentiment for each sentence named neg, pos, neu, and compound

Finally, the Compound will be used to predict the sentiments for each sentence. For instance, if the compound value is less than 0, sentiment will be negative, if the value of the compound is greater than 0, sentiment will be positive and if the value of the compound is 0, sentiment will be neutral.

2) *Topic Modelling*

After that, topic modeling was used to find the topic or overview of the chat which was performed. For instance, it will analyze the chat and suggest us two to four topics that conclude the chat. It was also done using the TextBlob

3) *Data Cleaning and Chat Analysis*

After that, data cleaning was performed again because the chat was generated with the user and the user can type anything. Different types of cleaning were performed like removing emojis and URLs, dealing with missing values, removing stop words, removing punctuations, etc. Then, different type of chat analysis was done like the total number of messages per user and chatbot, and a word cloud was also generated to check the overall content of the chat, and all of this was then visualized using the Streamlit library for web deployment.

4) *Model Deployment*

After the analysis, the model was deployed using Streamlit named Conversation Analyzer for visualization. Data will be provided to the Conversation Analyzer and all the analysis will be performed there with just one click.

C. *Aims and Objectives*

The following are the aims and objectives of this project:

- To improve the efficiency of the Skillbot Chatbot by increasing the data and its quality.
- Evaluation and testing of the Trained Model of Skillbot
- Performing conversation analysis like sentiment analysis and Topic Modelling for the chat etc.
- Deploying Conversation Analyzer

D. *Paper Structure*

In the first section, the introduction of the project is explained along with the problem statement, aims and objectives, and technologies used in this project. In section II, the Literature review and background study are depicted. In section III, the actual implementation of the project is explained in detail with all the steps taken to do so. In section IV, results, testing, and evaluation are performed and explained properly. In section V, the future scope of the project is explored and explained properly like what innovations can be carried out or what improvements can be made. Finally, in section VI, the conclusion is made critically. Then at the end, references are provided .

II. LITERATURE REVIEW

The literature review covers the different background work of technologies used in this project i.e., Chatbots, RASA Framework, sentiment Analysis, etc.

A. *Chatbots*

Before the rise of technology, manual labor or people were there to answer the queries of the customers. Technology advanced, and chatbots were introduced. The conversational bots that are used to converse with people and intelligently respond to human questions automatically are known as chatbots. Initially, they were only text-based chat windows, but as technology advanced, they became more dynamic, intelligent, and sophisticated conversational platforms with aesthetically pleasing interfaces.

The first ever chatbot was older even than the personal computer and it was developed by MIT artificial intelligence laboratory in 1966 it was named Eliza [4]. It was a very simple chatbot as it is only capable of triggering the output according to the defined set of rules after receiving input from humans. After Eliza, a more advanced chatbot was created in 1972 by Kenneth Colby, Stanford University's professor named Parry [5]. This chatbot was created to emulate the behavior of paranoid schizophrenia patients. It was able to mimic human behavior. It was the first-ever chatbot to pass the Turing test. After Parry in 1988, another advanced chatbot named Jabber wacky [6] was developed by British developer Rollo Carpenter. It aimed to simulate human chat entertainingly and interestingly. It was the first-ever chatbot with a working mechanism of artificial intelligence technique called Contextual pattern matching. Later, it was also used for academic research purposes.

In 1995, ALICE (Artificial Linguistic Internet Computer Entity) [7] was constructed by Richard Wallace. The program works with XML Schema known as Artificial intelligence markup language (AML).

After 5 years, Active Buddy Inc. created another artificial intelligence-based chatbot in 2000. It was the first commercial messaging bot, and it has about 30 million users in the US, but its popularity becomes less in 2010 with the creation of SIRI by Apple for IOS. SIRI is a more intelligent virtual assistant [8] which receives voice queries and responds in voice and makes recommendations to the users according to their needs. To compete with Apple, in 2012 Google also launched a software named Google now [9]. The functionality of google now (aka Google Assistant, 2017) is the same as Siri. It provides recommendations based on the location, time, and search made by the user. After Siri and google now Microsoft also introduced Cortana for windows mobiles and PCs in 2014 and the functionality of Cortana is same as the Siri and google now. After that, Amazon also developed an intelligent virtual assistant Alexa [10] which works same as the Siri, google now, and Cortana.

In conclusion chatbots, from the 1960s to now are evolved in every aspect like functionality, intelligence, and interface. Similarly, for this study, a chatbot named Skillbot was built which helps users to find jobs and answer their queries

B. Rasa and its applications

RASA is a machine learning-based open-source AI framework. It is a python framework [11] used to build any type of text and voice-based chatbots or AI assistants.

RASA has become so popular, and it is being used in various industries to build chatbots. An intelligent virtual laboratory was built using the Rasa framework. It is built for teachers and students working in the laboratory. It helps them find the errors and solve them in the research in LAB. Another chatbot “jollity chatbot” is used to interact with users to make them happy and give them support in tough times. So, there are numerous applications of the RASA framework.

C. Sentiment Analysis

Sentiment analysis is the contextual deep mining of text using techniques of natural language processing and machine learning. It is used to extract meaningful information from the text and detects sentiments in the text [12]. Sentiment analysis is mostly used by businesses to understand the feedback about the business and brand by monitoring online conversations on the web and social media applications.

The primary objective of evaluating sentiment is to do a content analysis on reviews and conduct an examination of sentiment ratings. There are different levels of sentiment analysis [13].

- Sentence level: In this, the whole sentence is analyzed, and a score is provided for the sentence
- Document Level: In this, the score of sentiment is analyzed for the whole document or essay
- Word level: Word level gives us the sentiment for each word
- Aspect level: It gives us the polarity of the given target in the sentence

1) Types

There are 4 popular types of sentiment analyses for calculating the polarity [14].

- **Graded Sentiment Analyses:** In graded sentiment analyses, the sentiments of clients are differentiated

between positive, neutral, and negative feedback. For example, Google’s 5-star rating system. Here 1-star means very negative, 2-star negative, 3-star neutral, 4-star positive, and 5-star very positive feedback by the clients.

- **Emotion Detection:** Emotion detection sentiment analyses detects the emotions and feelings in the client’s feedback from happy to satisfied and angry to frustrated. Emotion detection sentiments use lexicons or complex machine learning algorithms. In lexicons, it uses a list of words that convey emotions.
- **Aspect-based sentiment analyses:** Aspect-based sentiment analyses (ABSA) is a text analysis technique used to detect the aspects and detect the positive or negative opinion about each aspect from a complete sentence.
- **Multilingual sentiment analyses:** Multilingual sentiment analysis is the sentiment analysis of clients’ feedback and reviews on social media in their native language instead of English. It is important as many of the world’s population didn’t speak English.

In this study, Sentence level sentiment analysis is used because we are dealing with conversational data. Emotion detection is detected using the Vader NLTK library [15] introduced in 2014. It is a pre-trained model which uses lexical features for the sentiment’s identification. This library was used to detect the sentiment because we didn’t have the labeled data with polarity available.

III. PROJECT IMPLEMENTATION

This project is divided into two sections which contain:

- Improving the Skillbot efficiency by training the Skillbot model with the cleaned data
- Doing chat data analysis and sentiment analysis of the chats between Skillbot and the user.

A. Improving skillbot efficiency

1) Business Problem

The Skillbot is a Council Project which was developed by the developers working under the LSBU management. The problem was that the Skillbot chatbot was not performing efficiently. This problem was solved by getting more data by using the data scrapping technique.

2) Data Collection

The data was scrapped from the internet using the GPT2 model for custom text generation. It is a generative pre-trained transformer 2 used for extraction of the generalized or memorized based data [16]. Hence, data was scrapped for almost 257 Intents. The data was in raw format. Then, there was another hitch which was cleaning this data

3) Data Cleaning

Data was the main aim and concern of the project as the efficiency of Skillbot could only be improved by feeding more accurate data. The first issue with raw data was that it was not in the format required for NLU training format. So,

the data was put into the Intent file and in the proper formatting required for the NLU training. Data had a lot of inconsistencies and anomalies, and the size of the dataset was massive of size almost 40,000 sentences for all the intents.

It is shown in Figure 2, After scrapping, Data had some inconsistencies like contextual errors, punctuation, incomplete sentences, meaningless sentences, etc.

examples:

- I want career in clinical health care. I care about my health. I work a good job,
- I want career in clinical health care," she said what this legislation has done is we've
- I want career in clinical health care." So he did. He also started off his career in research
- I want career in clinical health care for ALL patients, the best way to do it would be by
- I want career in clinical health care, I want personal care. I want to build a reputation as
- I want career in clinical health care," he said As a physician, I want
- I want career in clinical health care." But he said he could find a way to do
- I want career in clinical health care. For all the women who have gone to college and found careers
- I want career in clinical health care, but it does take some work, and most of us have
- I want career in clinical health care we need healthcare professionals there because they have the skills to help
- I want career in clinical health care so that a few of my colleagues or colleagues can have access for
- I want career in clinical health care." And yes, that's how the bill makes the very important
- I want career in clinical health care. I want to teach our kids what they can do. I
- I want to work at clinical health care, a full schedule of health services that has the potential to help
- I want career in clinical health care because they didn't treat it too well. Now, what they
- I want to work at clinical healthcare," he says I want my son to understand that they
- I want to work at clinical healthcare and I want to do health maintenance and I want to do everything
- I want to work at clinical healthcare. I want to be able to look for solutions that work,
- I want to work at clinical healthcare. I want to find ways to bring that into our system that
- I want to work at clinical healthcare. My father was a real great businessman who went to great lengths
- I want to work at clinical healthcare centers, clinics like OBGYNs. My heart is always pounding
- I want to work at clinical healthcare, as well as provide access to high quality health care and better
- I want to work at clinical healthcare! While not really saying that, she said
- I want to work at clinical healthcare as part of the NIH, and so I have to do something
- I want to work at clinical healthcare centers around the world. I'm interested in how patient experience impacts
- I want to work at clinical healthcare because the only way your life for you is a mess is to
- I want to work at clinical healthcare, medical education, and education as a therapist." Dr

Figure 2 Uncleaned data

Majorly, data were cleaned manually. Different sorts of cleaning were performed in the dataset that is as follows:

a) *Contextual Errors*: The first and most important issue with the data was the contextual inconsistencies. It happened during the scrapping. The data was scrapped using data augmentation in which half a sentence is given to the model and in return model scraps the data through the website and completes that sentence. For example, a set of keywords will be fed to the model like this:

- "I want a career in clinical health care"

And the model will take up the sentence and find a similar sentence on the website and find its complete sentence. So, in our case it gave out the sentence like this:

- "I want a career in clinical health care, a full schedule of health services that have the potential to help"

So, by doing so, sometimes the model was unable to identify the context. For instance, the following is the case as an example to show how it was missing the context.

- I am looking for a job as a port worker. I have installed many computers servers

In the above example, the intent is about ship transportation but the model scrapped data with the wrong context and understood it as it is looking for data like ports that are used in computer architecture designs and configurations.

So, many similar sentences in some Intents were found and dealt with by careful observation of each sentence in the dataset.

b) *Issues with scrapping keywords*: The second most important issue was within the keywords which were used by the model to scrap the data. There were some spelling errors in the keywords. For instance, some errors are shown in Figure 3 below. It can be seen that with the wrong keywords, the model was unable to scrap data properly. The model could only scrap two correct sentences with that specific Intent. These human-made errors were difficult to identify

I want to work as a lawyer Just a lazy, poor lawyer. You
 I want to work as a lawyer too so let's talk business." Now, let's
 I want to work as a lawyer so she'll have to look and talk as a
 I want to work as a lawyer myself at least to get this up, but I don
 I want to work as a lawyer but it's really hard. If you have had
 I want to work as a lawyer who was more than happy to work with us. If

Figure 3: Errors in Keywords for scrapping

c) *Managing Meaningless utterances*: There was an issue with the data where the model could not scrap the data with meaningful sentences. For example, there was no meaning in the sentence which was confusing for the model to identify Intent. In this Figure 4 below, some of the meaningless sentences can be seen.

I want career as a barista, but I didn't want to look like my dad "
 I want career as a barista and a painter, said Ms. P\xE9rez, whose work"
 I want career as a barista, but I'm a man," she said. The couple had
 I want career as a barista," she said Do you know who is a
 I want career as a barista at least," said Bill For a long time

Figure 4: Managing Meaningless utterances

d) *Removing Unnecessary Sentences*: This step of cleaning the data correlates with the above heading. Some of the sentences were rephrased but when the data for specific Intent became enough, the remaining meaningless sentences were removed from the dataset.

e) *Removing Punctuation*: The other issue with the data was removing the punctuation. There were so many punctuations, semi-colons, inverted commas, unnecessary question marks, and dots in the dataset shown in Figure 5 below, which were removed to make the data consistent.

I want career as a Bar person. To see my family..." There is no indication that
 I want career as a Bar person. I can make ' 'it happen. I want a wife. I
 I want career as a Bar person for those ' ? things and maybe maybe it would help
 I want career as a Bar person, not as a bartender. That's kind of my big question

Figure 5: Punctuation Errors

f) *Removing duplication*: There were also some duplicate sentences in the dataset which created data redundancy. These were also analyzed and located carefully and removed during the process of data cleaning.

g) *Removing Extra Spaces*: In the dataset, there were some extra spaces available in the sentences, which would take extra time for the model to train and sometimes can confuse the model to interpret or understand during the training. So, such extra spaces were removed from the data.

4) Model Training and Evaluation

After the hefty task of cleaning the data, the data was merged, and it was ready for the training of the model for Skillbot. Then, stories, Intents, entities, rules, and actions were put together with the data.

- **Rules**: Rules are kind of small chunks of conversation that usually go the same way i.e., Greeting like Hi, hello, etc.

- **Entities:** Entities are the information that is inside the user messages. It tells the bot what kind of information the user is looking for.
- **Intent:** These are classifications of the classes which contain the possible user utterance and when the user gives input, the model classifies and Identifies that Intent, and performs the actions. The data in the Intent are the example to say things by the user. It is normally named NLU.YML
- **Actions:** Actions are the pre-defined responses in the chatbot. If the user sends a message. The Intent will be identified and based on that Intent the response will be given back to the user by action
- **Stories:** Stories are considered the training data that our chatbot uses to teach itself what should do after the intent is found. Stories would guess based on the confidence level that what should be the next response. It checks the Intent and on the bases of the Intent, it looks for the action which is trained in stories. It is a kind of conversational pattern which maintain the context of the conversation.

Finally, the data were imputed to the Rasa NLU and the model was trained based on that cleaned data.

The Skillbot model was trained successfully. It was trained with the 85% confidence level of Intent identification to give back a response. 85% confidence was chosen because if the confidence level was less than that the Skillbot was giving unrelated responses back to the user. On the other hand, if the confidence level was more than 85%, the Skillbot was unable to identify the intent because some of the user utterances don't get much confidence. After the training, Skillbot is now performing much better giving excellent responses, and identifying the Intent of the user much more efficiently. A sample of the chat as intent recognition can be seen in Figure 6 below. Skillbot is identifying the Intent and gives responses accordingly.

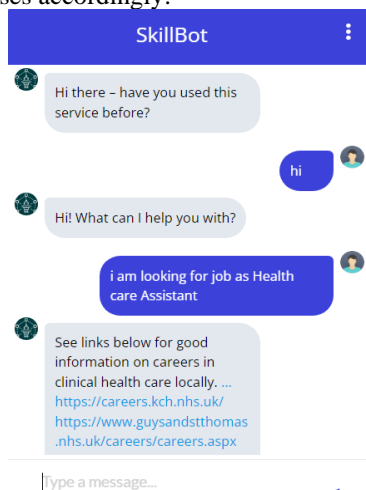


Figure 6: Skillbot chat sample

After the successful implementation and testing of the model, the model will be evaluated for intent as major work was done on the Intents classification improvement to increase the efficiency of the model. So, simple "Rasa Test" command will be run for evaluation Then, the results will be

compared for both the previous and latest models for evaluation. Particularly, the Intent classification report will be discussed as work has been done on Intent. The intent classification report gives the precision, recall, f1 score, and support for each Intent. It will also give the micro average, macro average, and weighted average Report that gives the overall situation and is important in this case.

- **Micro Average:** In this report, All the values contribute to the final average metric
- **Macro Average:** In this, instead of values, all the classes contribute to giving the average metric.
- **Weighted Average:** The contribution of each class to the average is weighted by its size.

Hence, we evaluated our model for intents evaluation using Weighted Average because it is a multi-class evaluation problem. So, Weight Average gave accurate reports.

B. Chat Analysis and Sentiment Analysis

Chat analysis and sentiment analysis are very important factors in the conversations as these are helpful to understand the users' needs, and satisfaction with our product.

1) Business Problem

In this part, the chat analysis and sentiment analysis were performed after *the chat has happened*. It also expands to the sentiment analysis of the user's emotion only during conversation and the overall conversation. Finally, the topic modeling technique is used to recommend the topics of the whole conversation that happened between the user and the Skillbot, to understand the context of the whole conversation.

2) Data Acquisition

After the training of the Skillbot, the chat was performed with Skillbot by different users for testing purposes. That conversation was stored in the Mongo DB. The chat was extracted in the format of Bot & user. The sample chat that happened between two people can be seen in Figure 7 below. It is used here for demonstration purposes of implementation. Later, a real chat with the labeled sentiment will be used to evaluate the performance of the Conversation Analyzer.

	Sender	Message
0	Bot	Yes what happened?
1	User	Just minute
3	User	Teacher sent me this topic
4	Bot	For what?
5	User	But I couldn't understand what is the topic
...
88	User	Yeah for example online demo and online platfo...
89	Bot	Yeah...Kind of...
90	User	Hmm
91	Bot	This is only study wise..Add some living style...
92	User	For 3Should I write questions too 😊

89 rows × 2 columns

Figure 7: Sample Chat dataset

3) Data Cleaning

After getting the data, data cleaning was again required because there was an interaction of the user and the user can type any type of data i.e., emojis, links, and null values. Some of the data cleanings were done based on the assumptions which can be done by the user i.e., duplication, etc.

a) **Removing Null Values:** In this conversation dataset, null values were checked and there were no null values available conversation the dataset.

b) *Removing Duplicate values*: Another traditional thing in doing the data analysis is to look for duplicate values. There were checked and removed from the dataset.

c) *Removing URLs*: URLs were removed in this step because they don't add any meaning during the sentiment analysis. To remove the URLs, the URLExtract library was used to find out the links available in the conversation.

d) *Removing Emojis*: Emojis are a new way of expressing your feelings. Emojis were removed from the dataset to do the sentiment analysis because they were creating some errors while parsing for the sentiment analysis.

e) *Removing Punctuation*: Punctuation was also removed to do the sentiment analysis. Because the model would have been confused and could not identify the confidence of the punctuation marks like "?" or "... " or exclamation marks etc.

f) *Removing Stop words*: After that, stop words were removed from the dataset for doing the data analysis.

For this purpose, the NLTK corpus of stop words is used which are accessed using the following command:

```
"from nltk.corpus import stopwords"
```

4) Conversation Analysis

After the cleaning of the dataset, conversation analysis, sentiment analysis, and topic modeling are performed.

The Analysis performed is deployed on the website and data can be analyzed. Moreover, this analysis can do all analysis performed on both User chat & bot chat individually and as a whole conversation. Both individual and Whole chat analysis and sentiment analysis will be shown in the Results Section where this analytical tool will be deployed.

The following are the questions answered in this type of analysis.

- Word cloud of the conversation
- Most Frequent Words
- Distribution of Sentiments in the Conversation
- Percentage and number of the Messages by both user and Bot
- Total Messages in the Chat
- Total Words in the Chat
- Number of Links shared

a) *Word Cloud of the Conversation*: Word cloud is a sophisticated and very powerful way of visualization and representing text processing. It highlights the words which are frequently used in bigger and bold letters by using different colors [17]. Word cloud can be used for different purposes like analyzing the customers' feedback, finding trending topics on social media, or, analyzing the essay.

b) *Chat Messages Analysis*: The number of messages, total words in the messages, and several links shared were analyzed for both individual bot and User and overall analysis was performed.

c) *Most Frequent Words*: The number of Most Frequent words in the whole chat and chat of the user.

5) Implementation of Sentiment Analysis

For the sentiment analysis, the Vader Analyzer model was used. It is used by API for NLP tasks. It is a lexicon-based sentiment analyzer. i.e., it does the sentiment orientation based on phrases or words The reason to choose this analyzer

was that for this conversational analysis, labeled data was not available. Vader is a pre-trained model which works well with unsupervised learning. Vader analyzes each sentence and returns us four values.

- Pos: The probability of Positive sentiment
- Neg: The probability of Negative sentiment
- Neu: The probability Neutral Sentiment
- Compound: A normalized compound score is a number that ranges from -1 to 1 and is calculated by adding all lexical ratings.

Vader is based on a lexical approach that maps the words to lexicons and those lexicons can be used to assess the sentiment of a sentence. It takes words excluding stop words and then calculates sentiments for those words using tokenization and then it takes the average of all the words for each sentence and gives the sentiments of the sentences in the four values mentioned above. Then, the sentiment was extracted for each sentence using the Compound value. Finally, the score was calculated for each sentence. All the positive, negative, and neutral sentiments in the chat are summed and then the maximum of all the sums will be considered as the sentiment for the whole chat.

6) Topic Recommendation (Modeling)

Finally, Topic modeling was also performed based on the conversation using TextBlob. Topic modeling recommends us the overview or the topics of the whole conversation. First, all the Messages were merged into one single String. Finally, the string was passed to the blob. Then blob extracted the nouns and tags from that string after lemmatizing the words. *"Lemmatization groups a word's inflected forms so they can be analyzed together. Lemmatization adds context to words. It links similar-meaning words to one."*

Then, the 'word' function of TextBlob was used to manipulate the words. Words were converted into plurals and random 4 nouns were extracted to recommend the topics of the Conversation.

7) Evaluation

In this part, different evaluation metrics are used to evaluate the implementation of Sentiment Analysis like classification report, confusion matrix, and accuracy.

8) Deployment

After the Cleaning, Implementation, and evaluation, the conversation analyzer is deployed on streamlit to analyze and do all the processes explained to have a quick analysis of the conversation. The data will be provided to the Conversation Analyzer and then with one click it automatically performs all the steps. *All the analysis steps can be taken for the Overall chat between the user and bot and individually by either the bot or user.*

IV. RESULTS, TESTING AND EVALUATION

A. Skillbot Intent Evaluation

After cleaning of data, it was imputed to the Skillbot model, and the model was trained with new refined data. Then, the model was tested using the "Rasa test" command. Different files of results were generated. So, Intents based on 143 stories for testing are evaluated and explained here for both previous and current models. A comparison of the old

Intent report and the new intent report is shown in Figure 8 below.

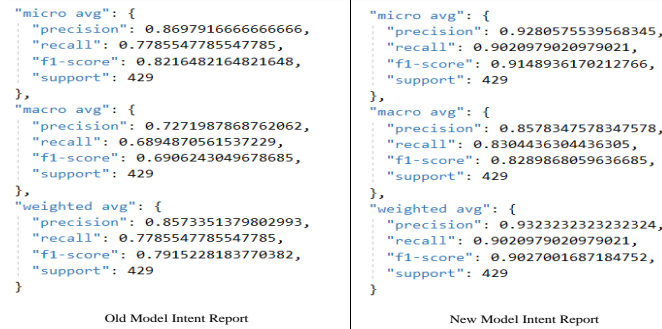


Figure 8: Comparison of Old vs New Intent Report

1) Micro average

It is calculated as the precision of all values. In the old model, the precision is 86% while in the new model it shows 92% which has improved in the new model. The recall is 77% in the old model while it is increased to 90% in the new model for the identification of all positive instances. While f1 score for the old model is 82% and it has improved to 91% which is harmonic means of both recall and precision. Support for both was 429.

2) Macro Average:

It is calculated as an average of the precision of all the classes. The precision in the old model was 72% while in the new model, it has improved to 85% which is the percentage of correct positive classification. For recall, in the previous model, it was 68%, while in the new model, it is increased to 83% drastically and support for the old model is 69% while in the new model average harmonic means is increased to 82%. In Figure 9, the old model to the left side was unable to identify the correct Intent and expected it to be the wrong interview intent. But after data cleaning and training the new model (shown on the right side), the same query was passed to the Skillbot, and it was successful in identifying the Intent of Tech Careers. Hence, our model is performing much better after the training model based on newly cleaned data.

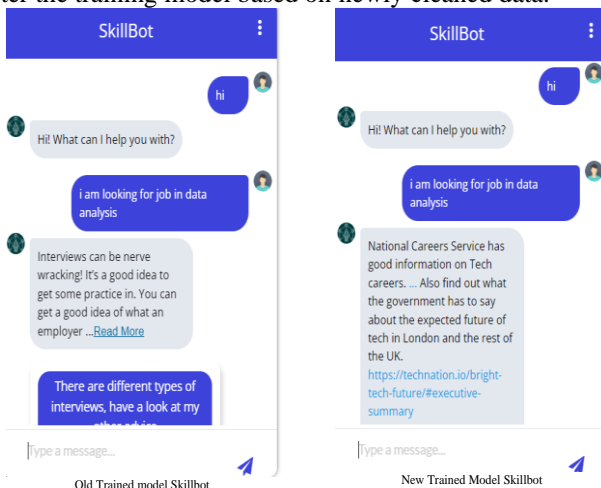


Figure 9: Old vs new model-based Skillbot Results

B. Conversation Analyzer and Sentiment Analysis

On streamlit, the conversation taken from Skillbot will be uploaded with the "Upload" Button and It will be shown as a Data Frame on display shown in Figure 10 below.

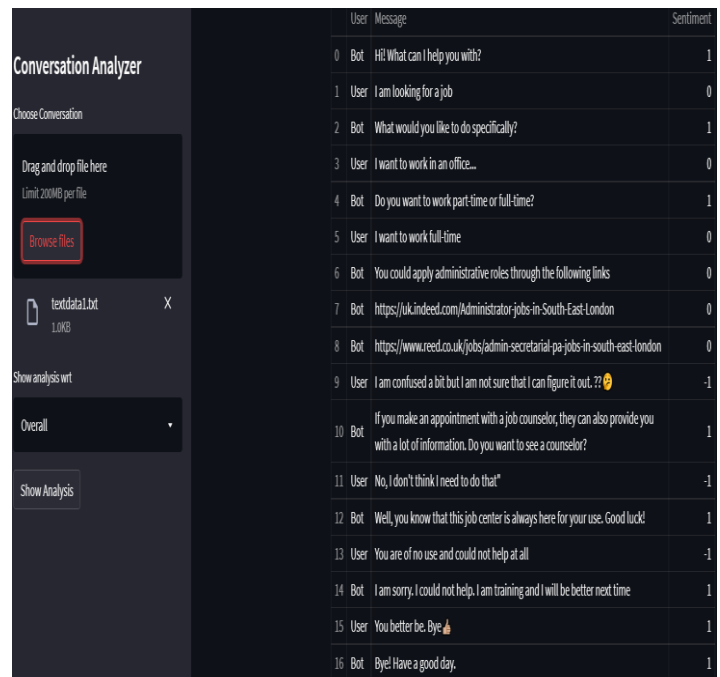


Figure 10: Display of Conversation Analyzer

After the loading of data, Conversation analysis will be performed with just one click which can be seen on the left bottom. Here, the drop-down button can be seen which means analysis can be performed on an overall conversation between the user and Skillbot as well as individual analysis can also be performed on either Skillbot or the User. Only overall will be explained here for demonstration purposes.

1) Overall Conversation Analysis

First, the Overall analysis was performed shown in Figure 11 below. The total number of Messages is 17 containing 155 words in the conversation. Then Total links shared in the conversation by the Bot can be seen which are only 2 in this conversation. Through this, we can analyze how much Skillbot helped the user by providing the Links for queries. After that, sentiment analysis is computed which is "Positive" in this case. Through analysis, we can analyze, how our conversation went or if it needs consideration in serving the User. Then, the Contribution of conversation by each user and bot in numbers and percentages are shown which are very helpful in analyzing how much responsive our chatbot is to the user. In this case, Skillbot contributed 59% and users contributed 41% of chat.

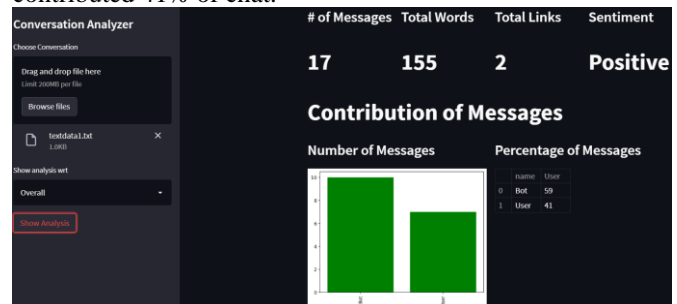


Figure 11: Overall Conversation Analysis

After that, the word cloud of the whole conversation is displayed in Figure 12 below. Word cloud gives us an idea about the Textual data and highlights the most used words in bold and colored format. Here, it can be analyzed that the

conversation is about a full-time job, work or office, or counselor. We can analyze that chat is being conducted about full-time jobs and someone is looking for jobs.

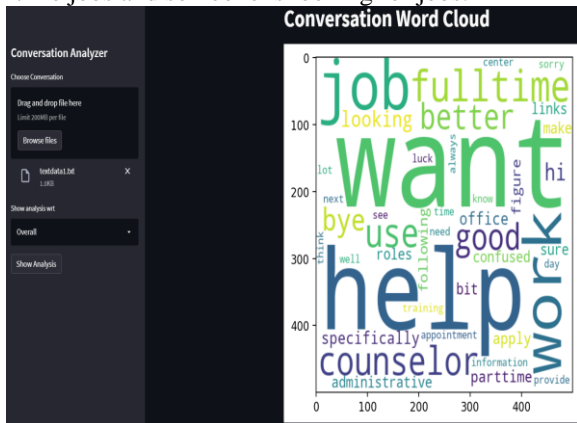


Figure 12: Overall Conversation Word Cloud

After that, the most frequent words and recommended topics can be seen in Figure 13 below. Through frequent words, we can analyze what the discussion was about and through topic modeling, we can have an idea about the conversation being happened. For instance, jobs and counselors are topics suggested and this chat happened regarding jobs and a referral was given to the counselors.

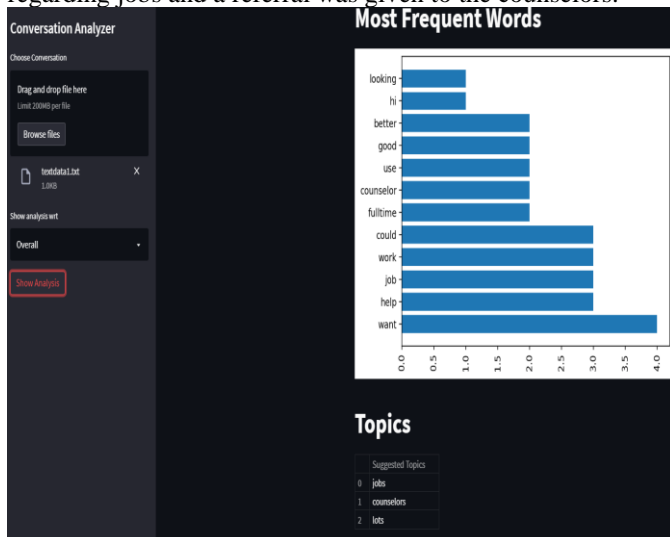


Figure 13: Most frequent words

Then, the Sentiment Distribution is also plotted shown in Figure 14 below. Here Predicted Positive sentiments are 10, 4 Neutral Sentiments, and 3 Negative Sentiments.



Figure 14: Overall Sentiment Distribution

Finally, at the end of the Overall Conversation, Sentiment Evaluation is performed using a Confusion matrix, Accuracy, and classification report which can be seen in Figure 15 below. A confusion Matrix plotted can be seen that all the values are predicted True except for two only. Similarly, we got an accuracy of 88 percent which assures our sentiment is performing well. Finally, a Classification report is generated. For Neutral (0), The precision is 100 percent, whereas recall is 67% which means 67% positives were correctly identified but the F1 score is 80% which shows the good harmonic mean. For Positive (1), precision is 80%, recall is 100% and f1 score is 89% which shows good performance in identifying positive occurrences. Finally for Negative (-1), precision, recall, and f1-score gave 100%, which means our model was excellent in identifying the negative sentiment. The weighted average of the f1 score and accuracy is 88% which gives a lot of reliability to our model for predicting the model.



Figure 15: Overall Evaluation of Sentiment

V. FUTURE SCOPE

The Skillbot can be improved with more training and adding new features. The followings are some features that can be added in the future.

To improve efficiency, data was scrapped, cleaned, and then imputed to the model. The data was extensive and very dirty with a lot of anomalies. Although data was cleaned properly and got excellent accuracy of intents identification. Still, small room for improvement is there due to the lack of time. More training will be done with the passage of time and usage of Skillbot, at this point the cleaning won't be much manual. Now the chat would be performed with the Skillbot and if the Skillbot is unable to identify the intent, it would give the fallout intent and it would be stored in the conversation in DB, where fallout Intents would be observed and rectified in the model for improvement. Rasa has introduced a new feature Insights [18], which analyzes the NLU training data and gives suggestions to improve the training data. It identifies the Intent with low confidence and tells us where misclassification is happening.

There is one more approach where we can increase the intent classification. First similar words in the intents can be extracted into one Intent and then can be accessed by identifying entities (values) from the user message and match with the actual Intent [19]. For instance, if for different intents query substrings are “I am looking for a job in (medicine)” or “I want a career in (tech)”. Then, these substrings will be extracted and put into one intent: assumably “searching career” and then the later substring, which is mentioned in braces above, will be taken as an entity and matched with the actual intent of medicine and tech. By doing this, the model will not get confused in identifying the intent and perform more accurately.

Moreover, If the user asks the Skillbot about something that is not trained in the model, it will be referred to the developer and the developer will add that new inquiry to the model. This way new data will be added to the model with time.

Live sentiment analysis can be performed which would enhance user satisfaction. For instance, if the user is talking with Skillbot and gets annoyed at some point, the model will analyze the sentiment of the user and at some certain threshold, it will be redirected to the live chat support team.

In this analyzer, topics are also suggested based on the pre-trained model of TextBlob. Our model can be trained using Skillbot data. We must manually add topics to the sentences and make a huge dataset for training which is not available for such kind of chatbot.

Finally, another feature in the future can be added which is the latest technology “Abstract Summarization” [20]. It takes the whole document and summarizes the document into shorter paragraphs which is quick to read. This feature can be added to the Skillbot analyzer to quickly analyze the conversation and understand what the user wants or looking for.

Hence, these all suggestions or features can be introduced to Skillbot in the future to make it a more competitive product that can help people to find jobs or knowledge regarding jobs easily. However, the expected work regarding the project scope is implemented successfully.

VI. CONCLUSION

The study aimed to achieve two goals: (1) to get and clean the dataset to train Skillbot to improve efficiency. (2) perform sentiment analysis and build a Conversation Analyzer to better understand the users’ needs.

First, the inconsistencies in the dataset like contextual, extra punctuation, scrapping keywords, and meaningless and unnecessary sentences were removed. Then, data were imputed and trained with an 85% confidence threshold for the Skillbot model and as a result, Skillbot became more efficient in analyzing user queries and responding more accurately. It was also evaluated using the classification report of Intents, where its weighted average accuracy and f1 score was 90% improved from 79% percent.

Secondly, labeled conversation with actual sentiment data was taken. First, sentiment analysis was performed on it using Vader Sentiment Algorithm. It was performed on the whole conversation and the user’s Messages. Accuracy during the evaluation was 88% and the

weighted average f1 score was also 88% which depicts the model was good at predicting the correct sentiment. Topic modeling was also performed to predict appropriate topics for the whole conversation.

Both of those goals were achieved, and the project was successful. Skillbot performance increased surprisingly with data wrangling.

ACKNOWLEDGMENT

The work presented in this paper is carried out as part of Skillbot – a chatbot-based Job market Intelligence tool project funded by The Mayor and Burgesses of the London Borough of Lambeth.

VII. REFERENCES

- [1] Ina (2017). The History of Chatbots - From ELIZA to Alexa. [online] Chatbots and Voice Assistants from Online Available at: <https://onlim.com/en/the-history-of-chatbots/>.
- [2] AskLua (2020). What is RASA? — the open-source AI for building conversational chatbots. [online] Medium. Available at: <https://medium.com/@asklua/what-is-rasa-the-open-source-ai-for-building-conversational-chatbots-8a86bef47ec4>
- [3] Nltk.org. (2019). *nlk.sentiment.vader* — *NLTK 3.4.5 documentation*. [online] Available at: https://www.nltk.org/_modules/nltk/sentiment/vader.html
- [4] Njit.edu. (2018). Eliza, a chatbot therapist. [online] Available at: <https://web.njit.edu/~ronkowitz/eliza.html>.
- [5] Kuligowska, K. (n.d.). Chatbot Parry, Kenneth Mark Colby | Virtual Assistant Parry | Virtual agent Parry | Chat bot Parry | Conversational agent Parry | (9706). [online] chatbots.org. Available at: <https://www.chatbots.org/chatbot/parry/>.
- [6] Jabberwacky.com. (2018). jabberwacky - About Thoughts - An Artificial Intelligence AI chatbot, chatterbot or chatterbox, learning AI, database, dynamic - models way humans learn - simulate natural human chat - interesting, humorous, entertaining. [online] Available at: <http://www.jabberwacky.com/j2about>.
- [7] B, S. and B, S. (2020). *Chatbot History: The ALICE Chatbot - YakBots*. [online] yakbots.com. Available at: <https://yakbots.com/chatbot-history-the-alice-chatbot/>
- [8] <https://www.facebook.com/thebalancesmb> (2019). *A Quick Guide to What a Virtual Assistant Is and What It Does*. [online] The Balance Small Business. Available at: <https://www.thebalancesmb.com/virtual-assistant-1794441>.

- [9] Webopedia. (2013). *What is Google Now?* [online] Available at: <https://www.webopedia.com/definitions/google-now/> [Accessed 17 May 2022].
- [10] developer.amazon.com. (n.d.). *Amazon Alexa Official Site: What is Alexa?* [online] Available at: <https://developer.amazon.com/en-GB/alexa>.
- [11] Gour, R. (2019). *Python Web Framework — A Detailed List of Web Frameworks in Python*. [online] Medium. Available at: <https://towardsdatascience.com/python-web-framework-a-detailed-list-of-web-frameworks-in-python-1916d3c6222d#:~:text=Python%20Web%20framework%20is%20a>
- [12] Hussein, D.M.E.-D.M. (2018). A survey on sentiment analysis challenges. *Journal of King Saud University - Engineering Sciences*, [online] 30(4), pp.330–338. doi:10.1016/j.jksues.2016.04.002.
- [13] Hussein, D.M.E.-D.M. (2018). A survey on sentiment analysis challenges. *Journal of King Saud University - Engineering Sciences*, [online] 30(4), pp.330–338. doi:10.1016/j.jksues.2016.04.002.
- [14] Summa Linguae. (2022). *Vibe Check: An Introduction to Sentiment Analysis*. [online] Available at: <https://summalinguae.com/data/introduction-to-sentiment-analysis/#:~:text=Graded%20Sentiment%20Analysis>
- [15] Calderon, P. (2018). *VADER Sentiment Analysis Explained*. [online] Medium. Available at: <https://medium.com/@piocalderon/vader-sentiment-analysis-explained-f1c4f9101cd9>.
- [16] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I. (n.d.). *Language Models are Unsupervised Multitask Learners*. [online] Available at: https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.
- [17] Analytics Vidhya. (2020). *What is Word Cloud | Guide To Create A Word Cloud in Python*. [online] Available at: <https://www.analyticsvidhya.com/blog/2020/10/word-cloud-or-tag-cloud-in-python/>.
- [18] Rasa. (n.d.). *Discover Actionable Ways to Improve your Training Data with Insights*. [online] Available at: <https://rasa.com/blog/improve-your-training-data-with-insights/> [Accessed 19 May 2022].
- [19] Rasa Community Forum. (2021). *Improve NLU accuracy / Avoid intent confusion*. [online] Available at: <https://forum.rasa.com/t/improve-nlu-accuracy-avoid-intent-confusion/42182/3> [Accessed 19 May 2022].
- [20] Parida, S. and Motlicek, P. (2019). *Abstract Text Summarization: A Low Resource Challenge*. [online] ACLWeb. doi:10.18653/v1/D19-1616.