CHATBOT DESIGN DOCUMENTATION

MODULE CODE: ISYS30221

MODULE NAME: ARTIFICIAL INTELLIGENCE 202021

NTU ID: N0830182

NAME: DAYEETA DAS

MODULE CODE: ISYS30221
MODULE NAME: ARTIFICIAL INTELLIGENCE 202021
NTU ID: N0830182
NAME: DAYEETA DAS

CONTENTS Page Number

1.INTODUCTION

Chatbots have been around for a long time. They can provide services at any hour of the day. The massive success in making the bots of these generation more and more sensible in understanding the human interrogation has expanded their services in a larger sector. From retail to banking and online food and grocery delivery, chatbots have become the ultimate choice for businesses in expanding their domain.

1.1. Overview

This project is concerned with the development of a chatbot named "Dahlia" which serves as the primary assistant for an online flower decoration website named "Bloom". Dahlia makes use of extensive AI features such as rule-based and similarity based model, logic-based model and image classification for helping out the customers with their purchase. Additionally, It also has a multilingual component and is capable of having conversations in 10 different languages: Bengali, Hindi, Punjabi, French, Italian, Assamese, Greek, Japanese, Odia and Arabic. It also has features of speech recognition, sentiment analysis and handwriting recognition.

1.2. Aim of the coursework

- ❖ Create a rule-based chatbot to make use of a pre-defined set of rules contained in an .aiml file to answer to the queries of the customer.
- ❖ Add a similarity-based component to the chatbot so that it can make use of a saved knowledgebase as a .csv file to answer to the user's queries based on the tfidf and cosine similarity.
- ❖ Add an image classification model using CNN for predicting single and multiple images uploaded by the user to the chatbot.
- ❖ Add a logic based component to the chatbot for making use of pre-defined logic contained in a knowledgebase(.csv file) for answering queries of the customer based on patterns.
- ❖ Add a multilingual component that will allow the bot to have conversations in different languages.
- ❖ Add a feature that would enable the bot to understand queries from a voice recording.
- ❖ Add a feature that will enable the bot to understand queries from a handwritten note.
- ❖ Add a feature of sentiment analysis to the bot that will enable it to analyse the sentiments from the reviews of the customers.

1.3.

Goals	Status
Add rule-based component.	Completed
Add similarity-based component.	Completed
Add image classification model using CNN.	Completed
Add logic-based model.	Completed
Add multilingual component using Microsoft Azure Services.	Completed
Add voice recognition feature using Microsoft Azure Services.	Completed
Add handwriting recognition feature using Microsoft Azure Services.	Completed
Add sentiment analysis component using Microsoft Azure Services.	Completed

2. SYSTEM REQUIREMENTS

2.1. System Requirements based on user's perspective:

- The chatbot should display a welcome message to the customer.
- The chatbot should make use of the best possible model (that is the rule-based/similarity-based/image classification/logic-based) to answer to the user's queries.
- There should be an option for helping a new user out.
- The chatbot should allow the user to upload an image or a set of images for prediction based on the user's choice.
- The chatbot should maintain a general etiquette while conversing with the user

2.2. System Requirements based on a developer's perspective (MosCoW Analysis):

2.2.1. Must Haves:

- The chatbot must have a rule-based and a similarity-based component for answering the customer's queries.
- The chatbot must have an image classification model for predicting images.
- The chatbot must make use of a logic-based knowledgebase for answering the user's queries based on patterns.
- The chatbot must employ the Microsoft Azure's multilingual component for enabling conversations in at least 2 languages other than English.

2.2.2. Should Haves:

- The chatbot should have a welcome message for a customer.
- The chatbot should make use of the best component for answering the user's query and should answer it as appropriately as possible.
- The chatbot should enable the user to upload a single image/ multiple images based on his/her choice during prediction.
- The chatbot should maintain a general etiquette while conversing with a user.
- The chatbot should try to match a translated query to the best possible answer from the rule based or the similarity-based files.

2.2.3. Could Haves:

- The chatbot could have a user-friendly GUI which is easily navigable.
- The chatbot could save a backup of all the conversations with a particular customer in text form onto the disk.

- The chatbot could have a voice recognition (using MS Azure) feature that would enable a user to record and upload a query via a voice file.
- The chatbot could have a handwriting recognition feature (using MS Azure) that would allow the user to upload a handwritten query for recognition and it could be matched to an appropriate answer.
- The chatbot could have a sentiment analysis feature (using MS Azure) that would enable it to analyse the sentiments from the reviews of the customers. This would prove to be vital for improving the chatbot facility.

2.2.4. Won't Haves:

- The chatbot won't have a video-processing feature.
- The chatbot won't have a timeout feature.

3. EMPLOYED AI TECHNIQUES

3.1. Rule-Based Component

For the rule-based component of the chatbot, it uses a set of pre-defined rules contained within the Dahlia.aiml file. The rules serves as the problems for the chatbots with which it can deal with and provide an appropriate solution. For example: If the user queries "I WANT TO PURCHASE FLOWERS" → this becomes the problem statement for the chatbot to which the solution in our case would be "Great, what king Roses, Lilies or Marigolds?". This component cannot answer any questions(problems) which are outside these set of rules.

3.2. Similarity-Based Component

For the similarity-based component of the chatbot, it uses a number of techniques: Bag of Words(BOW) model, tfidf and cosine-similarity for matching the most similar answer to the user's query and presenting that as a solution.

3.2.1. Bag of Words(BOW):

The BOW model transforms any random text to a fixed-length vector by counting the number of instances for which each word appear in it. This is also termed as vectorization.

Example:

1. This is a rose.

2. This is a red rose.

3. This is a red rose in a pot.

The words found from the above 3 sentences are: this, is, a, rose, red, in, a, pot. Vectorization table:

Document	this	is	a	rose	red	in	pot
This is a rose.	1	1	1	1	0	0	0
This is a red rose.	1	1	1	1	1	0	0
This is a red rose in a pot.	1	1	2	1	1	1	1

Now, we can see that we a 6-lengthed vector for each document. Hence, NOW literally tells one what words occur and not where they occur.

3.2.2. TFIDF(Time Frequency Inverse Document Frequency):

First we take the tokenize the given document.

Word	Count
a	4
this	3
is	3
rose	3
in	1
pot	1

Next, we find the tf:

TF = (Number of repetitions of word in a document) / (# of words in a document)

Then, we find the idf:

IDF = Log[(Number of documents) / (Number of documents containing the word)]

Next, we stack all the words next to each other. We can see the percentile equivalent to a word. Let's say in our case the words "this", "is", "a", "rose" has the highest percentiles then they form the most important words and the chatbot will provide an answer that contains most of these 4 words.

3.2.3. Logic Behind Cosine-Similarity:

The cosine similarity is beneficial because even if the two similar data objects are far apart by the Euclidean distance because of the size, they could still have a smaller angle between them. Smaller the angle, higher the similarity.

The formula for cosine similarity is as follows:

$$\mathrm{Cos}(x, y) = x \cdot y / ||x|| * ||y||$$

X and y here are the vectors. In this case vectors are the vectorized sentences as shown above in the BOW model section. The lesser the angle the higher the similarity and the more significant are the words which are used for providing answers by the chatbot.

3.3. Image Classification:

Convolution Neural Networks (CNN) is a special computer neural network architecture used for classifying images. In image classification, an image is passed through a series of convolutional, nonlinear, pooling layers and fully connected layers and then an output is generated.

The convolution layer always comes first.

Imagine that the reading of the input matrix begins at the top left of image. Next the software selects a smaller matrix there, which is called a filter (or neuron, or core). Then the filter produces convolution, i.e. moves along the input image. The filter's task is to multiply its values by the original pixel values. All these multiplications are summed up. One number is obtained in the end. Since the filter has read the image only in the upper left corner, it moves further and further right by 1 unit performing a similar operation. After passing the filter across all positions, a matrix is obtained, but smaller then a input matrix (Ksenia Sorokina, 2017).

The nonlinear layer is added after each convolution operation. It has an activation function, which brings nonlinear property. Without this property a network would not be sufficiently intense and will not be able to model the response variable (as a class label) (Ksenia Sorokina, 2017).

The pooling layer follows the nonlinear layer. It works with width and height of the image and performs a down sampling operation on them. As a result the image volume is reduced. This means that if some features (as for example boundaries) have already been identified in the previous convolution operation, than a detailed image is no longer needed for further processing, and it is compressed to less detailed pictures (Ksenia Sorokina, 2017).

After completion of series of convolutional, nonlinear and pooling layers, it is necessary to attach a fully connected layer. This layer takes the output information from convolutional networks. Attaching a fully connected layer to the end of the network results in an N dimensional vector, where N is the amount of classes from which the model selects the desired class (Ksenia Sorokina, 2017).

The activation function of this model is Relu. This function setts the zero threshold and looks like: $f(x) = max(0,x)$. If $x > 0$ — the volume of the array of pixels remains the same, and if $x < 0$ — it cuts off unnecessary details in the channel (Ksenia Sorokina, 2017).

Max Pooling 2D layer is pooling operation for spatial data.

After three groups of layers there are two fully connected layers. Flatten performs the input role. Next is Dense — densely connected layer with the value of the output space (64) and Relu activation function. It follows Dropout, which is preventing overfitting. Overfitting is the phenomenon when the constructed model recognizes the examples from the training sample, but works relatively poorly on the examples of the test sample. Dropout takes value between 0 and

1. The last fully connected layer has 1 output and Sigmoid activation function (Ksenia Sorokina, 2017).

Binary cross entropy loss function shows the sum of all individual losses.

Batch size the number of training examples in one forward/backward pass (or for 1 epoch, which is expected).

The flow_from_directory(directory) method is added for training and testing data. First, the path to the folders is specified. Further, the target size follows. It shows width and height to which images will be resized. Next, the batch size is added. Finally binary class mode is set (Ksenia Sorokina, 2017).

3.4. Logic-Component

query of the user based on the logic using the input pattern, then the chatbot replies “Correct” else the chatbot replies “Incorrect”.

To site an example:

The rule-based .aiml file has 2 patterned rules “I KNOW THAT * IS *” and “CHECK THAT * IS *”.

Here the first ‘*’ corresponds to the object and the next corresponds to the subject in a sentence entered by the user.

Let’s say that the user has entered “I KNOW THAT ROSE IS RED” and then “CHECK THAT ROSE IS WHITE”.

The logic defined in the .csv file is as follows:

RED(X) -> -WHITE(X) → implies that Red (something) is not equivalent to White(something). In this case the words “RED” and “WHITE” are the subjects to which x is an input given by the user which is the object and in our case it is “ROSE”. Since the user has entered that Rose is Red and it had matched with the pre-defined logic, the chatbot has remembered it. However, if the user enters “CHECK THAT ROSE IS WHITE”, the bot replies back “Incorrect” as based on the logic a red rose is not equivalent to a white rose. Hence, it was wrong.

3.5. Multilingual Component:

Natural Language Processing (NLP) is used for translating one language into another using computers. In order to make sense from a piece of text in one language, this piece is disintegrated into paragraphs, sentences and then to units which are converted into numbers. These numbers are converted back to words that make up another language. In order to achieve this process, 3 main algorithms are needed:

1) **Preprocessing:** Tokenization and UNK Replacement
2) **Word Embeddings:** Vectors and Dimension Reduction
3) **Sequence to Sequence:** Encoding and Decoding (Zhao, 2020).

Some of the vital steps that are involved in language translation are described below:

Preprocessing Stage: Takes place during the start of machine translation. Text gets converted to raw data. 2 major process is involved: Tokenization and UNK replacement. This stage might convert all the words to lower case so that words that are same but are in uppercase are also treated same.

```
[ ]  1 tokenize_en("God is Great! I won a lottery.")

['God', 'is', 'Great', '!', 'I', 'won', 'a', 'lottery', '.']
```

Fig 1. Sentence tokenization (Anon, n.d.)

Tokenization: Disintegrates big data in form of paragraphs or sentences into "tokens" which are units of languages. 2 extra tokens are used, one at the Start of a Sentence (SOS) and the other at the End of a Sentence (EOS).

UNK Replacement: If there are words used in a program that the machine cannot understand, then it replaces them with UNK which is the abbreviation of Unknown. These missing words are filled up later during post-processing.

Word Embeddings: Vector representation of the text. Each sentence is assigned a series of numbers, also known as a vector, with an arbitrary number of dimensions. These dimensions help describe the words of a sentence (Zhao, 2020).

Sequence To Sequence: This model is used for translating a sentence or sequence of words. It has 2 major components: Encoder and Decoder. The Encoder converts the original text into a vector. Decoder converts this vector back to a human language but to that language in which we wanted to translate the original text to.

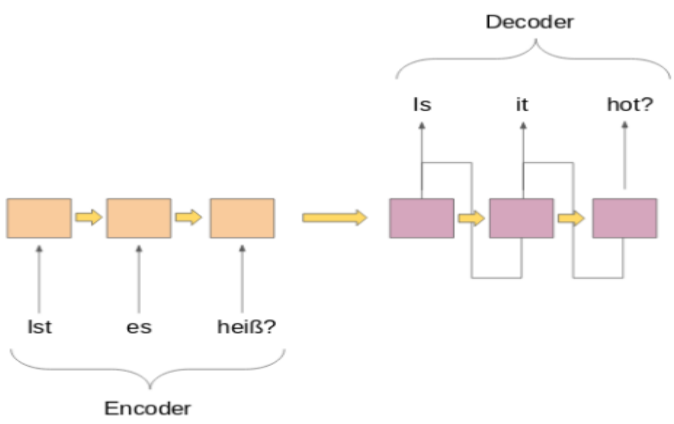

Fig 2. Encoder and Decoder

Neural Machine Translation (NPT): This concept is used for translating text in one language to another by training the computer with a large amount of data.

Datasets: A training dataset is provided to the program which contains the majority of the data for learning. Validation dataset is used for evaluating the model. This check if the model is working correctly. Test data is used for performing the final evaluation on the model.

The text that has been transformed into vectors are further assigned weights to the words in it.

```
"I    love  eating  toasted  cheese  and  tuna  sandwiches"
0.1   0.7   0.8     0.4      0.6     0.1  0.6   0.8
```

Fig 3 Giving word weightings(Zhao, 2020).

Weights allow NNs to understand the words which are more important than the others.

Parallel Corpus: Sometimes the machine is fed a series of sentences with their translations in a different language. The machine can utilize this data as a reference.

Teacher Forcing: A big problem in translation can be that a sentence or a word might depend on the context that has been established by the sentences or words preceding it. This might cause an error to propagate throughout the Neural Net. Teacher forcing algorithm feeds a reference word which is essentially a translated answer in the network that prevents this problem from occurring.

3.6. Speech recognition

Speech recognition has several categories under it based on the type of speech that a model is supposed to recognize.
Isolated word: has 2 states: Listen and Not-Listen. Accepts a single word utterance at a time.
Connected word: Closer to the Isolated word approach but in this case several words can be uttered with very little pause in between.
Continuous speech: Enables the user to speak naturally.
Spontaneous speech: A number of words can be run together during utterance.

There are 4 stages to speech recognition:

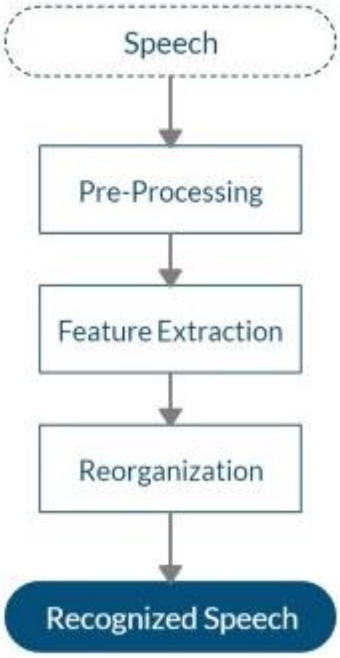

Fig 4 Four stages of speech recognition(Senaratne, 2020).

Speech Capturing: Speech can be recorded using a microphone. This analog signal then needs to be converted to a digital sound using sound card.

Pre-Processing: Has a number of steps:
1) Background Noise and Silence Removing
2) Preemphasis Filter
3) Blocking into Frames
4) Windowing (Senaratne, 2020).

Feature Extraction: Transforms the speech into feature vectors. A digital filter like Fourier transformation or Linear predictive coding is used for extracting these feature vectors.

Reorganization: Has 2 parts: training and testing part. Training part is that in which the system experiences and learns the speech. During training, system considers an unknown speech signal to reference patterns that are closest to the recognized word.

3.6.1. Models of Speech recognition Systems

1. Dynamic Time Wrap (DTW)

DTW is a technique that can find an optimal match between two given sequences of speech. And also this method facilitates non-linear mapping from one signal to another signal by reducing the distance between the two signals. DTW is a template-based method and also to understand this, there are two concepts to deal with (Senaratne, 2020).

1.1. Symmetrical DTW

Speech always depends on time because it is time-dependent. There are several ways in pronouncing the same word that can have different time durations and also utterance of the same word with equal duration will deviate from its middle. There are different spoken rates due to different part of the word.

- Matching paths cannot reverse in time.
- All frame in the input must be used in matching point.
- Global distance is obtained by combining local distance scores.

This is known as Dynamic Programming (DP). DP can find the minimum distance path through the matrix and also can reduce the amount of calculation (Senaratne, 2020).

1.2. Asymmetrical DTW

In this approach, the input pattern which is in each frame is used only one time. It means that dispense and template length normalization and for diagonal transition no need to add local distance twice. This method is used to as asymmetric DP (Senaratne, 2020).

2. Hidden Markov Model (HMM)

Speech can be recognized mathematically using this approach. This is a doubly embedded stochastic (having a random probability distribution or pattern that may be analyzed statistically but may not be predicted precisely) process with cannot directly observable (hidden) stochastic process. But this process can be observed from another stochastic process which gives a sequence of observations. The main idea behind HMM is that the set of hidden states has Markov dynamics. The model is defined by using two sets of parameters, the transition matrix whose ij element is P(ut+1=j | ut=i) and the emission matrix whose iq element is P(at=q | ut=i).
By using the probabilistic model, stochastic modeling deals with incomplete and uncertain data/ information. Incompleteness and uncertainty are occurred in speech recognition, for example, speaker variability, contextual effect, confusable sounds, homophones words, etc. (Senaratne, 2020).

3.7. Handwriting Recognition

The major technique involved in handwriting recognition are described below:

Neural Network System for Continuous Handwritten Word Recognition: A method for continuous handwritten word recognition is derived when the word is segmented into triplets (containing 3 letters). Two subsequent triplets have 2 common letters. The biggest challenge for recognition systems is to perform operations on a continuous word. In this, each word is subdivided into triplets, each containing three letters. Figure 10a shows triplet "aba" and figure 10b shows triplet "ban". Two neighbour triplets always contain two common letters which represent the overlapping between letters. This kind of overlapping results is a higher recognition rate (Learning, 2020).

3.8. Sentiment Analysis

For Sentiment Analysis, Natural Language Processing (NLP) is used for taking a piece of text and then extracting vital words from it. Then it uses an algorithm to classify words that have a positive or a negative sentiment attached to them For example: words such as "terrible" and "bad" has a negative emotion attached to them while words such as "good" or "helpful" or "nice" has a positive emotion attached to them. The presence of these words are checked in a text. Next, it is checked if the presence of negative words are or more than the positive words or is it the vice versa. If negative words has a greater presence than positive words, then the text is labelled to have a negative emotion attached with it on the whole while if the text has a greater presence of positive words then it is classified as a positive statement.

4. USES OF IMPLEMENTED AI TECHNIQUES IN REAL WORLD

The AI techniques that has been implemented in building this chatbot is of high importance in real world. This section provides a brief description on the usage of these techniques in real-life:

Rule-Based and Similarity-Based component: This is of vital importance to online companies especially those which are related to e- commerce. The chatbot that I have created itself is themed to have a e-commerce look to it. A lot of E-commerce sites tend to depend solely on chatbots for dealing with their customers. In order for them to have a good relationship with the customers, it is necessary for them to make sure that the bot understands and provides an accurate answer to the customer. Common questions related to a site can be defined in a rule-based model while questions related to descriptive answers can be defined in a similarity based model.

Image Classification: It is an added advantage if the chatbot can understand a particular product that the customer wants by identifying it through it's image and then the bot can let a customer know if this product would be available in the store or not.

Multilingual Component: Every company today wants to expand their reign in a global market. However, this is not an easy task because attracting a global market involved engaging customers of different ethnicities who speak in different languages. Having a chatbot that can converse with customers in different languages can definitely make a company stand out from all other companies and help to accelerate it's business to a larger extent.

Speech recognition: Popular messaging applications like Messenger and WhatsApp utilizes this technique. One person can send a voice message to another one if they are in a hurry or they are in a situation in which typing becomes pretty difficult (like going up a fleet of stairs!). This feature can definitely prove to be pretty effective with a lot of people. Apart from that recording a speech takes lesser time than typing. Many sectors of business like healthcare and retail uses speech recognition for providing a better customer service. Also, it saves money on the company's part as a bot that can recognize speech replaces live agents who would be needed to be paid.

Handwriting recognition: Applications of offline handwriting recognition are numerous: reading postal addresses, bank check amounts, and forms. Furthermore, OCR plays an important role for digital libraries, allowing the entry of image textual information into computers by digitization, image restoration, and recognition methods (www.sciencedirect.com, n.d.).

Sentiment Analysis: Sentiment Analysis is very important for almost every company today. Sentiment Analysis is used for identifying the feedbacks of customers which in turn can help companies to improve themselves.
Sentiment analysis can identify critical issues in real-time, for example is a PR crisis on social media escalating? It’s estimated that people only agree around 60-65% of the time when determining the sentiment of a particular text. Tagging text by sentiment is highly subjective, influenced by personal experiences, thoughts, and beliefs. By using a centralized sentiment analysis system, companies can apply the same criteria to all of their data, helping them improve accuracy and gain better insights (MonkeyLearn, n.d.).

5. GENERAL OVERVIEW OF THE CODE

This section provides a general overview of the code that has been used for implementing the AI techniques previously mentioned in the document.

Choice of IDE : Google Colaboratory(Colab) has been used as the working environment for developing the code.

Pre-Installations: Certain libraries were needed installed for implementing certain tasks. This includes aiml (rule-based component) , certain features from azure cognitive services like text analytics (translations and sentiment analysis) , speech (speech recognition) and computervision (for handwriting recognition).

Azure keys: The Azure key, endpoint and region has been initialized for using them for implementing the Azure features later in the code.

Header files: All the header files that are necessary for running the program has been imported right after defining the azure keys and endpoints. The program uses keras with TensorFlow backend for Deep Learning, it also uses nltk for building up a knowledgebase for the similarity-component as well as the logical component of the bot. Other necessary libraries for ML and DL like sklearn, pandas and matplotlib has been imported along with others.

Image Classification: After importing the header files a sequential model has been created for image classification. To this sequential model a CNN layer, pooling layer, flattening layer and fully connected layers has been added. Then for the output layer, 11 neurons has been specified for 11 different classes of images. Then the training data generator has been used for preparing the train dataset and the test dataset utilising the flow from directory structure. After that the model is compiled. The optimizer is set to adam and categorical crossentropy has been used as the loss function. The number of epochs has been set to 30. A summary of this model has been generated and finally, this model has been stored in a .h5 format. It is later loaded back for prediction.

Rule and Similarity component: The system is made to learn the rules defined in the .aiml file using the kernel.learn function. Then the .csv file containing the knowledgebase is loaded for implementing the similarity component. This is then tokentized and lemmatized and all the stopwords are removed from it.

Logic component: The logic based .csv file is loaded next.

Operations

This section describes all the functions that has been used in the code.

A function named **response()** is defined which takes the **user_response** as a parameter. It tokenizes and lemmatizes the user response and finds out the cosine similarity of all these tokens with respect to the knowledgebase. If the statement is found to be similar to any other statements in the knowledgebase then it produces that statement as an answer to the user's response else it prints a statement stating it did not understand the user's query.

A function named **predict_product()** has been used which first initiates an operation that would require the user to upload an image file(.jpg) . The image classification model is then loaded and this image is compared to the index of the different classes used in the training dataset. If the image corresponds with the index of a class then that class is set to be the label of that image thus predicting it. The function **pred()** is used for multiple image classification where the user will have to upload a number of images. Colab stores multiple images in form of a dictionary. The keys of these images are sent to the image classification model chronologically for prediction.

The function **translate_text()** is used for translating one language to another. It takes language codes of the current language and the language to be translated to as an input and then utilises MS Azure's REST API for translating a given text to the required language. MS Azure's other features such as speech recognition, sentiment analysis and handwriting recognition has also been utilised inside the main loop.

A bunch of welcome statements is provided inside the print statements in order to welcome a customer to the site. Then the main loop is initiated which terminates only when a customer enters 'EXIT' as input. The above mentioned functionalities are implemented based on conditions which are essentially the input that an user provides when the program is running.

6. POINTS FOR FUTURE WORK

This project can be further expanded in future. I have mentioned some points that might be taken in account as a future work on this project:

- The chatbot would have a user friendly GUI attached with it giving it the look of a real-life app. As the chosen theme of this chatbot is that of a virtual assistant for an online flower decoration site, a mobile based application might suit it the best.
- The chatbot would have a feature that would enable the user to converse with it in real-time through an input device like a microphone.
- The chatbot would have a feature that would enable it to converse with a speaker in real time in a different language.
- The chatbot would be able to recommend other products to a user based on their previous purchases.

7. REFERENCES

- Ksenia Sorokina (2017). Image Classification with Convolutional Neural Networks. [online] Medium. Available at: https://medium.com/@ksusorokina/image-classification-with-convolutional-neural-networks-496815db12a8.
- GeeksforGeeks. (2020). Cosine Similarity. [online] Available at: https://www.geeksforgeeks.org/cosine-similarity/ [Accessed 23 Feb. 2021].
- Madan, R. (2019). TF-IDF/Term Frequency Technique: Easiest explanation for Text classification in NLP with Python. [online] Medium. Available at: https://medium.com/analytics-vidhya/tf-idf-term-frequency-technique-easiest-explanation-for-text-classification-in-nlp-with-code-8ca3912e58c3.
- Zhou, V. (2019). A Simple Explanation of the Bag-of-Words Model. [online] Medium. Available at: https://towardsdatascience.com/a-simple-explanation-of-the-bag-of-words-model-b88fc4f4971.
- Zhao, S. (2020). *Natural Language Processing*. [online] Medium. Available at: https://medium.com/swlh/nlp-11ff6aeb8259#:~:text=In%20order%20for%20the%20computer [Accessed 27 Mar. 2021].
- Anon, (n.d.). Available at: https://medium.com/swlh/nlp-11ff6aeb8259#:~:text=In%20order%20for%20the%20computer,thus%20completing%20the%20translation%20process. [Accessed 27 Mar. 2021].
- Senaratne, R. (2020). The Promise of Natural Language Processing in Speech Recognition. [online] Medium. Available at: https://ai.plainenglish.io/the-promise-of-natural-language-processing-in-speech-recognition-dfdc4d276ba1 [Accessed 27 Mar. 2021].
- Learning, G. (2020). How To Apply Machine Learning to Recognise Handwriting. [online] GreatLearning Blog: Free Resources what Matters to shape your Career! Available at: https://www.mygreatlearning.com/blog/how-to-recognise-handwriting-with-machine-learning/ [Accessed 27 Mar. 2021].
- www.sciencedirect.com. (n.d.). Handwriting Recognition - an overview | ScienceDirect Topics. [online] Available at: https://www.sciencedirect.com/topics/computer-science/handwriting-recognition#:~:text=Applications%20of%20offline%20handwriting%20recognition [Accessed 27 Mar. 2021].
- MonkeyLearn. (n.d.). Everything There Is to Know about Sentiment Analysis. [online] Available at: https://monkeylearn.com/sentiment-analysis/#:~:text=Sentiment%20analysis%20is%20extremely%20important [Accessed 27 Mar. 2021].