Chatbot Using RNN

Jeevan Reddy Gopu
Lakehead University
Student ID: 1105340
jgopu@lakeheadu.ca

Vishnu Sudheer Gannamani
Lakehead University
Student ID: 1110200
vgannama@lakeheadu.ca

Prashanth Kumar Daram
Lakehead University
Student ID: 1116540
pdaram@lakeheadu.ca

Akhil Kumar Bandaru
Lakehead University
Student ID: 1105900
abandaru@lakeheadu.ca

***Abstract*—The purpose of the chatbot is to make a conversation between both humans and machine. Chatbots were initially made using the concepts of machine learning with Natural Language Processing (NLP) and it had some flaws in it. In order to overcome those problems, we are using some deep learning techniques with NLP to create an advance chatbot. Chatbots are the latest technologies created for the modern world. Chatbots uses the concept of NLP to communicate and share data. The way they are trained and the methods that are implemented to make the conversation sensible and reasonable is stated. The machine has been trained with certain knowledge to figure out the sentences and deciding by its own to generate a response to answer a question. The response should be matching to the input sentence from user. From the input sentence, it will perform certain calculations to get the similarity of sentences. This paper provides an overview of some modern deep learning techniques that were used to build the chatbots. The methods of Recurrent Neural Network (RNN) and Long short-term memory (LSTM) are implemented in our model to generate the outputs.**

keywords*—*Chatbot, RNN, LSTM, Deep Learning, batch size, epochs, Input layer, Embedded layer.

I. Introduction

Building a perfect conversation agent in this modern world is still lacking and it can be considered as one the drawback, and lot of work is to be done in the field of artificial intelligence, Mobiles and other gadgets that became key to humans. They give assistance to humans when we need something. The insights that we draw upon should be accurate in this fast world driven by the data. Conversation is key to share and exchange data and it occurs from person to person and machine to machine. Conversation agents have been changing since many years as per technology. Chatbots can be worked well until the user diverts them from their actual usage. The machine is linked with huge loads of data that needs larger space and methods.

Chatbots are used to deal with basic human interactions and its primary purpose is to return back to the user what they are looking for. It's like answering a question given by the user. Chatbots are feeded with huge amounts of data. This paper mainly describes the tools that help in enhancement of the functionality of the chatbot. Deep learning and information extraction can be considered as a key factor for chatbots. In this paper we want to create a chatbot with deep learning methods.

Chatbots can be easily defined as Help Desks. They have the capability to reply an answer related to user's question. Chatbots now became a trending tool in the industry for the customer satisfaction. Questions are be pre-setup by the company related to the previous trends and general patterns. Chatbots provides an articial assistance to the user. Now-a-days most people are using chatbots to do their basic things such as booking a flight and many more. Chatbots provides us service in both voice and textual method. The working of a chatbot mainly depends on the user inputs. There are many flaws that are present in the chatbots. So to overcome those flaws we developed a robust chatbot with the help of LSTM.

II. possible Challenges

- Building or creating can be considered as one of the key factor.
- Developing Cost should also be considered.
- How to reach it to the audience and making it popular.
- To decide to do a voice or textual chatbot.
- Training it with data and making it to understand the human language.
- There will be some limitations with the natural language.

III. Earlier Works

The idea of chatbot [1] originally originated in MIT by Weizenbaum in the year 1996, where he created a chatbot called ELIZA to emulate psychotherapist. The idea is based on keyword matching and it is very simple. The input will be checking for the presence of the keyword. If keyword is found, then the sentence will automatically be mapped according to the keyword. If the keyword is not mapped, then earlier transformation is retrieved. For example, if the input contains word father then ELIZA asks us to tell me more about the family because mother and family are related to the psychological problems. Later many other Chatbots [2]

978-1-7281-4387-3/19/$31.00 © 2019 IEEE

like MEGAHAL in 1996, CONVERSE in 1999, ELIZABETH in 2002, HEXBOT in 2004 and ALICE in 2002 have been developed.

IV. Literature Review

We have chosen two different papers in which the authors have worked to build the chatbots and we would like to explain about the techniques that are implemented in it and limitations in them. Each has its own ideas in implementation and working.

Google AI team [1] has developed a chatbot recently in 2019 by a group of members, Meena was its name and it used deep learning algorithms involving end to end connected neural networks. It contains 2.6 billion parameters to generate a solution. A sensible and meaningful chatbots are created in order to make conversation. Meena was so sensible in giving reply to the user's question. The Sensibleness and Specificity Average (SSA) catches the important attributes for the human conversation. Meena was able to achieve the SSA score of 76% which is so better compared to other chatbots. In the previous models all the evaluation metrics were complicated and the final output predicting was comparatively a tedious task. The SSA considers basic and most valued features. The number of layers used while training the model are high. The figure 1 shows results of Meena with other chatbots.

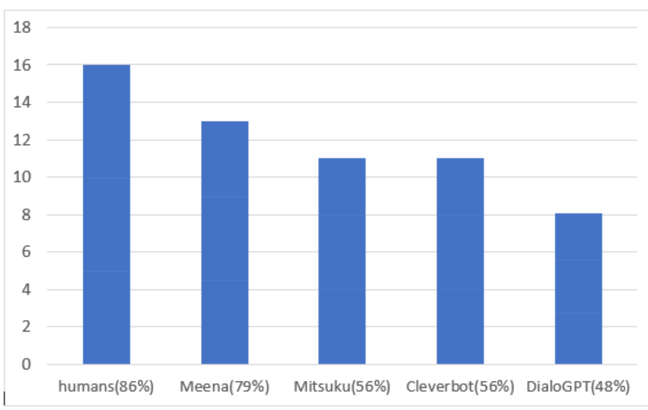

Fig. 1. Comparision results of meena with other chatbots [1].

The major goal [2] was building a socialbot which means a spoken conversational agent that has capablility of conversing coherently and engaging with the human on more popular social topics such as entertainment, fashion, politics, sports, and technology. The best deep learning based socialbot is MILABOT.

The dialogue systems which were developed earlier was mainly based on states and set of rules which are hand-crafted by human experts. Some of the modern dialogue systems followed hybrid architecture which is combination of both hand-crafted states and machine learning algorithms. The MILABOT is built by using only statistical machine learning algorithms. Here, none of the human-crafted states are not considered. The system components will be trained independently on large data sets and then they are jointly trained on real-world user interactions. In this way, the system will learn all relevant states and rules for conducting open-domain conversations implicitly. Given an adequate number of examples, such a system should outperform systems based on hand-crafted states and rules.

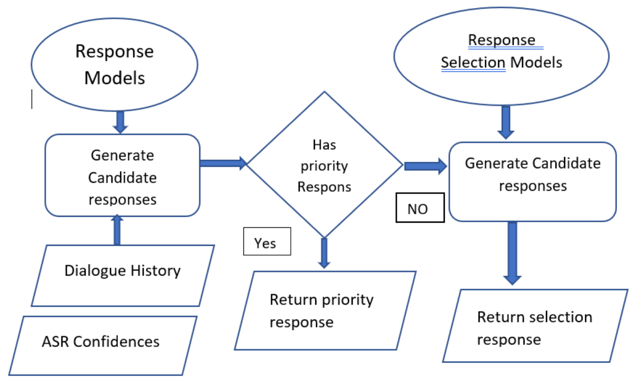

Fig. 2. Dialouge manager control flow [2].

Model responses are combined by the Dialogue Manager. The dialogue manager follows the below three steps .
• All response models are used to create candidate responses.
• If priority response exists it will be returned by the system.
• If priority responses does n't exist, then response is selected by model selection policy.

V. Existing Model

In Existing model, LSTM was used in order to generate a chatbot. They have used the Bag of words, which is used to extract only unigram model of words. So that we can create unordered list of words without POS tagging, semantic and syntactic. The model consists of one input layer, one embedding layer, one LSTM layer, one dense layer with soft-max activation function. This model generated an accuracy around 95%.

VI. Proposed Model

The Main Steps that are involved in the Model:

- **Pre-setup (Importing all the required packages)**
- **Dataset Processing**
- **Network Creation**
- **Training the model**
- **Testing the model**

The proposed model is based on implementation of RNN using LSTM for generating the output. We need to Load the movie review dataset into Google Colab. Once dataset is loaded, we need to perform the text normalization and data pre-processing steps and then we will be creating a customized model from the scratch using RNN. Further, we will be dividing the dataset into training and testing. In this model we are dividing dataset into 80% for training and 20% for testing. we will be creating a customized method to train our new model in batches and let it run for few epochs. At last, we need to evaluate the model based on the testing dataset to see its performance.

A. *Pre-Setup*

Firstly, the coding part has done on the CO-LAB platform which is provided by the Google. The Google CO-LAB usually uses GPU because it has a very fast processing time. Secondly, import all the required libraries that are necessary to work with the model. We use pandas library to read our dataset. After loading the data set, we need to perform the Data Pre-processing steps.

B. *Data Pre-processing*

Text Normalization: Data Pre-processing consists of five main components: dictionary mapping, tokenization, stop words removal, punctuation removal, stemming and lemmatization. Tokenization is an important task in NLP. After text was segmented into words, every word will be mapped with an integer by the dictionary index. We will be getting the length of the list of integers which is equal to number of words in the given text.

Data splitting: we split the dataset into training and testing portions. We are splitting the dataset into 80% for training and 20% for testing. Next, the data is placed into the data frames to get numpy arrays out and then we perform Vectorization. We used Tfidf Vectorizer to convert our text reviews into vectors which is the size of our entire vocabulary.

C. *Network Creation*

The classification model is of type neural network which takes input from the pre-processing to categorize the questions. The model contains three layers. Firstly, the embedding layer where each word is mapped to the real number vectors in order to define the fixed size vocabulary of the text. Secondly, the LSTM layer is a fixed kind of RNN, which has capacity of learning the sequential data. The LSTM layer enables the RNN to memorize the input data for long amount of time. Finally, the output layer with the softmax activation is used in multiple number of multi-class classification methods. The softmax activation function presented in the output layer defines the categorical distribution across the labels of the class that helps in obtaining the probabilities of each input that belongs to each label.

D. *Training and Testing the model*

To train the model, we are using the adam optimizer and binary_crossentrophy loss. The performance of the model is measured using the accuracy metric. Finally the model is tested.

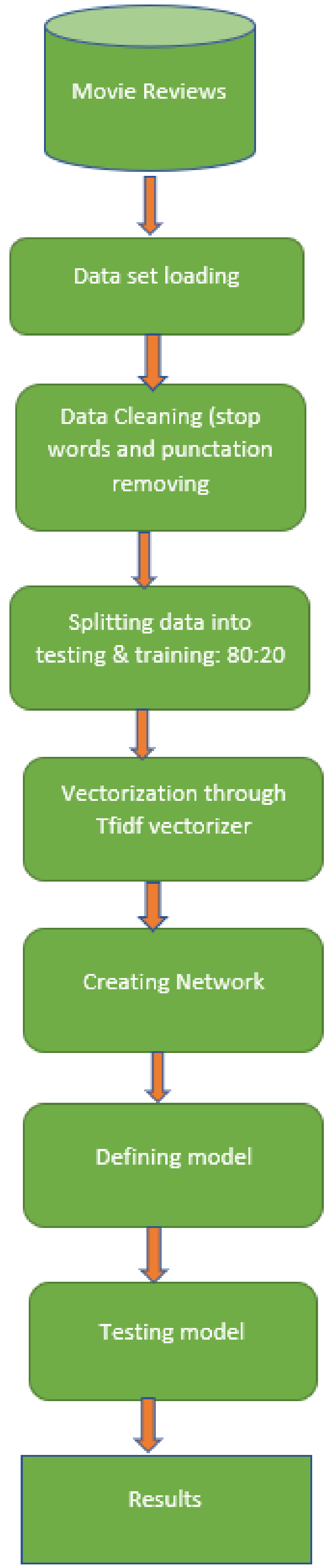

Fig. 3. Flow chart diagram of the model [3].

E. Hyperparameters tuning

Hyperparameters are tuned in optimizing the model. They also help in finding the closest value of hyperparameter that gives the highest performance. We tuned three hyperparameters that include 1) embedding dimension, 2) epochs, and 3) batch_size. We use a different random searches using different values of hyperparameters. We keep on performing the both rescaling and tunning until the performance of the model improves.

VII. Experiment and Evaluation

Model 1:
In model 1, We used two input layers, two embedding layers, two LSTM layers and one dense layers. Firstly, the model was trained with batch_size equal to 64 and epochs equal to 5. Secondly, the same model was trained with batch_size equal to 128 and epochs equal to 5.

TABLE I
Model 1 results.

Model 1			
Epochs	Batch Size	Loss	Accuracy
5	64	0.0026	0.995
5	128	0.0030	0.986

Model 2:
In model 2, We used two input layers, three embedding layers , three LSTM layers and one dense layer. Firstly, the model was trained with batch_size equal to 128 and epochs equal to 10. Secondly, the same model was trained with batch_size equal to 256 and epochs equal to 10.

TABLE II
Model 2 results.

Model 2			
Epochs	Batch Size	Loss	Accuracy
10	128	0.0035	0.975
10	256	0.0038	0.968

Comparison of All 2 Models: The table 3 shows all the two models scores. In all two models, model 1 is better than other two models. It have accuracy and other metrics more than other two models. So, model 1 was the final model.

TABLE III
Comparision of models.

Comparison of Models			
Epochs	Batch Size	Loss	Accuracy
5	64	0.0026	0.995
10	128	0.0035	0.975

Graphs: The graphs of model 1 accuracy and loss are shown in figure 4 and figure 5.

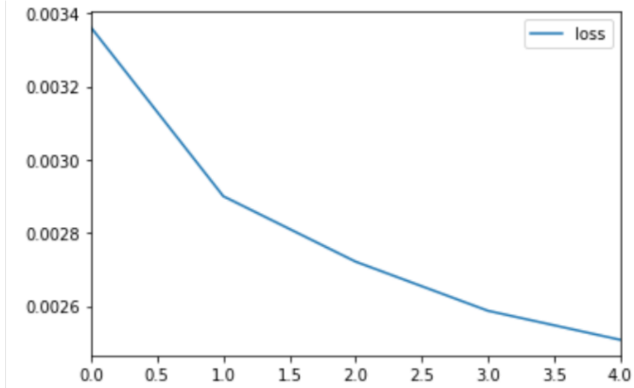

Fig. 4. Model 1 Loss v/s Epoch graph.

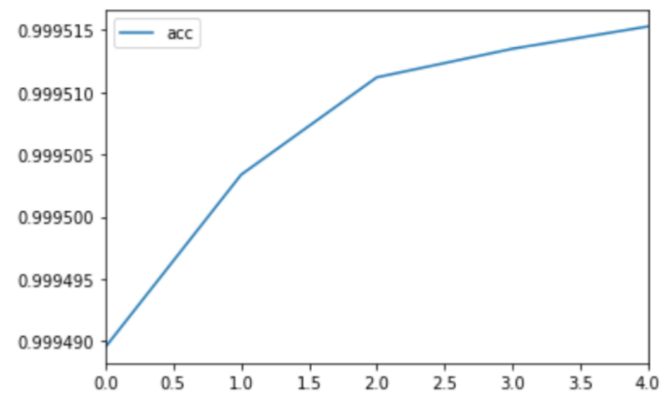

Fig. 5. Model 1 Accuracy v/s Epoch graph.

VIII. Conclusion

We have developed a chatbot with LSTM using RNN model. Our chatbot takes the questions and provides the answers in the text format. The existing model have used Bag-of-words for data pre-processing where the accuracy is around 95%. In the proposed model, TF-IDF have been used where the accuracy is around 98%. The results are shown in table 4.

TABLE IV
Final results.

Epochs	Batch Size	Loss	Accuracy
5	64	0.002	0.995

GitHub Link:
https://github.com/jeevanlakehead/new.

IX. Contribution

The key contributions of the team mates are listed below

Jeevan Reddy Gopu- Pre-Processing dataset.
Akhil Kumar Bandaru- Implementing existing model.
Vishnu Sudheer Gannamani- Creating proposed model.
Prashanth Kumar Daram- Comparing results and reporting.

REFERENCES

[1] Naz Albayrak, Aydeniz Ozdemir, Engin Zeydan, ”An overview of artificial intelligence based chatbots and an example chatbot application”, SIU, 2018.

[2] Alessandro Bozzon, ”Enterprise Crowd Computing for Human Aided Chatbots”, SE4COG, 2018.

[3] Ramya Ravi, ”Intelligent Chatbot for Easy Web-Analytics Insights”, ICACCI, 2018.

[4] Wen Zhang, Heng Wang, Kaijun Ren, Junqiang Song, “Chinese sentence based lexical similarity measure for artificial intelligence chatbot”, ECAI, 2016.

[5] Neelkumar P. Patel,Devangi R. Parikh, Darshan A. Patel, Ronak R. Patel, “AI and Web-Based Human-Like Interactive University Chatbot (UNIBOT)”, ICECA, 2019.