Chatbot Using RASA

Nikhil Mishra, Yashvi Desai, Manish Katheeth, Chinmay Tawde
Address 1
Nikhilmishra2498@gmail.com
Address 2
Desaiyashvi98@gmail.com

Abstract. Chatbot is an Artificial Intelligence software that is designed to converse with humans through messaging or mobile applications, web-based applications.These chatbots are usually existing for banks, hospitals, colleges, etc but we have decided to construct our bot for Restaurant service. Since we people are crazy for new varieties of cuisine and also the influence of social media on us is another factor of the increasing hype of outside food. So by using our bot,we will not only guide you for the cuisines you are looking for or the nearby restaurants but also you can try and make reservations for the same. So in this guiding chatbot system the user has to type in their query or command and the proposed system then by using Natural Language Processing (NLP) engine will search in the database. After the research, the user will be provided with an answer or an approval for completion of the task. All this takes place within seconds so the user gets an immediate response to the query without wasting any time.

1 Introduction

Conversational systems are becoming eminent as a basis for human computer interaction as we seek more natural ways to integrate automation into everyday life. Some well-known examples of conversational AI include Siri, Alexa, Cortana, where the common tasks include scheduling meetings, booking flights, and customer support tasks. Chatbots are typically used in dialogue systems for various practical purposes including customer service or information acquisition.

Our Chatbot is built for Restaurant specific tasks. This platform provides users with different features such as like finding the near-by restaurant, suggesting cuisine-wise restaurants, sending details on provided mail id, booking a table, etc. The bot uses Natural Language Processing (NLP) engine to search for the queried message in the database and provides the user with the best solution.

We have used RASA to build our chatbot. Rasa is an open-source conversational AI framework which provides a set of tools to build a complete chatbot. In Rasa there is no pre-defined model so it is easy to customize. The user can build,deploy or host Rasa in his own server or environment with complete control on it.

2 Related Work

2.1 Sequence to Sequence Learning with Neural Networks by Ilya Sutskever, Oriol Vinyals, Quoc V. Le

Multilayered LSTMs are used to map input sequence to a vector of fixed dimensionality followed by another deep LSTM to decode the target sequence from the vector. The results suggest that the approach of using multilayered LSTMs will likely do well on other challenging sequence to sequence problems. Although, It was found that LSTMs trained on reversed dataset had little difficulty translating long sentences.

2.2 End-to-end Adversarial Learning for Generative Conversational Agents by Oswaldo Ludwig

Neural language modeling uses RNNs to create effective models for different tasks in NLP, such as open domain dialogue generation which aims at generating meaningful and coherent dialogue responses given the dialogue history. This paper is about generative models, which generate new responses from scratch, usually based on the sequence to sequence modeling. The Problems encountered were sometimes chatbot couldn't retain the context properly and sometimes couldn't change the subject, When human changed it.

2.3 A Neural Conversational Model by Oriol Vinyals & Quoc V.Le

In this research paper,we present a conversational model that uses Seq 2 Seq framework,which is based on RNN.It reads input sequence, one token at a time and predicts the output sequence,one token at a time.During training,the output sequence is given back to model,so that better learning can be done through back propagation. The constructed model requires a substantial modification to be able to deliver realistic conversations. It also lacks a coherent personality. (i.e human touch)

2.4 Rasa: Open Source Language Understanding and Dialogue Management by Tom Bovklisch, Joey Faulker, Nick Pawlowski & Alan Nichol

This paper serves as an introduction to RASA's pair of tools, RASA NLU and RASA Core. RASA NLU is used for natural language understanding, while RASA Core is used for dialogue management.

3 METHODOLOGY

3.1 Architecture of RASA

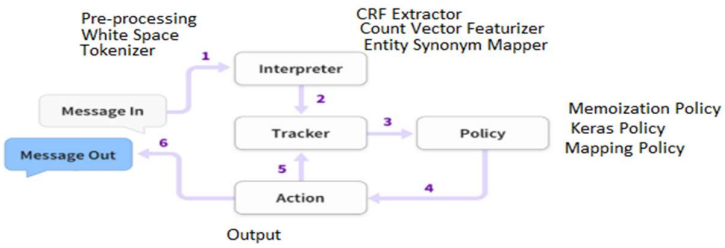

Fig. 3.1. Architecture of RASA (Source: rasa.com)

3.1.a Input data

'Message in' or the input data is the data fed to the bot by the user.This data can be a user created data or a prebuilt dataset.An input data also called as utterance ccomprises of 2 major components intents and entities.Intents are the intentions of the end user and enities highlights the objects or the terms that are relevant to the detected intents.

3.1.b Interpreter

Interpreter is a part of RASA NLU. It implements Natural Language Understanding and translates the message into structured output, i.e. intents and entities.Interpreter consists of a pipeline and a training data.A Pipeline is a cluster of components that are excuted sequentially in order to extract the intents and entities from the utterance.
Pipeline
.

```
# Configuration for Rasa NLU.
# https://rasa.com/docs/rasa/nlu/components/
language: en
pipeline:
- name: WhitespaceTokenizer
- name: CRFEntityExtractor
- name: CountVectorsFeaturizer
- name: EmbeddingIntentClassifier
  epochs: 50
- name: EntitySynonymMapper
# Configuration for Rasa Core.
# https://rasa.com/docs/rasa/core/policies/
policies:
  - name: MemoizationPolicy
  - name: KerasPolicy
  - name: MappingPolicy
```

Fig. 3.2. Pipeline and Policies used in the bot.

WhitespaceTokenizer: This component segments the input string using whitespace and returns the segments as tokens. It uses a simple *split()* function in order to segment the string.

CRFEntityExtractor: The tokenized words are tagged using Inside-outside-beginning (IOB) and Part-Of-Speech(POS)tagging. POS tags are used for assigning words in a text corpus to their corresponding part of speech. A chunker is used to chunk the annotated data on the basis of a specific tag. NamedEntityRecognition is the process of locating and identifying named entities present in a text and classifying them based on pre-defined categories. The NER process is conducted using a Conditional Random Field model.

CountVectorsFeaturizer: CountVectorFeaturization is also known as one-hot encoding, which helps in converting text data into numbers. Another method to do it would be Bag-of-Words. Input from the user is termed as a document. Next, the number of times a word occurs is counted. This is accomplished by a library in Python named Sci-kit learn. In our case words like a restaurant, cuisine, table, location, etc. would be repeated but along with these, there would be common words like the, and, of which are of less priority. So, to know the importance of each document TF-IDF (Term Frequency - Inverse Document Frequency) is used. The number of occurrences of a word is directly proportional to the TF-IDF value.

TF indicates the total number of occurrences of a word appears in a document per the total number of words in the document. Whereas, IDF indicates the weight of a given word possesses. More weight more is the significance of the word.

EmbeddingIntentClassifier: Word embedding is again a representation of vector by converting word text into numbers, but here it is used for similar words or similar meaning. So to measure this similarity between 2 vectors, Cosine similarity is used

EntitySynonymMapper: Maps synonymous entity values to the same value. If the training data contains defined Synonyms this component will make sure that detected entity values will be mapped to the same values..

3.1.c Tracker

The Tracker is the object which keeps track of conversation state. It receives the info that a new message has come in. Then based on dialog model we generate using domain and stories policy chooses which action to take next. The chosen action is logged by the tracker and response is sent back to the user.[8]

3.1.d Policies

Policies are responsible for optimum selection of actions which helps the bot to respond the user's request. There are various types of policies that are used when the input data satisfies a specific condition. Fig.2 shows the policies used in the bot.

MemoizationPolicy keeps a record of the conversations and stores it in the training data. If the conversations match with the training data then the bot correctly predicts the next action i.e. with a confidence of 1.0.

KerasPolicy uses a The KerasPolicy uses a neural network implemented in Keras to select the next action. The default architecture is based on an LSTM

MappingPolicy can be used to map intents directly to actions. The mapping is done by assigning trigger properties to a intent. The bot will perform the mapped action once it receives a message of the triggering intent. Standard prediction will resume for the next user message.

3.1.e Actions

Actions is a part of RASA Core. Actions are operations that the bot conducts in response to the user's input. RASA Core predicts which action to take from a predefined list. There are three kinds of actions in Rasa Core namely *default actions, utter actions ,custom actions.*

```
responses:
  utter_my_message:
    - "this is what I want my action to say!"
```

Fig. 3.3 ActionUtterTemplate (Source: rasa.com)

In this bot, Utter Actions are used. Fig .3 is an example of Utterance template. Conventionally an Utterance Action starts with 'utter_' and send a specific message to the user.

3.1.f Message out

The bot produces the appropriate response corresponding to the user's request.

4 Results

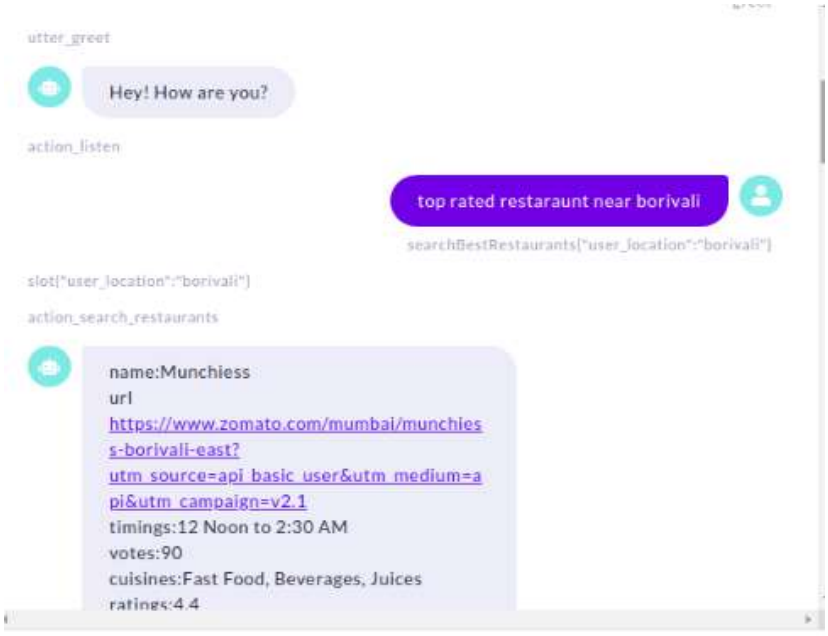

Fig. 4.1. Conversation with the bot

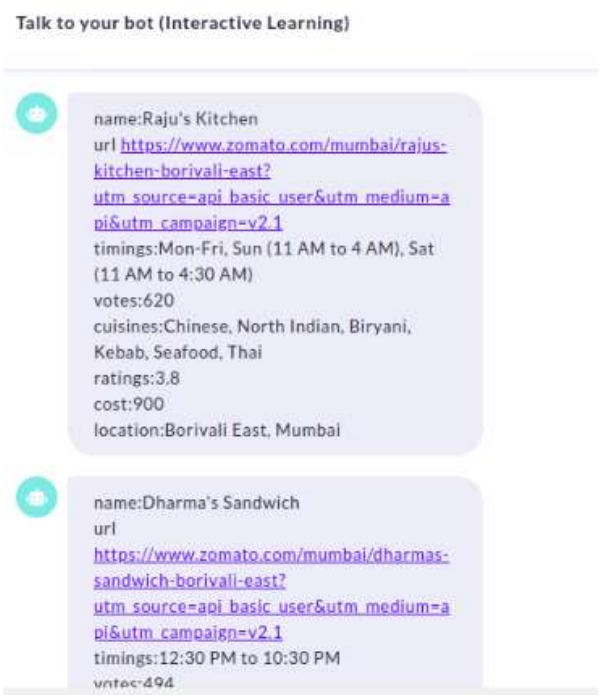

Fig.4. 2. Alternative responses by the bot

When the user greets or asks to do a search for a restaurant, Rasa NLU will find the intents of it and give the corresponding reply in the form of a text. As shown in the example Fig.1 above, the user is asking for restaurant options. The model notes the intent of finding top rated restaurants and entity as restaurants near Borivali. Then as per the training the following options are displayed as shown in Fig.4.1 and 4.2.

5 Conclusion

In this project restaurant chatbot, the whole chatbot is an android application and it is based on Artificial Intelligence (AI) to solve queries of local people regarding the restaurants and to provide relevant solutions. Having already sorted and efficiently stored data sets resulting reduce in service time for getting the most appropriate answer to a restaurant based query. Thereby, reducing the complex process of exploring the internet and various portals. Chatbot uses natural language processing unit of Rasa. It mainly uses natural language processing engine to decode the user query and according to the data category search it in the database.

References

[1] I. Sutskever, O. Vinyals, Q. Le ," Sequence to Sequence Learning with Neural Networks" , Neural Information Processing Systems Conference.,in arXiv:1409.3215 [cs.CL] , Dec 2014.

[2] O. Ludwig," End-to-end Adversarial Learning for Generative Conversational Agents" ,in arXiv:1711.10122 [cs.CL],November 2017.

[3] O. Vinyals, Q.Le ,"A Neural Conversational Model",in arXiv:1506.05869 [cs.CL], July 2015.

[4] T.Bocklisch, J.Faulkner, N.Pawlowski, A.Nichol, "Rasa: Open Source Language Understanding and Dialogue Management",in arXiv:1712.05181 [cs.CL] ,December 2017

[5]"A Chatbot from Future: Building an end-to-end Conversational Assistant with Rasa",towardsdatascience,,n.d.[Online].Available:https://towardsdatascience.com/a-chatbot-f 38 rom-future-building-an-end-to-end-conversational-assistant-with-rasa-ai-51a1c93dabf2[Acce ssed:October 20, 2019]

[6]"Build a Conversational Chatbot with Rasa Stack and Python-Rasa NLU", medium, n.d.[Online].Available:https://medium.com/@itsromiljain/build-a-conversational-chatbot-wit h-rasa-stack-and-python-rasa-nlu-b79dfbe59491 [Accessed:October 20, 2019]

[7]"Designing Rasa training stories",blog.rasa,n.d.[Online].Available: https:// blog.rasa. com/designing-rasa-training-stories/.[Accessed:November 3, 2019]

[8]"Your Complete Guide to Building Stateless Bots Using Rasa Stack",medium ,n.d.[Online].Available:https://medium.com/velotio-perspectives/your-complete-guide-to-buil ding-stateless-bots-using-rasa-stack-bc22a22e2fc7[Accessed:November 3, 2019]

[9]"Building a Weather Chatbot with RASA NLU and RASA Core",chatbotslife,n.d. [Online].Available:https://chatbotslife.com/building-a-weather-chatbot-with-rasa-nlu-and-ras a-core-8a295f29c405[Accessed:November 3, 2019]