



Chatbot for Weather API Using Python

Sreeja Dadishetty

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

July 3, 2021

CHATBOT FOR WEATHER API USING PYTHON.

Dadishetty Sreeja
B. Tech, Dept of CSE.
KITS, Warangal
Telangana, India
pinkysreeja321@gmail.com

*Abstract---*A chatbot is an application that enables auditory and textual approaches as a person to be used to launch and continue a conversation. Chatbots can be used by end customers or normal people, they identify the intension of customer and respond to it accordingly. Many organizations have been using chatbots extensively. Chatbots became well known since they are available 24/7, and they provide the consumers with positive service and are much cheaper.

Chatbot for weather API helps in extricating the temperature. for instance, in the event that I need to know a temperature at a specific area that is given as the input, when we run the code which is written in python coordinates to whether API where the connection referenced and it likewise utilizes Rasa NLU that is fundamentally used to assemble chatbots and voice applications, as the outcome bot reacts to the information and showcases the temperature both by typescript and audio.

Keywords--*Weather API, Chatbot, Rasa NLU, Application*

I. INTRODUCTION

A. Introduction

A chatbot is a chat interface service that communicates with people. By talking and typing a person in the same way you may ask questions. The chatbot usually

responds in a dialog style and can perform actions as part of your talk.

There are mainly two types of chatbots that are present they are:

1. Chabot based on Rules:
2. Chatbot based on Ai:

1. Chabot based on Rules:

1.A set of laws are applied to one. It can only answer very specific commands. The chatbot does not know what you are doing, because you do not use proper commands or phrases.

2.They are programmed to identify certain words and patterns that enable you to address preset responses.

3.This chat type does not allow the human being to adjust and analyze well the expectation and purpose of an individual.

4.The bot is restricted it can only reply to certain commands.

2. Chatbot based on Ai:

The second kind employs machine learning and artificial intelligence to offer the best answer. These AI-enabled chatbots are going to be called. It:

- Recognizes language and orders.
- Will benefit from user experiences endlessly to help forecast their needs.

- Will talk with an individual in the same manner.
- The details received from each contact can be stored and categorized.
- It will evaluate information to distinguish the information that is worthless and not.
- Know where to save this stuff, so that it will be able to use it again.

II. RELATED WORK:

The weather API chatbot assists with the temperature extraction. In case I have a temperature to know in a particular input region, for example, when the code that was written in python is running is syncing whether or not an API where the connection reference has taken place. And also, when the outcome bot reacts to the information and shows all temperatures with the Rasa NLU which is essentially used for assembling chatbots and voice applications.

III. METHODOLOGY

A. RASA

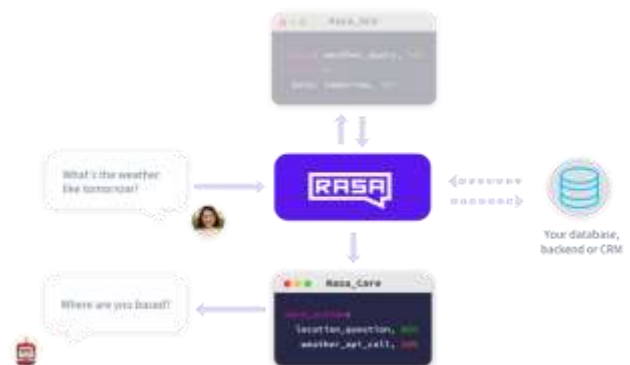
Rasa is the framework that plays an important role in the weather chatbot. Rasa is basically a framework using which we can create chatbots. Rasa has 2 frameworks further:

1. Rasa nlu
2. Rasa core

Rasa nlu: Teach our chatbot to recognize what user is saying (or) it is trained on different questions user can ask to the chatbot. Rasa NLU offers an intentional classification (decides what the user asks for) open-source natural language processing platform, extracts the bot in the form of structured data and lets the chatbot understand what the user is thinking.

Rasa core: Responsible for responding to the user on the source of request what user asks to a chatbot.

A chatbot-based dialog architecture that uses a probabilistic model such as an LSTM neural network to take the standardized data from the NLU and predict the next best step, rather than what it is/the argument. In other words, the Rasa NLU task is to view the user feedback as organized data and it is the task of Rasa Core to determine the next chatbot series of acts. Rasa Core and Rasa NLU can be operated independently and are distinct from each other.



B. Installation

The initial installation instructions are:

- 1) `python -m venv myvenv`
- 2) `myvenv\Scripts\activate`
- 3) `python -m pip install --upgrade pip`
- 4) `python --version`

Note: If you are working with a virtual environment just make sure to upgrade this pip.

Other installations are:

- 5) pip install rasa_nlu
- 6) python -c "import rasa_nlu; print(rasa_nlu.__version__)";
- 7) pip install spacy
- 8) python -m spacy download en_core_web_md
- 9) pip install -U rasa_core
- 10) pip install sklearn_crfsuite

Above mentioned 10 steps are required in order to install and the required packages and set up the complete rasa environment.

C. weather chatbot working



There are 4 phases involved in the working of the weather chatbot they are:

Phase 1: Train: Firstly, you need to train your bot what it requires to know regarding the weather/temperature. Here the rasa nlu play an important role in training the chatbot.

Phase 2: Build: Creating the weather-related dialog flows. i.e.; different types of ways that the user can ask questions to the bot.

Phase 3: connect: link your bot to openweatherAPI platform. i.e.; first you need to create an account in that particular platform. Then it gives you a unique API key, you should give the URL of

openweatherAPI and its key such that the bot gets connected.

Phase 4: monitor: Get insights into the weather chatbot preparation and working. The last step involves the user to monitor weather the chatbot is working well or not otherwise repeat the same procedure from phase1 and notice the problem where it is going wrong.

Most conversations are based on a fundamental idea of:

Entities: what the user is talking about.

Intents: It catches the bot which should respond on a text basis. This can be categorized into a small number of categories such as greet, goodbye, inform, positive, negative, status, etc. The aim is to find information on the Bengaluru weather if a user asks, "What's the weather in Bengaluru?"

Responses: The response is provided by the chatbot. It identifies from its source the necessary predefined answers. The most suitable response is then selected based on intent and context.

IV. Requirements:

A. Hardware Requirements:

Pentium pro-processor: Intel produced and manufactured Pentium Pro, the sixth generation x86 microprocessor was launched on November 1st, 1995. The P6 microarchitecture has been implemented and was initially designed in a whole variety of applications to replace the existing Pentium.

RAM 512MB or more: the minimum size of random-access memory is 512 MB and maximum can be anything above that.

B. Software Requirements:

Language: python

OS: Windows/Linux.

V. PROS AND CONS

PROS:

- 1) This chatbot helps to know the weather conditions at the near or far distance with in marginal amount of time. i.e., saves time.
- 2) Help to provide info regarding temperature any time i.e., available 24/7.

CONS:

- 1.Limited Functionality.
- 2.Don't Understand Natural Language.

VI. RESULTS AND DISCUSSION:



VII. CONCLUSION AND FUTURE SCOPE:

Weather Chatbot helps to know the weather conditions using the weather API at the near or far distance within marginal amount of

time. And Rasa Nlu and Rasa Core are helpful to extend chatbots. Users will only continue to utilize a chatbot prototype if they are happy after a first testing phase and the high client value is provided in the new solution. At the end of the online poll, participants were asked whether they would continue to utilize the chat to get information on the weather.

VIII. REFEEENCES:

- [1] Baya Abu Shawar and Eric Atwell, 2007 "Chatbots: Are they Really Useful?"
- [2] D. Mladenić, and L. Bradeško, "A survey of chatbot system through a Loebner prize competition," 2012.
- [3] Atwell, E. (2005). Web chatbots: the next generation of speech systems? European CEO, November-December:142–144.