NLU Chatbot for Flight Offers Search Project Report

Yuetao Li
Computer Science, University of Michigan, Ann Arbor

Abstract- The project is a chatbot for best flight offers searching and other flight related utilities built with Rasa NLU as the natural language understanding framework and uses Telegram as an interface. It can extract entities such as the departure date, return date, origin and destinations from user's natural language and use them to search for flight offers from a rest API and uses Telegram bot as an interface to interact with user. The chatbot is capable of intention recognition, entity extraction, contextual conversation and negation handling.

Index Terms- Chatbot, NLU, NLP

I. Introduction

NLP (Natural Language Processing) is one of the most important field of Artificial Intelligence. Natural language is different from programming language in that it doesn't have strict syntax and the meaning of it is ambiguous, which makes NLP particularly difficult for computer program. However, with the development of machine learning, a chatbot with NLP capability becomes more and more viable and easy to make. In this project, various basic concepts and techniques are put into practice to achieve a chatbot for flight offers search that is capable of NLP.

II. Background

1. Natural Language Understanding

The most important distinction between clickbots and AI powered chatbots is the ability to process user input in natural language. The AI chatbot can use machine learning to learn from a

big set of existing data and later properly process any user input in natural language. It doesn't require developer to list all possible user inputs.

2. Contextual Conversation

It means that when the chatbot is replying, it not only considers current user input but also the context in which they are talking. It memorizes previous user input and take all of them into consideration when replying. For example:

- What's the weather today?

- The high temperature for today in San Francisco will be 78, and the low will be 64.

- What about Los Angeles?

- The high for today in Los Angeles will be 84, and the low will be 68.

In the second question, the user doesn't say weather but the chatbot knows that they are talking about weather because the context in which they are talking.

3. Negation Handling

To handle negation in natural language is very easy for human being, but it's very difficult for computer program to do that because simply looking for negative words cannot guarantee

correctly identifying negation. For example: “I don’t want to eat bread”. is the negation of “I want to eat bread”, but “I didn’t say that I don’t want bread”. implies that bread is okay.

4. Rasa

Rasa is an open source framework used to do NLP in this project. It has built in TensorFlow, Spacy and other commonly used NLP tools. It is used for intent classification and entity extraction in this project.

I. PROJECT ANALYSIS

1.Training Data

The training data is from the ATIS (Airline Travel Information System) and is transformed to Rasa NLU JSON format to be used by Rasa NLU.

Besides data from ATIS, regular expressions are also added to features to extract entities such as flight number, airline code and airport which have a fixed pattern. Moreover, examples with intents like negation, greeting, and termination are also added to training data set. The pipeline used to process training data is shown in the picture. Since training data includes more than 1000 total training examples. TensorFlow and CRF Entity Extractor is used instead of pretrained intent classification by Spacy. Moreover, the regex features is only available with CRF Entity Extractor. In the case of training data with large number of examples and in a specific field of conversation, supervised training pipeline has better performance than the pretrained pipeline. In

the end, Duckling by Facebook is used to extract the time. The duckling server is running locally through docker.

```
language: "en"

pipeline:
- name: "WhitespaceTokenizer"
- name: "RegexFeaturizer"
- name: "CRFEntityExtractor"
- name: "EntitySynonymMapper"
- name: "CountVectorsFeaturizer"
- name: "EmbeddingIntentClassifier"
- name: "DucklingHTTPExtractor"
  # url of the running duckling server
  url: "http://localhost:8000"
  # dimensions to extract
  dimensions: ["time"]
```

2.Front End

Telegram Bot is used as the interface for the chatbot to interact with user. Python Telegram Bot is used to access the official API of Telegram Bot.

3. Search Result

The actual flight offer search is done by Amadeus for Developers API. Amadeus-python is used to access the API. Another option would be using knowledge graph and database such as MySQL to store the possible answers locally.

4. Entity and Intent Extraction

The project uses a state machine to control the conversation (policies of state machine is shown in the image). It uses the interpreter of Rasa NLU to extract entities such as origin airport and destination airport. Entities with more fixed pattern such as airport code and class type is

extracted by using regular expression. Global variables are used to save parameters in order to achieve contextual conversation.

```
# Define the policy rules
policy = {
    (INIT, "greet"): greeting,
    (INIT, "inquiry"): inquire,
    (INIT, "default"): confusing,
    (INIT, "checkin_link"): getCheckin,
    (INIT, "flight"): getFlight,
    (INIT, "terminate"): backToInit,
    (CHECKIN_LINK, "default"): getCheckin,
    (CHECKIN_LINK, "terminate"): backToInit,
    (CHECKIN_LINK_CHOOSE, "decline"): backToInit,
    (CHECKIN_LINK_CHOOSE, "default"): setCheckinAir,
    (FLIGHT_OFFER, "default"): getFlight,
    (FLIGHT_OFFER, "terminate"): backToInit,
    (FLIGHT_OFFER, "decline"): backToInit,
    (FLIGHT_OFFER_ORG, "default"): setFlightOrg,
    (FLIGHT_OFFER_ORG, "terminate"): backToInit,
    (FLIGHT_OFFER_DEST, "default"): setFlightDest,
    (FLIGHT_OFFER_DEST, "terminate"): backToInit,
    (FLIGHT_OFFER_DEP_DATE, "default"): setFlightDep,
    (FLIGHT_OFFER_DEP_DATE, "terminate"): backToInit
}
```

Negation is handled by identifying negative words. The time and date of the flight is extracted by using Duckling made by Facebook. Duckling server is running locally through Docker.

Intent	**Explanation**	**Example**
greet	Greeting	“Hi, what can you do for me?”
inquiry	Ask for a flight offer	“Can you find me a flight?”
terminate	Stop and start a new search	“Start a new search.”
decline	Negative expression	“No”
flight	Start a flight offer search	“Can you find me a flight from ORD to JFK next Monday?”
default	Default intent	“aklsdjfklasdj;f”
Checkin_link	Check in with a specific airline	“I want to check in with DL.”

5. Functions

Function	Intent	Example
getCheckin	Get the check in link for a specific airline	“I want to check in with UA.”
getFlight	Search for the best offer given the date, origin and destination	“Find me a flight in business class from ORD to JFK tomorrow.”

III. Conclusion

To build a chatbot with contextual conversation, negation handling, and natural language processing ability, the following core functions are needed: 1. A machine learning framework used to extract intent and entities of user input. 2. A state machine used to control the contextual conversation. 3. A knowledge graph and MySQL database used to store the possible answers to user’s question (or just use API to acquire answer). In this project, Rasa NLU is used as the natural language processing framework and TensorFlow and CRF entity extractor are used to do train the interpreter. The answers to user input is acquired using Amadeus for developer API and it’s python wrapper. The front end used to interact with user is Telegram Python bot.

IV. References

[1] Rasa Official - Legacy Rasa Docs
[2] Python Telegram Bot - Python Telegram Bot's Documentation
[3] Duckling - Duckling
[4] Amadeus for Developers – amadeus4dev