NLP Final TP: Rule-Based Chatbot

Project Summary

As a part of the curriculum of the Master 1 (M1) course entitled “Natural language Processing (NLP)”, the students will complete this project work focusing on building a rule-based chatbot focused on customer service. The students are to divide themselves into **groups of 4**, and are to present their work during the final session of the course. Further information is provided below. Also, the project tasks are explained.

For any further details, please contact the instructor: **Khodor Hammoud**

## 1. Project Scenario

You are a data scientist working at an airline company, and you have been assigned, with a team of 3 others, to build a chatbot that can alleviate some of the requests and questions done by clients off the customer support team. This chatbot is to accomplish this task by having pre-prepared answers to a predefined set of well-known customer questions. In order for the chatbot to answer user questions, a pipeline of data preprocessing and matching has to be done.

## 2. Chatbot Types

There are primarily 3 types of chatbots:

1. Rule-based: can answer a pre-defined set of statements (questions, chats or requests), and default to a base response in case an unknown statement was provided. This type of chatbots can be useful and accurate when the conversation topic (and potentially questions) are known. Example: that chatbots found at the contact page of websites (<https://communaute.red-by-sfr.fr/#redclicid=A_Menu_Assistance> )
2. AI-based: these chatbots can train from a provided corpus, and can learn to respond to novel questions by generating responces from the provided corpus. This type of chatbots could be more useful in scenarios where discussion topic Is unknown, like in the case with general-purpose chatbots. Example: online conversation chatbots (<https://www.kuki.ai/> )
3. Hybrid: if the statement provided fits the criteria of a pre-defined set of answers, this type of chatbot replies with a pre-defined answer. Otherwise, it can answer in an AI-based method.

For the purposes of this project, a **rule-based** chatbot is sufficient, although you can expand upon it to cover non pre-defined statements, and turn your chatbot into a hybrid one.

Chatbots can also learn from conversations, where it can, for example, ask the user about the proper response for a new statement. In our case, simply replying with a pre-defined statement for unknown statements is sufficient.

## 3. The Training Data

The training data provided is a series of chats and responces to clients to an airline company. It’s purpose is to provide you with a sample of real-life conversations between clients and client support to consider for your chatbot. You can use this data to extract **intents** (described in section 4) from the client questions and to add some of the responses to your chatbot’s predefined list of responses (if you so choose).

## 4. Intents

Intents are the topic of a conversation. For example, the intent of the sentence:

“@AmericanAir just curious how late a plane had to be before it's marked as delayed??”

could be “departure”, or “departure delay”.

Intents form the category of a conversation. You can use the “questions” column from the training data to extract intents. You can achieve this by using one of the below methods:

1. Topic Modeling: is an un-supervised NLP technique that classifies texts based on their words similarities. The sentences that contain similar texts are more likely to be closer to each other in topic than those that don’t. In topic modeling, you give the ML model your preprocessed data, and n: the number of topics you want the model to divide the data into. The output is n groups of text each supposedly belonging to a different topic. It is up to you to read the texts in each group and deduce the topic. For algorithm reference, you can use *Latent Dirichlet Allocation (LDA)* available in multiple python libraries, like: [gensim.models.LdaMulticore](https://tedboy.github.io/nlps/generated/generated/gensim.models.LdaMulticore.html).
2. Similarity Search with word embeddings: this approach is a little more habds-on, but can yield better results. You can vectorize all the texts in the “questions” column, and measure the distance between each of them. Those sentences with the closest distance would be more likely to have the same intent.   
   Notes: You need a vector representation of each sentence to be able to compare similarity between sentences. And although we’ve already done this by embedding every word in a sentence, and then averaging all the embeddings to get one vector representation for a sentence, we can get better accuracy by using [Doc2Vec](https://radimrehurek.com/gensim/models/doc2vec.html). Keep in mind that Doc2Vec needs to be trained on your data before you can use it. Example on how to train it can be found [here](https://medium.com/@mishra.thedeepak/doc2vec-simple-implementation-example-df2afbbfbad5).

## 5. The Process

Once intents are extracted, you can basically define a dictionary of “intent”: [responses], where for every user question, you extract the intent (by passing the question through the same process you used to extract the intents from the provided dataset), and then provide the response that is closest in distance to the user’s question (by using vectorization again, like Doc2Vec for example).

To get you started, your intents list should contain the following:

* Greeting
* Goodbye
* Apology
* Thanks

## 6. Data Preprocessing

This is an essential step of the project as improperly processed data can lead to great in-accuracies in the chatbot. You are responsible in determining what are the proper preprocessing steps to be taken on every stage. Reminder, preprocessing includes:

* Tokenization
* Stopwords removal
* Lowercasing
* Stemming/Lemmatization
* Punctuation removal
* Emoji hadling (translation into symbols or deletion)
* Removing noise (URLs, hashtags, …)
* Removing non-english words

It is up to you to determine which pre-processing steps to be taken.

## 7. Deliverable

You and your group are to present a running chatbot during the project presentation. Your chatbot will be faced with a series of questions, and it is expected to have a comprehendable (and preferably helpful) answer. **Your submission to the moodle will be an HTML or PDF export of your project. Each project should contain the name of all the group members, and only 1 member of every group should submit.**