Movise Domain Bot
Final project

Andrea Tupini
MAT. 194578
andrea.tupini@studenti.unitn.it

Abstract

abstract.

1 Introduction

This project consisted of building a bot that could answer questions on the movie domain. We had to build both the NLU and dialogue modules. In this report we'll see a bit how the bot works and the decision made during it's design, plus some encountered problems.

2 Data Analysis

The original data that was provided consisted of a movie database that contains various information on the movie domain and the data needed for training the NLU module of the bot (this was given in NLSPARQL format).

2.1 NLU Data

Originally, the provided NLU data consisted of the following files:

NLSPARQL.test words and IOB tags for testing

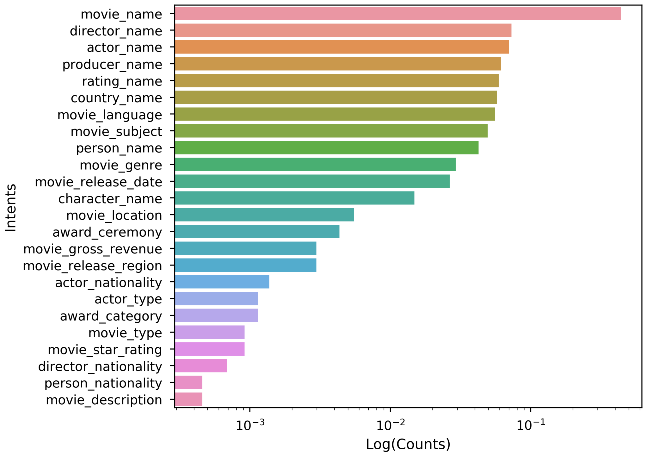
NLSPARQL.train words and IOB tags for training

NLSPARQL.test.utt.labels contains labels on what each question in the test set is about

NLSPARQL.train.utt.labels labels on questions of training set

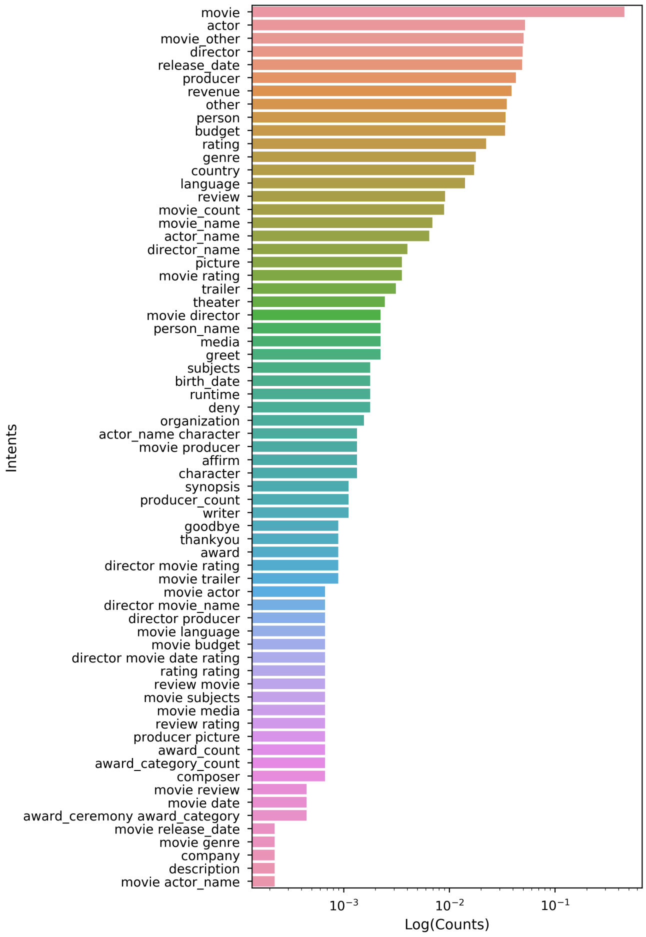
The NLSPARQL format is not recognized by Rasa, so these had to be converted to a properly formatted *json* file. In the rasa format, for each sentence we need to provide which are the entities and intent of the sentence (entities and intents are defined in section 3.1). Extracting entities was just a simple matter of using the IOB tags that are not *O*s, and the intents were taken directly from the *labels* files.

Here we have a graphic that shows the count of different entity types provided in the dataset (note that it is log scaled so as to account for the difference between the most common entity and the least common one):



Here we can see that the *movie name* entity is much more common than any of the other entities. But we do have a good amount of the entities we would expect our users to use the most, which is good so that the model is actually able to learn them.

Below is another graphic showing the count of intents (this has also been log scaled):



It is easy to note that also in this case *movie* is the most common thing talked about. We can see that the difference between the most common item (*movie*) and the least common one (*movie actor name*) is much more pronounced for intents than it is for entities.

Note that entity types may take many different values (for example, *movie_name* can be an entity representing diverse movie names in different sentences), while intents are always set (a sentence can only have one intent). This is the reason why we have much less entities than intents.

2.2 Database

The database basically contains all the knowledge that the bot has on the movie domain. If something is asked of the bot that is now in the database then it won't be able to answer. Each row in the database contains the information for one movie and it has the following columns:

- Title
- Actors
- Director
- Genres
- Country where it was made
- Release Date
- Original language
- Duration in minutes
- If it was released in color or not
- Budget
- Plot Keywords
- Gross Revenue
- IMDB core
- Likes on the Facebook movie page
- Link to the movie's IMDB page

The original database was fine, there were some empty columns but in those cases the bot will tell the user that it doesn't have the information to answer. However there was one issue that required to modify the database to fix it and it was that some rows had as *release date* years that were not possible (ie: *18000000*), so all of the publishing years above 2018 were deleted.

3 Bot Modules

3.1 NLU Module

taslk about entities

3.1.1 Rasa NLU

remember to talk about pipeline

3.2 Dialogue Module

3.2.1 Rasa Core

remember to talk about domain

3.2.2 Policies

3.2.3 Custom Actions

3.2.4 Forgetting

Not stremlined/goal otiented task.

4 Evaluation

ev

5 Speech

6 Difficulties

fw.

7 Possible Improvements

One thing we can see in the entity counts graphic in 2.1 is that we could actually join some of these entities toghether and so remove some entropy from the daataset.

8 Conclusion

remember to add references to tools’ sites

References