AMPLAB1 ASSIGNMENT REPORT

Santiago Diana Sánchez

With no doubt, this part was the hardest we had to do, dealing with RAM crashing and trying to make sense of everything we were doing. In this report I will explain how I interpreted what we had to do and how I managed not to have RAM crashing each cell.

First of all, in general I have always used !head operation for seeing how was the file disposed. I learned that command in class and it has been very useful. I also used tqdm bars just to know where I was in the process constantly.

First of all, I extracted all the information regarding "listening events" and added it to a set, to avoid repetitions and calculate the mapping in a more efficient way. It may also be noted that I had to save information into CSV files a lot of times because otherwise I lost the information for the next day of work.

After that, I did the mapping. I have only taken into account the ones that have exact_match or high_quality match_type, to avoid bad mappings. So I read both files, the original mapping file and my mapping file (the ones with the msID from listening events mapped), and performed the mapping into a dictionary. Then I save that dictionary into a CSV file (myMapping) and delete it. Otherwise, I would have RAM problems.

After that, the second part for me is more difficult. The artistmbID mapping. For that, I basically select, by chunks, the artist_mbids corresponding to certain recording_msid. For that, I get the very first artist that I encounter in the canonical_musicbrainz.csv file. I do that because I acknowledge that some recording_msids have more than one artist, so I thought that selecting the first was the best option. As the length of the artist mbID is fix, we just need to take from the 1st to 37th character of every artist mbID section and we will get the proper artist mbID.

After that, I create the msid_mbid_artist.csv, which contains the mapping between msid, mbid and artist mbID. So, we are near the last part. In the colab notebook you can find how I show the first lines of the csv.

The last part is the final CSV distribution. Just doing a bit more mapping through the files and everything was fine. I may say that I haven't been able to accomplish the last part, to write also the artist names, because of questions of time. That is why I have done the recommendations using recording IDs.

The last part is the **COLLABORATIVE FILTERING** thing, which makes the recommendations. This was not the most difficult part, as I found easy to get the artist_user_plays sparse matrix with scipy. In the jupyter notebook can be seen how do I train the model, save it, load it again and use some user_ids and mbID to do both recommendations and similarities. In the last part of the notebook I discuss an example of possible recommendations with a previous known artist.

PERSONAL OPINION

If I can say something, I think this assignment was too long in some of its parts. I learned a lot on how to manage large amounts of data, but personally I believe the information provided was a bit messy so I spent so much time trying to understand the data.

Maybe it is easier for the next time to have a more guided notebook, or at least not to make such difficult the steps to get the final CSV. I agree on the amount on data, and how to learn to handle it, but I'd maybe explain a bit better how to get to the final representation.

Thank you very much. I learned a lot!