Project
Machine Learning for Natural Language Processing
2022

Baulain Roméo
romeo.baulain@ensae.fr
Thelot Léonard
leonard.thelot@ensae.fr

1 Problem Framing

The initial aim of this project was to use question/answer amazon dataset to create a model able to answer a question about an amazon product. Unfortunately, the state of the art for question answering works only if we give to the model a question and a text, where a portion of the text answer the question, then the model return this portion of the text. We looked for description of amazon product to play the role of the text but didn't find anything. We act that because of a lack of information about amazon products it was impossible for a model to generate the right answer.

So we decided to work on the Amazon review dataset (2014 version), available here : http://jmcauley.ucsd.edu/data/amazon/links.html. The dataset is made of more than 1 million users' reviews on electronics products sold by Amazon, and the associated overall of the product according to the user (from 1 to 5).

With this dataset the aim was to create a model able to predict the overall thanks to the associated review. The problem is a classification model with a target taking values in $\{1,2,3,4,5\}$.

The plan is to use neural network model with a freezed first layer representing the words in a latent space, and then train multiple layers beyond the embedding one to adapt the model to our product. The loss used to assess the model was the cross entropy loss which is often used for classification model. Finally the metric used were recall, precision and accuracy.

2 Experiments Protocol

First of all, the only variables of interest were overall and review, hence we remove the other variables. Given the fact that the tested models, tend to overrepresented label 5 (because of imbalance), we choosed to balance our dataset, with 80 000 observations for each target. Hence we finally have 400 000 observations on the 1 600 000 available at the beginnig, this choice is made by RAM and time constraint on Colab. Besides, we choosed to use Glove with a latent dimension of 300 instead of FastText because it took less RAM on Colab.

After a baseline model, consisting of an embedding layer and a hidden layer following by a softmax one, we add several hidden layers of the same dimension that the first one to improve the results, finally we tried to adapt the BERT structure however we had a model running to long for the colab RAM so we couldn't go furthter into it.

3 Results

We are first interested in the distribution of the classes we are going to predict and we can see on the histogram (figure 1) that grade 5 is over-represented in the dataset compared to the others. It may therefore be necessary to rebalance the dataset.

We then check that Zipf's law is verified, which is indeed the case as can be seen in the figure 2.

By plotting the word clouds (see figure 3) by rating (without stop words), we can see that the share of ”good” increases a lot between ratings 1 and 3, then decreases a little for ratings 4 and 5, but with an increase in ”great” in return. These word clouds are therefore reassuring for the learning of the model because we can hope that the scores are consistent with the descriptions given by the clients.

We can see that the LDA (which you can find in the notebook) discriminates more according to the type of product than according to the consumer's opinion on the product. Cluster 5 represents the opinions concerning computer type products, cluster 4 represents positive opinions, cluster 3 represents the opinions concerning camera type products, cluster 2 represents the opinions concerning television type products and cluster 1 represents the opinions concerning the products of the stereo speaker type.

Let's now look at the model. We have taken the example of the TD 3-4 and adjusted it to our data. We have also added layers to make the model more complex and to obtain better results. Indeed, with only one layer, the network could not learn well during the training. The loss stagnated after a few epochs. Having thought that the gradient might be in a local minimum, we increased the learning rate to see if the results improved but without success. We increased the number of layers with success as we can see on the figure that the loss decreases with the epochs. On the other hand, the accuracy is not very good but we can be satisfied with the confusion matrix (figure 4). Indeed, we notice that when the model is wrong, it does so in such a way as to predict classes close to the expected one (e.g. if the expected score is 5 and the network is wrong, it is more inclined to predict 4 then 3 then 2 then 1 in that order). Given that scores and ratings are subjective to the users, this is not alarming.

4 Discussion/Conclusion

In this project we tried to predict the ratings of electronic products based on user reviews. After having observed via descriptive statistics the differences in vocabulary between the different ratings, we created a deep learning model based on embeddings of dimensions 300 available via Glove. Finally we added several layers to try to take advantage of these embeddings.

The results obtained are not very conclusive (accuracy of 0.32). We need to optimise the learning rate or perhaps make the network more complex, especially the dimensions of the layers. Moreover, the model training took about 45 minutes to converge for 10 epochs, a dataset of 400,000 lines (with an average size of 127 words per review) and using Google Colab GPUs.

We were unable to realise our original idea of answering user questions because of the structure of the data which did not include the answers to the questions and therefore made training impossible.

5 Appendix

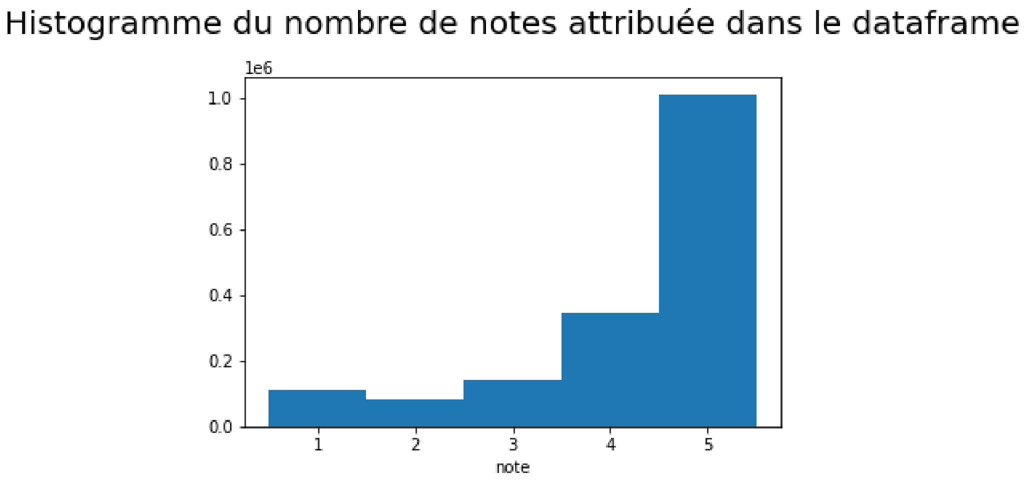

Figure 1: Class repartition

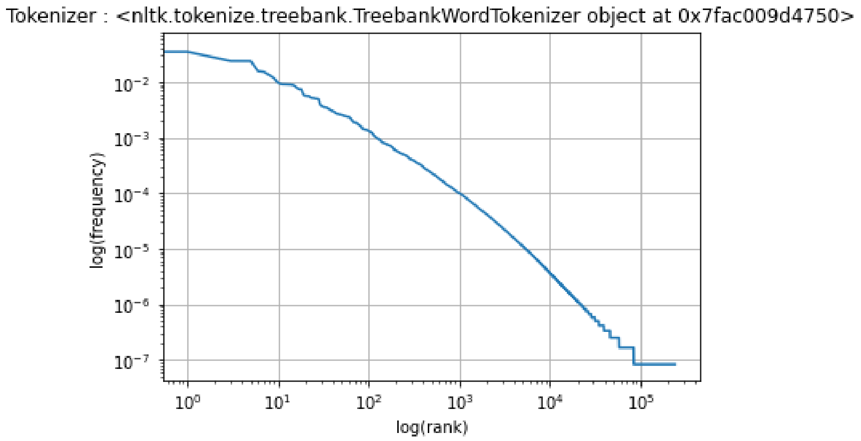

Figure 2: Loi de Zipf

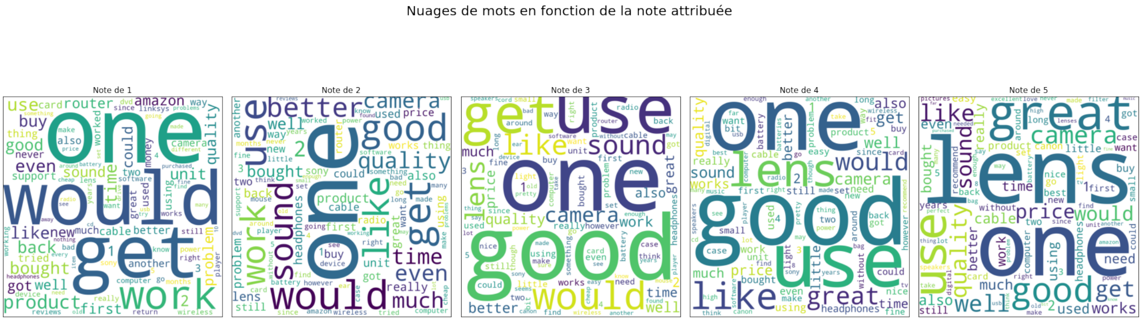

Figure 3: Word cloud by category

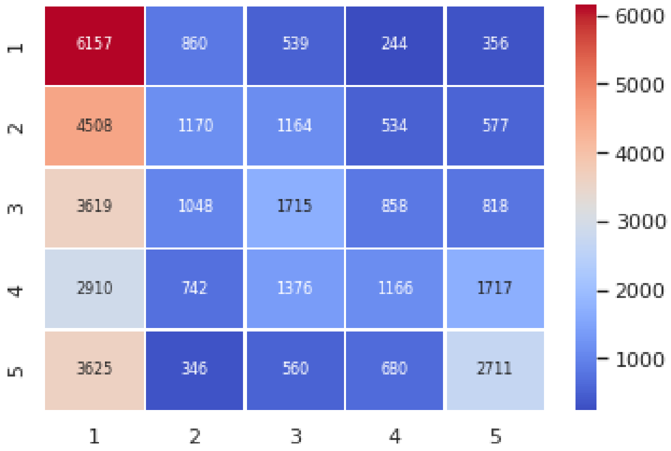

Figure 4: Confusion matrix