Davide Lagano
Georgios Patrikis

Lab Group 24

REPORT

Laboratory Exercise 1: Preprocessing and Utilizing Web Data

1. Introduction:

The first Laboratory Exercise aims to go through different steps:
Pre-processing data (cleaning, normalizing, noise reduction), processing data (building a word space model), visualising portions of the resulting model.

2. Discussion about tasks:

In the first task, we visualized a couple of sample pages using a web browser and a text editor. We noted that the information available regarding the website are provided in HTML format. For this reason, it is difficult to get information about the website, as the format is not immediately understandable.
Accordingly, we run the "text-extractor.pl" script in order to ‘clean’ all HTML from the web pages and extract just the essential words from the website.
Unfortunately, we saw that the "text-extractor.pl" script also deleted important information about the text; of course, lose key information is not an excellent way to proceed.
Thus, we have displayed a frequency list calculated over all documents in the folder. The most common words represented in the list are words like preposition, articles or conjunctions and are useless for our purpose. For example, we can see words like: ‘about’, ‘other’, ‘all’, ‘may’, ‘one’, ‘such’, ‘at’, ‘not’, ‘it’, ‘an’, ‘from’, ‘s’, ‘by’, ‘on’, ‘or’, ‘as’, ‘in’, ‘to’, ‘for’, ‘the’… Just some of the words are important for our purpose: ‘information’, ‘language’…

Furthermore, some words with the same meaning are classified as a different word. For this reason, we run the "lemmatizer.pl" script.

We can see that previously the script found 1377 times the word ‘language’ and 791 times the word ‘languages’, but after running the "lemmatizer.pl" script they are summarised in 2168 'language'. This also allows us to find words that previously wasn't in the list because was separated in different words. For example: initially, the word 'article' was found 522 times. After 766 times.

Then we trained and tested the model feeding him with the text. In this way, we were able to obtain the most important words and the ten words that best matched with them. Of course, without any preprocessing, a lot of stop words was paired with the principle words. For this reason, we trained and tested the model again but without the stop words. The difference between the two results is enormous. The best example to explain it is to take the word 'matrix'. We can see that without preprocessing, at least half of the words are step words like 'to', 'if', 'that'. Of course, after the preprocessing, the step words disappear and have been replaced with meaningful words like 'approximation', ‘determinant’…

3. Conclusion

In this laboratory exercise, we learned how to do preprocessing in order to obtain cleanest and most significant result.
Is important to keep in mind that “these models usually are trained on hundreds of millions of words, what you see here is the performance on a far much smaller data set. The performance usually increases with the size of the data.”