Homework 2: Document Classification

Form:	Zip file including Jupyter notebook file and images
Language:	English
Requirements:	The report should be clear, readable and include all code documented
Submission:	zip file via moodle. The file name should include the students' ids ment
Contact:	zolfi@post.bgu.ac.il Alon Zolfi
Deadline for submission:	**December 20, 2020**

Students will form teams of two people each, and submit a single homework for each team. The same score for the homework will be given to each member of the team.

Submit your solution in the form of an Jupyter notebook file (with extension ipynb). The notebook should include all the documented code required to answer the questions, graphs and tables you generated. We should be able to run the notebook. For each question, please answer the question and describe which functions you used to address it. Document clearly your functions. Python 3.6 should be used. The answers should be provided in a separate pdf file including answers to questions, explanation of what you did, the solution assumptions, challenges and Jupyter notebook output including graphs and tables.

The goal of this homework is to let you practice text analysis and classification with python.

Submission: Submission of the homework will be done via moodle by sending a zip file including Jupyter notebook with all the answers, code and final report. The name of the zip file should be a concatenation of the id number of the submitting students separated by a '_' (e.g. 333003321_222333456). The homework needs to be entirely in English. The deadline for submission of Homework #2 is set to December 20, 2020 end of day Israel.

Task

The IMDB dataset is having 50K movie reviews for natural language processing or Text analytics. This is a dataset for binary sentiment classification. It includes a set of 25,000 highly polar movie reviews for training and 25,000 for testing. Use the external libraries and resources presented in the class for task implementation. Please set a variable at the beginning of the exercise, with the dataset folder.

1. **Text pre-processing and exploration:**

- Download the corpus.
- Split to train and test
- Clean and normalize the text (e.g. tokenization, lower case, stop words removal, stemming)
- Explore the dataset (#of docs from each category, terms (uni-grams, bi-grams) distribution per category). Present a table of top 10 terms per category.

- Explain expected challenges

2. **Document classification – un-supervised learning:**

Here, you should use an un-supervised learning approach for the classification task.

- Use accuracy metrics to evaluate the accuracy of your model

3. **Document classification – supervised learning:**

Here, you should test combinations using 2 feature extraction methods and 3 machine learning models to train a classification model. Test the impact of changing at least one parameter per feature extraction and machine learning model on classification result.

- Implement feature extraction (Bag of words, n-grams, TF-IDF, any other feature - optional)
- Classify using machine learning methods (e.g. SVM, Naïve Bayes)
- Tune each model parameters, as well as pre-processing and parameters steps to optimize the results
- Use accuracy metrics to compare between the different models
- Use the best model selected in the previous steps for prediction on the test set. Present the accuracy of the model and the challenges.
- Describe the task challenges, and explain effective solutions

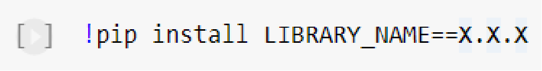
4. Supervised vs. unsupervised learning approach

- Compare the accuracy of the supervised learning model with the un-supervised one. Discuss the advantages of each approach
- Suggest an approach to combine the two to further improve the model accuracy. Explain your suggestion.
- Implement the suggested combined approach, present the model accuracy and discuss the results.

Good luck

Appendix:

Please make sure the code runs in Google Colab (https://colab.research.google.com/). For external libraries add an installation in the **first** cell of the notebook, for example:



```
!pip install LIBRARY_NAME==X.X.X
```

Where LIBRARY_NAME is the name of required library and X.X.X is the version number.

NOTE: Google Colab's default environment contains many built-in libraries. Thus, many libraries do not require explicit installation.

.ipynb files are different from .py – **don't** use a single cell for your entire code but rather split it logically to multiple cells.