[COM4513-6513] Text Classification with the perceptron

Instructor: Nikos Aletras

The goal of this lab session is to implement the binary perceptron and apply to it sentiment analysis, and in particular to predict the sentiment of movie reviews. The data you will use for this purpose are available from here: http://www.cs.cornell.edu/people/pabo/movie-review-data/review_polarity.tar.gz. Download it and take the first (alphabetic file order) 800 instances of each class as training data and the remaining 200 as testing and implement and evaluate the following:

- standard binary perceptron with bag-of-words representation (2 marks)
- randomizing the order of the training instances (use python's `random` library to fix the random seed so that your results are reproducible!) (1 mark)
- multiple passes over the training instances (show the learning progress in a graph) (1 mark)
- instead of using the last weight vector for testing, taking the average of all the weight vectors calculated for each class (0.5 mark)
- implement two feature types beyond bag-of-words. Discuss your choice of features. Does any of them help improve accuracy? (1 mark)
- What are the most positively-weighted features for each class? Give the top 10 for each class and comment on whether they make sense (if they don't you might have a bug!). If we were to apply the classifier we learnt to a different domain such laptop reviews or restaurant reviews, do you think these features would generalize well? Can you propose better features for the new domain? (0.5 mark)

You should submit a python file (lab1.py) that can be executed as:

`python3 lab1.py review_polarity` where review_polarity is the data folder.

You are advised to have separate train and test functions (1 mark) and you should comment your code to explain it (1 mark). Please make sure to mention if you've used Windows (not recommended) to write and test your code. You also need to accompany it with a lab1.pdf (no more than two A4 pages) describing the evaluation results obtained by answering the questions about it. Make sure your code is Python 3 compatible. There is no single correct answer on what your accuracy should be, but correct implementations usually achieve around 80%. The quality of the analysis and clarity of your report is as important as the accuracy itself. Please make sure your report is well structured and presented.

You are advised to draft your lab reports using Latex. There are many free Latex editors (e.g. TexMaker), Overleaf is a good starting point.

This lab will be marked out of 8. It is worth 8% of your final grade in the module.

The deadline for this assignment is the beginning of Week 3 lab and it needs to be submitted via MOLE. Standard departmental penalties for lateness will be applied.