Cmpe 493 Introduction to Information Retrieval, Fall 2021
Assignment 3 - Text Classification
Due: 30/12/2021 (Thursday), 17:00

In this assignment you will implement the multinomial Naive Bayes (NB) and k Nearest Neighbor (kNN) algorithms for text classification. You will use the Reuters-21578 data set (the same data set from the first two assignments). Reuters-21578 contains 21578 news stories from Reuters newswire. There are 22 SGML files, each containing 1000 news articles, except the last file, which contains 578 articles. There are a total of 118 topics (classes) and each article is classified into one or more topics.

You should perform the following steps:

1. Pre-processing the Data Set: The text of a news story is enclosed under the <TEXT> tag. You should use the <TITLE> and the <BODY> fields to extract the text of a news story. Implement your own tokenizer to get the tokens from the news texts and perform normalization operations including case-folding, stopword removal, and punctuation removal. Please use the stopword list that you used in the first two assignemnts for stopword removal and you can use the list in "string.punctuation" for punctuation removal for Python. Please use "latin-1" encoding while you read the .sgm files due to the corruption in one file.

2. Creating the training and test sets: Will will use the top 10 classes (topics) in this assignment. So, you should first identify the most common 10 topics in the corpus and then select as a dataset the articles that belong to one or more of these 10 topics. The news articles that are denoted with the LEWISSPLIT="TRAIN" tag should be inlcuded in the training set and the articles denoted with the LEWISSPLIT="TEST" tag should be included in the test set. Either use 10 fold cross-validation over the training set or extract part of your training set as a development set for tuning your classifiers. You should NOT use the test set to develop/tune your classifiers.

3. Implementing a multinomial NB classifier: Learn the vocabulary and the parameters of your classifier from the training set. Use add-one smoothing.

4. Implementing a kNN classifier: Obtain the vocabulary from the training set and use tf-idf based cosine similarity as a similarity function. Determine the best value for k by experimenting with different values of k such as 1, 3, 5, 7, and 9 on the development set or by using cross-validation over the training set. Report the micro and macro averaged F-score for each tested value of k on the development set or using cross-validation.

5. Evaluation: Report the macro and micro-averaged precision, recall, and F-score values of the NB and kNN (for the best k value determined in the previous step) algorithms on the test set. Note that even if you had separated part of your training set as a development set, it is suggested that, in the end, you merge them, train your algorithm with the entire training set with the determined best set of parameters to create your final classifier.

6. Statistical significance: Perform randomization test to measure the significance of the difference between the macro-averaged F-scores of the NB and kNN classifiers on the test set.

You should use Python to implement your algorithms. We should be able to run your program by following the instructions in your readme file. You have to state the exact commands to run the learning and classification components of your algorithms. You should NOT use any third party libraries, except the ones available in the Python Standard Library.

Submission: You should submit a *".zip"* file named as YourNameSurname.zip containing the following files using the Moodle system:

1. Report:
   (i) Describe the steps you have performed for data preprocessing. Provide the size of your resulting vocabulary.

   (ii) Provide information about what the top 10 classes are and how many documents each class has in the training and test sets. What is the total number of documents in the training set and the total number of documents in the test set. How many documents are labeled with more than one of the top 10 classes.

   (iii) Describe how you performed parameter tuning. Did you use cross-validation or did you use part of your training set as a development set. Please explain and report the best set of parameters that you determined (such as k in kNN) and justify your selection. Please describe what strategy you used to assign a document to more than one class and discuss how successful your strategy is.

   (iv) Provide your evaluation results on the test set, and provide and discuss your randomization test results.

   (v) Provide screenshots of running your algorithms.

2. Source code: Commented source code.

3. Readme: Detailed readme describing how to run your program including the Python version.

Honor Code: You should work individually on this assignment and all the source code should be written by you. You are NOT allowed to use any available libraries or any code written by other people. Violation of the Honor Code will be strictly penalised, not only by a zero grade from the homework, but also by filing a petition to the Disciplinary Committee.

Late Submission: You are allowed 5 late days (until 04 January 17:00 o'clock) for this assignment with no late penalty. After 04 January 17:00 o'clock, 1 point will be deducted for each late hour (unless you have a serious excuse).