Practice III

Text classification

Specification

In teams of 3-4 members do the following

1) Load the corpus *arxiv_normalized_corpus.csv* generated in practice II

   - Title and Abstract columns must be concatenated and will be used as features

   - Section colum will be used as target (class)

2) Split the corpus in train an test sets using 80% for training and 20% for testing

3) Create different text representations of the corpus using unigrams

4) Use different machine learning methods to train a model and predict test instances

5) Evaluate predictions of models

Corpus split

- Use *train_test_split* function of scikit-learn
- Instance of corpus should be shuffled (*shuffled = True*)
- Set random seed (*random_state=0*)

Text representation

- Use the following text representation methods:
  - Binary
  - Frequency
  - TF-IDF

Machine learning methods

- Try different machine learning (ML) algorithms. The following algorithms are known to perform well in text classification:
  - Naïve Bayes Multinomial (*MultinomialNB*)
  - Logistic Regression (*LogisticRegression*)
  - Support Vector Machines (*SVC*)
  - Multi-layer Perceptron (*MLPClassifier*)
- At least 3 ML algorithms should be tested
- It would help if you tuned the algorithm hyperparameters to improve the results

Evaluation metrics

- A classification report (*classification_report*) should be applied to the predictions on the test set for each variation of:
  - Text representation
  - Machine learning method
  - Classifier hyperparameters

Evidence

- Source code
- A report in PDF format describing the following:
  - Task to be solved
  - Selected machine learning methods
  - Adjusted hyperparametersl
  - Classification report of each experiment

Evidence

A table describing the experiments performed showing the best configuration of each ML method

Machine learning method	ML method parameters	Text representation	Average f-score macro
Logistic regression	max_iter = 200	binarized	0.85
Naïve Bayes	default	frequency	0.88
...	...	...	...
Multilayer perceptron	hidden_layer_sizes = (200, 100)	Tf-idf	0.9