Big Data Mining Techniques

The objective of the exercise

The purpose of the work is to familiarize you with the basic stages of the process used to implement data mining techniques, namely: collection, preprocessing, cleaning, conversion, application of data mining techniques and evaluation. The implementation will be in the Python programming language using the SciKit Learn tool and the gensim library.

Description

The work is related to the classification of text data from news articles. The Dataset consists of CSV files whose fields are separated by the '\ t' (TAB) character. Two files are included:

1. train_set.csv (12267 data points): This file will be used to train your algorithms and contains the following fields:
    1. Id: A unique number for each document
    2. Title: The title of the article
    3. Content:The content of the article
    4. Category: The category of each document
2. test_set.csv (3068 data points): This file will be used to make predictions for new data points. The CSV file contains all fields of the training file except the 'Category' field. You have to estimate this using classification algorithms.

The article categories are 5 and are presented in the table below.

- Business
- Film
- Football
- Politics
- Technology

WordCloud Creation

At this point you have to create a Wordcloud for the five categories of articles with the most articles. To create a WordCloud for a particular category you will use all the articles of this category. An example of a WordCloud is shown in the following figure. You can use any Python library in order to create the WordClouds.

Duplicates Detection

Here you should find similar articles. In particular, the similarity between two articles will be measured using the cosine similarity between the term vectors of each article. Anyone who wants can use the LSH technique in order to quickly identify duplicates. Your code should accept a similarity threshold θ. Finally, all pairs of text with a similarity greater than 0.7 should be reported. The results will be stored in the file 'duplicatePairs.csv' and will have the following format:

Document_ID1	Document_ID1	Similarity

Classification

Here you have to test 2 classification Classification techniques:

- Support Vector Machines (SVM).
- Random Forests.

Also, you have to evaluate the performance of the above classification techniques using the following features:

- Bag of Words (BoW).
- SVD keeping the 90% of the total variance (SVD).
- Average word vector for each vector (W2V).

You should also evaluate and report the performance of each method using 10-fold Cross Validation using the following metrics:

- Accuracy
- Precision
- Recall

Beat the Benchmark

Finally, you should experiment with any Classification technique or approach you want, by doing any pre-processing to the data you want to overcome as much as possible the results achieved at your previous query. You should report and justify the steps you have taken.

Output Files

Your code should for the queries related to Classification should create the following files:

- EvaluationMetric_10fold.csv
- testSet_categories.csv
- roc_10fold.png

The format of the files EvaluationMetric_10fold.csv is shown below:

Statistic Measure	SVM (BoW)	Random Forest (BoW)	SVM (SVD)	Random Forest (SVD)	SVM (W2V)	Random Forest (W2V)	My Method
Accuracy							
Precision							
Recall							
F-Measure							
AUC							

The format of the file testSet_categories.csv, which should contain the categories of the articles that are given at the Test set is shown below:

Test_Document_ID	Predicted_Category
1	World News
2	Technology
...	

For the "testSet_categories.csv" file, the above formatting should be strictly used by separating the two fields with the TAB character ('\t') and the two headings (Test_Document_ID and Predicted_Category) should also be in the first line, and then the predictions of your model on the next lines should specify the document ID from the test set and the corresponding predicted category.