Sentiment Analysis

Your task in completing this assignment is to analyse a range of reviews for the most common words that appear for both positive and negative sentiments. The data are contained in a file called *sentiments.txt*, which you can download from the module assignment page on Canvas (a shorter version called *shortsent.txt* is also available for testing purposes). The file contains the type of item being reviewed (Restaurant, Movie, Product) followed by the review text and then a sentiment value (1 for positive, 0 for negative). Each review is on a single line of the file with the different fields separated by a tab character, as shown in the following example:

```
Restaurant I swung in to give them a try but was disappointed. 0
Restaurant I had a pretty satisfying experience. 1
Movie Some applause should be given to the "prelude". 1
Product A must study for anyone interested poor design. 0
```

Your task is to write a Java Hadoop Map/Reduce solution that will, *in a single pass*, find the 5 most common words associated with a given item type and sentiment. The result will be 6 rows of data consisting of the top 5 words for each item type and sentiment score in a form similar to the following (the format and order can be different but the words for each item/sentiment must be correct):

```
Restaurant 0 brother again law night eating
Restaurant 1 you'd any bean fry stir
Product 0 anyone must industrial study interested
Product 1 phone use restored simple performance
Movie 0 enter script watch unethical rated
Movie 1 however both superb rickman complex
```

In a similar manner to the practicals, you will be required to exclude common words in the reviews and the words that you should exclude are provided as a link on Canvas in the file *exclude.txt.* You could start by hard coding some of these words into your Mapper code but you are eventually expected to load them from a cache file when the program runs.

Along with your code, you should also submit a short written report, detailing your design and the results you found.

Step 1, HDFS – 20 Marks

Before you write any code, you will need to copy the data onto your own space in HDFS. In your report, give details of how HDFS stores data such as this (assume the file is much bigger than it really is for the purpose of your description). This section should be around half a page long, plus a diagram. Describe what HDFS is for, the architecture it uses, and the roles of different nodes in the cluster.

Document all the hdfs commands you used to create a directory for the data and place it your data on HDFS ensuring you explain why the data must be uploaded in this way. Make sure everything you put here, including the diagram, is your own work. Do not copy anything from other sources.

Step 2, Design – 20 Marks

Now consider the Map/Reduce design you will implement. You know there are only six different results that must be produced and a larger (but unknown) number of different words used in those reviews. In your report, consider and compare two different choices you could make to implement the given task. What keys and values will the mapper emit? Consider how much data will be moved across the network in each of your two designs. Also consider how many different reducers will be used in each case. Finally, choose the more efficient design, implement it and justify your choice.

Step 3, Implement – 60 Marks

Using the SentimentWords.java file provided on the assignment page in Canvas as a starting point, modify this code to produce the results requested above. This code is just a revised version of the original WordCount.java file from your practicals and will need significant changes to meet the desired requirements. It is supplied with the file TestSentiment.*java* that you can use with the Hadoop simulator *mochadoop* to check your logic on the smaller set of data found in shortsent.txt.

Your final output should however be produced from the full *sentimentss.txt* file that should be run on the Hadoop server.
You should implement your design in Java using the Hadoop API that we have been using in class. Your code should find the 5 most commonly used words (excluding those in the exclusion list) for each of the 3 item types and sentiment values. Once you have completed your code and run it for the final time, please include the results that are produced at the end of your report .