Search Engine for Movie Plot Summaries

We will work with a dataset of movie plot summaries that is available from the Carnegie Movie Summary Corpus site. We are interested in building a search engine for the plot summaries that are available in the file “plot summaries.txt” that is available under the Dataset link of the above page.

You will use the tf-idf technique studied in class to accomplish the above task. For more details on how to compute tf-idf using MapReduce, see the links below:

1. Good introduction from Coursera Distributed Programming course
2. Chapter 4 of the reference book Data-Intensive Text Processing using MapReduce.

This assignment has to be done using Scala/PySpark code that can run on a Databricks cluster. Remember that you have to write your code in the form of Scala/PySpark code that can run on Databricks.

Below are the stepwise details of the project:

1. Extract and upload the file **plot summaries.txt** from http://www.cs.cmu.edu/~ark/personas/ data/MovieSummaries.tar.gz to Databricks. Also upload a file containing user’s search terms one per line.
2. You will need to remove stopwords by a method of your choice.
3. You will create a tf-idf for every term and every document (represented by Wikipedia movie ID) using the MapReduce method.
4. Read the search terms from the search file and output following:
   (a) User enters a single term: You will output the top 10 documents with the highest tf-idf values for that term.
   (b) User enters a query consisting of multiple terms: An example could be “Funny movie with action scenes”. In this case, you will need to evaluate *cosine similarity* between the query and all the documents and return top 10 documents having the highest cosine similarity values.
   You can read more about cosine similarity at the following resources:
   - http://text2vec.org/similarity.html : *Read the cosine similarity section*
   - https://janav.wordpress.com/2013/10/27/tf-idf-and-cosine-similarity/
   - https://courses.cs.washington.edu/courses/cse573/12sp/lectures/17-ir.pdf

   For the search terms entered by the user, you will return the list of movie names sorted by their relevance values in descending order. Note again, that you have to return movie names, and not movie ID. You would need to use the **movie.metadata.tsv** file to lookup the movie names. The search queries used in the program are in **queries-1.csv**
5. You can display output of your program on the screen.