UNIVERSITY OF CRETE, COMPUTER SCIENCE DEPARTMENT

CS-562 Advanced Topics in Databases
Report Assignment 1

Iacovos Kolokasis
(kolokasis@csd.uoc.gr)
AM:1039

October 22, 2017

1 FREQUENT TERMS AND STOP WORDS

In this exercise, I implement a program that is calculate the frequency of each word, found inside the word datasets after removing the non-alpanumeric terms, and find the stop words. The first job is similar with word count example. Acctually the reducer produce an output file of the format <word, frequency>, where word is the key and the frequency is the value. The next job, use a mapper which take the output of the previous job, and use now as key the frequency of each word and as value the word. Using a comparator, sort the words by frequency in descending order and export the output. From the output we collect the words with frequency grater tha 4000.
The top ten most frequent words among with their frequency are shown in Table 1.1 :

Word	Frequency	Word	Frequency
the	184056	i	66407
and	148816	in	59715
of	99252	it	56849
to	91657	that	51213
a	87368	was	41832

Table 1.1: Top Ten frequent words

2 Measuring The Performance of Map Reduce

1. Using 10 reducers without combiner the job executed $29427ms \simeq 29.42s$.
2. Using 10 reducers with combiner the job executed $21048ms \simeq 21.05s$. The execution time is reduced and this is due to the cobiner that helps segregating data into multiple groups for Reduce phase, which makes it easier to process.
3. Using 50 reducers without combiners the job executed $37213ms \simeq 37.21s$. The excution time is larger than the previous. This due to the reducers number. The big number of reducers causes overhead to our system, because the datasets devide to unmballanced partitions.

3 Variation of an Inverted Index

We implement an inverted index for the documents words dataset. From these words, we excluded the alpanumeric terms and the stop words that we compute in the previous exercises. Well the total unique words excluded stopwords calculated by the number of reduce input groups and equalls to 90868. In order to calculate the words that appear only in one documents we set a global counter, defined either by the Map-Reduce framework or applications. Counter is an Enum type and defined in a seperate class. We increment the counter in reduce task if the word appear only in one document and from Driver class we get the value. As a result from the counter we calculate the total words that appear only in one document to be 73661.