HW 4 Problem 1

(a)

Mapper output

n 2.0
n 3.0
i 2.0
d 10.0
n 3.0
t 3.0
b 4.0
t 4.0

reducer out put

n 2.5
b 4.0
d 10.0
i 2.0
n 3.0
t 3.5

(b)

There are not any differences in the job execution, since we didn't add any parameters.

(c)

I use boolean caseSensitive=true to set case sensitive as my default. There are not any differences in the job execution, since we didn't add any parameters.

(d)

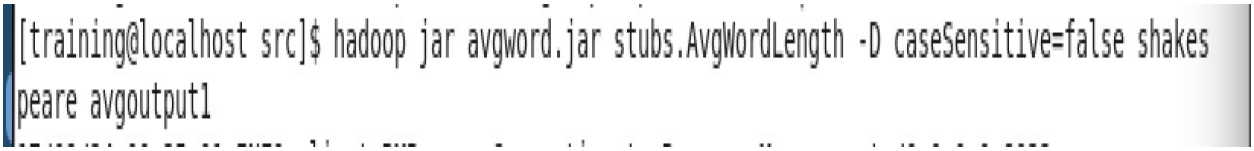
- A: 3.891394576646375
- W: 4.464014043300176
- a: 3.0776554817818575
- t: 3.733261651336357
- z: 4.672727272727273

(e)

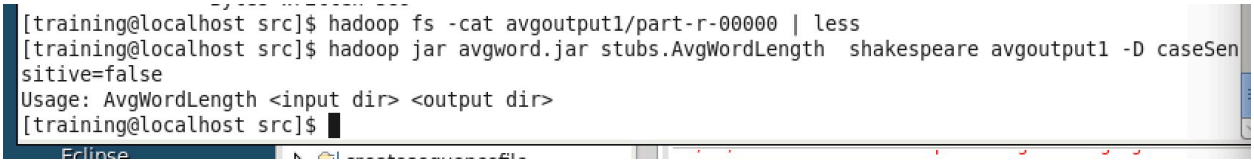
- a: 3.275899648342265
- w: 4.373096283946263
- z: 5.053333333333334

the command line:

hadoop jar avgword.jar stubs.AvgWordLength -D caseSensitive=false shakespeare avgoutput1



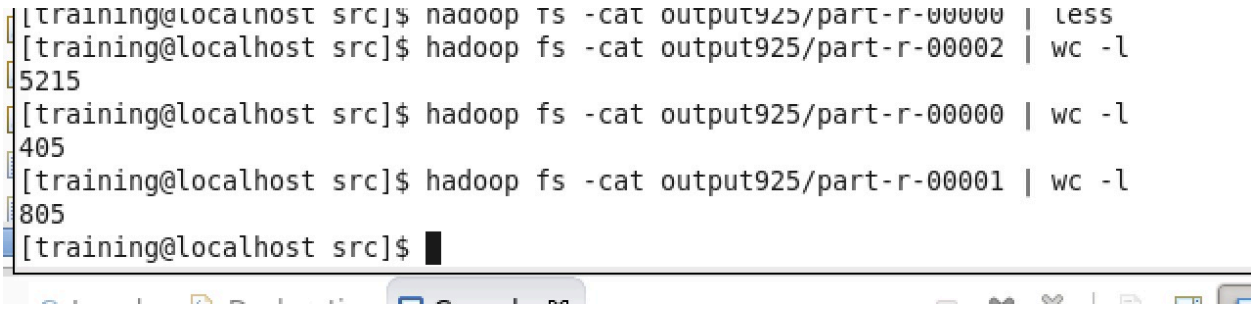
the order of command line matters since:



so we need pass the parameter before the input and output dorectory.

HW 4 Problem 2

(c)



As the output result shows, there are 405 positive words and 805 negative words used in the poems, so:

The sentiment score s = (405-805)/(405+805) = -0.3306

The positive score p = 405/(405+805) = 0.3361

Since the sentiment score is smaller than zero, so I think the Shakespeare's poems suggest negative emotion.

(d)

- This is not a good way to measure the emotion in the poems, because in the real sentences, there are many phrase and fixed matches of words, so it's tough to reveal the real meaning that the author want to express by just checking each single word.
- Some words may have many meanings in different environment, so it is not so convincing to simply divide the words into positive and negative words.

I think the mapper should can identify some phrases that have special emotions instead of just use the single words as keys.

While, the words should be judged according to the sentence, for example, we should pay more attention on the words such as "but, however, while" then determine its sentiment.