9313 Ass1 Report **Name**: PEIGUO GUAN **zID**: z5143964

Package assignment1

Class **Assignment1**

Includes three class:

Class TokenizerMapper:

1. The type is <Object, Text, Text, Text>
2. In map function:
   a. Using `context.getConfiguration().get("NumOfgram")` to get parameter Number of Gram parameter
   b. Using `context.getInputSplit()).getPath().getName()` to get currently processed file name
   c. When mapper read the line, split the line and set the key as the n-gram words Using for – for to get the ngram that we need.
   d. Then the ngram words appear once and combine the string "1" + string filename in the Text style and transfer to the Reducer

Class IntSumReducer:

1. The type is <Text,Text,Text,Text>
2. Using `context.getConfiguration().get("MinCount")` to get the Min Count parameter
3. Split the input value and the `values[0]="1" values[1] =filename` sum up the count and get the total number of the ngrams words appeared.
4. Put all the file name in a String and split it, filter the file name and if the file name exit, ignore it.
5. Using `Collections.sort(list);` to let the file in a alpha order.
6. Combine the total number and file name list to Text and write in the output file
7. There is a filter, only the sum is >= min count can be wrote.

Class cleanup:(ignore it please)

```
// I just write this part which is the same result as filter
in reduce
// Through it i know the use of clean up
// this part is new knowledge for me, so i keep it
// please ignore it
```

Class Main:(4 parameter: args[0]->ngram, args[1]->minCount, args[2]->InputPath(I using Absolute Path), args[3]->OutputPath(using Absolute Path))

1. Using
   ```
   conf.set("NumOfgram", args[0]);
   conf.set("MinCount", args[1]);
   ```
   to transfer parameter in mapreduce
2. Initial the path is on Assignment1.class path
3. Set Mapper and Reducer class
4. Not using combiner, it is not correct in this assignment

Note: Some other detail has been shown in Assignment1.java comment.