**Brief Report about Mahout Project**

Project 3 requires us to come up with a solution to use naïve Bayes classifier to predict profession(s) of different people, and then evaluate the accuracy rate.  Here is our brief explanation about our approach to achieve this goal.

**Short Description of what we achieved**

In short, what we did could be divided to three parts: preprocessing, creating models, processing the data with the models, and evaluating the accuracy.

       First, we created a new file to hold the records by preprocessing the ARTICLE LEMMA INDEX. In order to take the supervised learning strategy, we needed the elements from ARTICLE LEMMA INDEX for which we knew what the professions were. Therefore, we implemented a function to only include the people who were both in professions.txt and ARTICLE LEMMA INDEX, and then generated records as the following format: [profession] [Article Name] [list of words].

       Second, we created a Naive Bayes classifier statistical model, using the records generated by the previous step. Before creating the model, we converted the previous records to a sequence file in the form: /[profession]/[Article Name] [list of words]. Afterwards, we used the command line shown in our shell script file to create the model. As far as here, we accomplished the first step required in the assignment prompt.

     Third, we ran this model over the rest of the data, which was in the ARTICLE LEMMA INDEX but not in the professions.txt. This only gave us some records as [Article Name] [List of profession IDs]. Thus we need to run the result reader in order to acquire some human readable file. After running the result reader, it gave us the classification results required by the second step from the problem prompt, such as [Article Id] [List of professions]

      Last but not least, we used the first 600 thousand lines from professions.txt to form a training set, and put the rest of the lines into the testing set. Particularly, we built the model with the training set as how we did in the previous steps, and then test this model over the testing set. If at least one of the three predicted professions matched the profession(s) of a person in professions.txt, we concluded this prediction was correct. As for the accuracy results, please check our demonstrated evaluation in the latter part of this report.

Remark: when you are testing model over the testing set, please don’t convert the testing set into sequence file.

**Pointers on how to run our code**

Preparation

    1.     We built a maven project including our source code files and pom.xml. In particular, since we used fat jar, we also included the jar files of mahout-math 0.11, mahout-hdfs 0.11, mahout-mr 0.11 and lucene analyzers- common 4.6.0.

    2.     We uploaded professions.txt to the HDFS.

Preprocess

1. We conducted mapreduce to only include the people who were both in professions.txt and ARTICLE LEMMA INDEX, and then generated records as the following format: [profession] [Article Name] [list of words]. Here are command line needed to use in the terminal: yarn jar [jar file path] [input] [output]

     2.  We put the result file from last step back to local home directory, so that we could convert this result file to a sequence file on cluster locally. The command was yarn jar [package] [input] [output].The input here was the output of the last step. Since the output from last step was a directory, we specified the [input] as something like  [path]/part-r-00000. After conversion was done, we put the output file from this step back to hdfs for the next step.

Process

   1.  Creating the statistical model:

We used command line to create the model which are all included in our    assignment3\_create\_model.sh file.

   2.  Classifying the rest:

In a MapReduce Job, we ran the Classifier to classify each person in the rest part. After running the classifying job, we merged the output to a tsv file by using hdfs dfs -getmerge command. This merged output was automatically put in the local home directory by the system. And then we ran Result Reader job on the local home directory of cluster. This transformed the output format from [article Name] [List of professions IDs] to [article Name] [List of professions] by using label index.

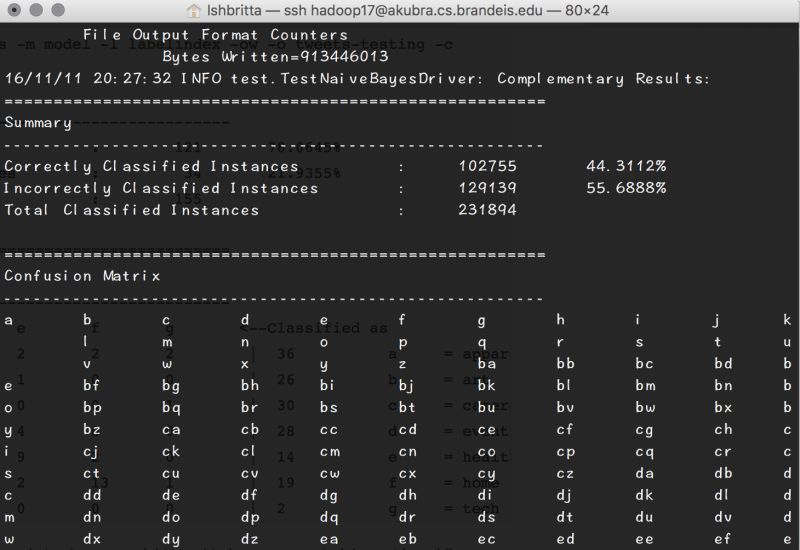
Remark: label index should be put in the local directory before running result reader

   3.  Evaluation of the classifier:

We split professions.txt into a training set with the first 600 thousand lines, and a test set with the rest of the lines. Then we did the same as step 1 and 2(Run the evaluation locally), after we modelized the training data and classified the test data. We wrote a java program to compare the classified data with the rest of training data in professions.txt list.

**Evaluation results**

Before conducting the mapreduce of classifier, the automatic evaluation from the mahout on our model was about 44% correct. That was because without the adjustment of classifier, the prediction output only provided one recommended profession per article name. However, some of the people may have more than one profession.



The self-testing was about 89% correct, while the testing on holdout set was 77% correct. The precise rate was improved because at least one of the three predicted professions matched the profession(s) of a person in professions.txt