LING 570: Hw8
Due on Nov 25

The goal of this assignment is to use the Mallet package for the text classification task. All the data files are under dropbox/09-10/570/hw8/. Let $dataDir be hw8/20_newsgroups, and $exDir be hw8/examples/. Note:

- When you type the commands, you need to replace $dataDir with hw8/20_newsgroups and $exDir with hw8/examples.
- All the options of Mallet commands (e.g., "--input") start with two "-"s, not one "-".
- Use the Mallet package on Patas, which is the correct version for this assignment.

Q1: Learning the Mallet commands (40 points)

(a) 5 points: Read the Mallet command-line totorial at /NLP_TOOLS/tool_sets/mallet/latest/doc/command-line-classification.html on Patas.

(b) 1 point: Run the following command to create a data vector, news3.vectors, using the data from the three talk.politics.* newsgroups:

```
text2vectors --input $dataDir/talk.politics.* --skip-header --output news3.vectors
```

(c) 2 points: Run the following command to convert news3.vectors to the text format, news3.vectors.txt, and then convert it back to the binary format.

```
vectors2info --input news3.vectors --print-matrix siw > news3.vectors.txt
```

```
info2vectors --input news3.vectors.txt --output news3.vectors.new
```

(d) 2 points: Run the following command to split news3.vectors into training (90% of the data) and testing files (10% of the data):

```
vectors2vectors --input news3.vectors --training-portion 0.9 --training-file train1.vectors --testing-file test1.vectors
```

(e) 15 points: Run vectors2classify to classify the data with five learners and complete Table 1.

- Use the train.vectors and test.vectors **under $exDir** for this classification task.
- The names of the five learners are: NaiveBayes, MaxEnt, DecisionTree, Winnow, and BalancedWinnow.
- The command for classification is:

  ```
  vectors2classify --training-file $exDir/train.vectors --testing-file $exDir/test.vectors --trainer $zz > $zz.stdout 2>$zz.stderr
  ```

  whereas $zz is the name of a learner (e.g., MaxEnt).

(f) 5 points: What conclusion can you draw from Table 1?

(g) 5 points: Write down the command lines for running the MaxEnt trainer with vectors2train and classify. Do you get the same results as in (e)?

(h) 5 points: Use the classifier2info command to convert the model created in Step (g) to the text format. Write down the command line.

Table 1: Classification results for Q1(e)

	Training accuracy	Test accuracy
NaiveBayes		
MaxEnt		
DecisionTree		
Winnow		
BalancedWinnow		

Q2: Creating attribute-value table (60 points)

(a) 20 points: Write a script, **proc_file.sh**, that processes a document and prints out the feature vectors.

- The command line is: proc_file.sh input_file targetLabel output_file
- The input_file is a text file (e.g., **input_ex**).
- The output_file has only one line with the format (e.g., **output_ex**):
  instanceName targetLabel f1 v1 f2 v2 ....
  - The instanceName is the filename of the input_file.
  - The targetLabel is the second argument of the command line.
- To generate the feature vector, the code should do the following (see Slide #33-35 in 11_18_Mallet.pdf) :
  - First, skip the header; that is, the text before the first blank line should be ignored.
  - Next, replace all the chars that are not [a-zA-Z] with whitespace, and lowercase all the remaining chars.
  - Finally, break the text into token by whitespace, and each token will become a feature.
  - The feature values will be the frequency of the sequences.
  - The (featname, value) pairs are ordered by the spelling of the featname.
- For instance, running "proc_file.sh $exDir/input_ex c1 output_ex" will produce output_ex as the one under the $exDir.

(b) 20 points: Write a script, **create_vectors.sh**, that creates training and test vectors from several directories of documents. This script has the same function as text2vectors, except that the vectors produced by this script are in the text format and the training/test split is not random.

- The command line is: create_vectors.sh train_vector_file test_vector_file ratio dir1 dir2 ...
  That is, the command line should include one or more directories.
- ratio is the portion of the training data. For instance, if the ratio is 0.9, then the FIRST 90% of the FILES in EACH directory should be treated as the training data, and the remaining 10% should be treated as the test data.
- train_vector_file and test_vector_file are the output files and they are the training and test vectors in the text format (the same format as the output_file in Q2(a)).
- The class label is the basename of an input directory. For instance, if a directory is hw8/20_newsgroups/talk.politics.misc, the class label for every file under that directory should be talk.politics.misc.

(c) 10 points: Classify the documents in the talk.politics.* groups under $dataDir.

Table 2: Classification results for Q2(c)-(e)

	Training accuracy	Test accuracy
(3) three talk.politics.* groups		
(4) four sci.* groups		
(5) four rec.* groups		

- Run create_vectors.sh from Q2(b) with the ratio being **0.9**, and the directories being talk.politics.guns, talk.politics.mideast, and talk.politics.misc.
- Run **info2vectors** to convert the vectors to the binary format, **vectors2train** for training (with MaxEnt trainer) and **classify** for testing.
- Suppose you run info2vector on train_vector_file and create train.vectors. When you run info2vectors for the test_vector_file, remember to use the option "--use-pipe-from train.vectors". That way, the two vector files will use the same mapping to map feature names to feature indexes.
- Save all the files (the vectors in text format and binary format, the MaxEnt model, the classification output) under a directory called **q2c**.
- What are the training and test accuracy?

(d) 5 points: The same as Q2(c), except that you will use the four sci.* groups under $dataDir. Save the files under a directory called **q2d**.

(e) 5 points: The same as Q2(c), except that you will use the four rec.* groups under $dataDir. Save the files under a directory called **q2e**.

Fill out Table 2 with the results from (a)-(e).

Submission: In your submission, include the following:

- Shell scripts for proc_file.sh and create_vectors.sh, and the code called by the shell scripts.
- The directories q2c, q2d and q2e created in Q2(c)-(e).
- Completed Tables 1 and 2.
- The answers to (e)-(h) in Q1.
- No need to submit anything for (a)-(d) in Q1.