CSC 555 Mining Big Data

Assignment 5

Due Tuesday, 3/6

Suggested Reading: Hadoop: The Definitive Guide Ch19; Mining of Massive Datasets: Ch9

1)

a) Solve 9.3.1-a, 9.3.1-e

	a	b	c	d	e	f	g	h
A	4	5		5	1		3	2
B		3	4	3	1	2	1	
C	2		1	3		4	5	3

Figure 9.8: A utility matrix for exercises

Exercise 9.3.1: Figure 9.8 is a utility matrix, representing the ratings, on a 1–5 star scale, of eight items, a through h, by three users A, B, and C. Compute the following from the data of this matrix.

(a) Treating the utility matrix as boolean, compute the Jaccard distance between each pair of users.

(e) Normalize the matrix by subtracting from each nonblank entry the average value for its user.

2)

a) Where does Spark typically read the data from (and how does it ensure that data is not lost when a failure occurs)?

b) What is the difference between content-based and collaborative filtering based recommender systems?

c) Describe a strategy that is used to make a utility matrix less sparse

3)

a) Add one more node to your existing cluster (e.g., go from 3 to 4 nodes) following the instructions from the previous assignment and examples in class. You can do that by creating a new AWS instance, setting up ssh access (public-private key) and copying Hadoop to that new instance as you have with two other workers before. Keep in mind that you do not need to configure anything again except for editing the slaves file to

reference the new worker node private IP. Everything else should be taken care of by your already existing cluster setup.
Submit a screenshot of the new cluster view.

b) Pick one of the hadoop streaming tasks (from this or previous homework) and run it as-is on the new cluster. Record the time it took (you can time a command by prepending it with time, e.g., time hadoop jar…). You do not need to write any new code, just time one of your existing examples.

c) Repeat the previous task (3-b), but shut down one of the nodes (from Amazon console, imitating a failure) **while the task** is running. Record the time it took. Was it slower or the same? Why or why not?

d) Finally, modify one of the configuration files in Hadoop to introduce a typo (you can do that on the cluster or on the single-node setup) so it produces an error when you start dfs or yarn. Take a screenshot of the modified config file and the corresponding error message that you received.

4) Run a recommender on the MoveLens dataset.

(Create a directory for movie lens dataset)
mkdir MovieLens
cd MovieLens
wget http://rasinsrv07.cstcis.cti.depaul.edu/CSC555/ml-1m.zip

(Unzip the dataset, this one happens to be compressed with Zip rather than GZip)
unzip ml-1m.zip
cd ..

Take a look at the data file:
more MovieLens/ml-1m/ratings.dat
(you can press q or Ctrl-C to exit, more command shows the first few lines worth of text. Each line contains user ID, movie ID, user rating and the timestamp of the rating, as already discussed in class)

The next step is to use aa Linux command to convert :: separated file into a comma-separated file. First part (cat) will simply output the file. Second part substitutes , for :: and third part of the command extracts just 3 attributes relevant to us (no timestamp)
cat MovieLens/ml-1m/ratings.dat | sed -e s/::/,/g| cut -d, -f1,2,3 > MovieLens/ml-1m/ratings.csv

(NOTE: if you wanted to extract all 4 columns from the original data set, you could run the same command with “1,2,3,4” instead of “1,2,3”).

Create a movielens directory and copy the articles over to HDFS into that directory:
$HADOOP_HOME/bin/hadoop fs -mkdir movielens
$HADOOP_HOME/bin/hadoop fs -put MovieLens/ml-1m/ratings.csv movielens

Split the data set into the 90% training set and 10% evaluation set. In this case we are using Hadoop to perform the split. Naturally, you can change the percentages here to any

other value instead of 0.9/0.1. bin/mahout will only work from the $MAHOUT_HOME directory, or you can change it as others.
bin/mahout splitDataset --input movielens/ratings.csv --output ml_dataset --trainingPercentage 0.9 --probePercentage 0.1 --tempDir dataset/tmp

Verify and report the file sizes of the input ratings.csv file and the two sampled files (the two files are in the /user/ec2-user/ml_dataset/trainingSet/ and /user/ec2-user/ml_dataset/probeSet directories on HDFS side). Do the sampled file sizes add up to the original input file size?

Factorize the rating matrix based on the training set. As always, this is a single line command, be sure to run it as such. The --numfeatures value configures the set of "hidden" variables or the dimension size to use in matrix factorization. --numIterations sets how many passes to perform; we expect a better match with more iterations
time bin/mahout parallelALS --input ml_dataset/trainingSet/ --output als/out --tempDir als/tmp --numFeatures 20 --numIterations 3 --lambda 0.065

Measure the prediction against the training set:
bin/mahout evaluateFactorization --input ml_dataset/probeSet/ --output als/rmse/ --userFeatures als/out/U/ --itemFeatures als/out/M/ --tempDir als/tmp

What is the resulting RMSE value? (rmse.txt file in /user/ec2-user/als/rmse/ on HDFS)

Finally, let's generate some predictions:
bin/mahout recommendfactorized --input als/out/userRatings/ --output recommendations/ --userFeatures als/out/U/ --itemFeatures als/out/M/ --numRecommendations 6 --maxRating 5

Look at recommendations/part-m-00000 and report the first 10 lines by running the following command. These are top-6 recommendations (note that --numRecommendation setting in the previous command) for each user. Each recommendation consists of movieID and the estimated rating that the user might give to that movie.

$HADOOP_HOME/bin/hadoop fs -cat recommendations/part-m-00000 | head

What is the top movie recommendation (movie ID) for users 3, 4 and 5?

5) Extra Credit: Set up stand-alone minimum 3-node Spark cluster (will discuss during Lecture9 in in class, instructions available at http://spark.apache.org/docs/latest/spark-standalone.html). Note that you can use your existing cluster, you just need to configure Hadoop-env.sh and add slaves file to the conf directory in spark folder. Browser page is at port 8080.

Submit a single document containing your written answers. Be sure that this document contains your name and "CSC 555 Assignment 5" at the top.