Assignment 5

MET CS 777 - Big Data Analytics

Using Spark MLlib:
Logistic Regression, SVM, and Features Selection (20 points)

GitHub Classroom Invitation Link

https://classroom.github.com/a/d5tFrqJ5

1 Description

In this assignment, you will use Spark's MLlib to implement logistic regression and support vector machines, and compare the accuracy and computation time. At the end of your classification task, you will use a feature selection method to reduce the problem's dimensionality and redo the classification tasks.

2 Data

You will be dealing with a data set consisting of around 170,000 text documents (this is 7.6 million lines of text in all) and a test/evaluation data set consisting of 18,700 text documents (almost exactly one million lines of text in all).

Your task is to build a classifier that can automatically determine whether a text document is an Australian court case. We have prepared four data sets for your use.

1. The Training Data Set (1.9 GB of text - 520MB compressed). This is the set you will use to train your logistic regression model.
2. The Testing Data Set (200 MB of text, 57MB compressed). This is the set you will use to evaluate your model.
3. The Small Data Set (37.5 MB of text 10MB compressed). This is for you to use for training and testing your model locally before you try to do anything in the cloud.
4. The Small Testing Data Set to use when testing your model.

Public links

- https://storage.googleapis.com/met-cs-777-data/TestingData.txt.bz2
- https://storage.googleapis.com/met-cs-777-data/TrainingData.txt.bz2
- https://storage.googleapis.com/met-cs-777-data/SmallTestingData.txt.bz2
- https://storage.googleapis.com/met-cs-777-data/SmallTrainingData.txt.bz2

Google Cloud Storage

- gs://met-cs-777-data/TestingData.txt.bz2
- gs://met-cs-777-data/TrainingData.txt.bz2
- gs://met-cs-777-data/SmallTestingData.txt.bz2
- gs://met-cs-777-data/SmallTrainingData.txt.bz2

Data Details: You should download and look at the SmallTrainingData.txt.bz2 file before beginning. You'll see that the contents are sort of a pseudo-XML, where each text document begins with a < *doc id* **=** ... > tag, and ends with < */doc* >.

Note that all Australian legal cases begin with something like < doc id = "AU1222" ... > that is, the doc id for an Australian legal case always starts with AU. You will be trying to figure out if the document is an Australian legal case by looking only at the document's contents.

3 Assignment Task

Task 1. Vectorize the data (5 points)

Your goal is to vectorize the text documents using the available functions in the MLlib in the format suitable for MLlib machine learning models. First, you need to remove the stop words from the text documents and then vectorize the documents using CountVectorizer and TF-IDF with a vocabulary size of 5000.

Subtasks:

1. Print the first ten words of the vocabulary
2. Print out the total time needed to vectorize the data.

Note:

- You need to use the newer MLlib, so your data needs to be in Data Frames
- Cache the resulting Data Frame because it will be used in the following tasks

Task 2: Using the Logistic regression model (5 points)

You need to write Spark code that uses the spark MLlib (Dataframe based) to train a Logistic regression model using the dataset prepared in task 1.

Subtasks:

1. Set the max number of iterations to 20
2. Train the model on the train data
3. Evaluate the model on the test dataset and print the F1-measure the confusion matrix
4. Print out the total time needed to train the model, evaluate the model using the test dataset, and calculate the performance metrics.

Task 3: Using the SVM model (5 points)

You need to write Spark code that uses the spark MLlib (Dataframe based) to train an SVM model using the dataset prepared in task 1.

Subtasks:

1. Set the max number of iterations to 20
2. Train the model on the train data
3. Evaluate the model on the test dataset and print the F1-measure the confusion matrix
4. Print out the total time needed to train the model, evaluate the model using the test dataset, and calculate the performance metrics.

Task 4: Feature Selection (5 points)

Apply the feature selection technique to reduce the data dimensions from 5K to 200 dimensions. You can use some of the available Feature Selectors from MLlib or implement your own. Describe the selected feature selection technique and discuss why you chose the specific technique. Discuss if your dimension reduction approach is applicable to very large data sets.

Subtasks:

1. Apply the feature selection method to the dataset from task 1
2. Repeat tasks 2 and 3 with reduced features and report the F1 measure, confusion matrix, and computation time.
3. Comment on the differences in the results

Note: If you decide to implement your own features selection or dimension reduction technique, you can start with a very simple Random Selection method or Random Projection method. These are simple and computationally efficient ways to reduce the dimensionality of the data by trading a controlled amount of accuracy (as additional variance) for faster processing times and smaller model sizes.

Important Considerations

4.1 Machines to Use

One thing to be aware of is that you can choose virtually any configuration for your Cloud Cluster

- you can choose different numbers of machines and different configurations of those machines. And each is going to cost you differently! Since this is real money, it makes sense to develop your code and run your jobs locally, on your laptop, using the small data set. Once things are working, you'll then move to Cloud.

As a proposal for this assignment, you can use the **n1-standard-4** or **e2-standard-4** machines on the Google Cloud, one for the Master node and two for worker nodes.

Remember to delete your cluster after the calculation is finished!!!

More information regarding Google Cloud Pricing can be found here https://cloud.google.com/products/calculator. As you can see average server costs around 50 cents per hour. That is not much, but **IT WILL ADD UP QUICKLY IF YOU FORGET TO SHUT OFF YOUR MACHINES**. Be very careful, and stop your machine as soon as you are done working. You can always come back and start your machine or create a new one easily when you begin your work again. Another thing to be aware of is that Google and Amazon charge you when you move data around. To avoid such charges, do everything in the ”Iowa (us-cental1)” region. That's where data is, and that's where you should put your data and machines.

- You should document your code very well and as much as possible.
- Your code should be compilable on a Unix-based operating system like Linux or macOS.

4.2 Academic Misconduct Regarding Programming

In a programming class like ours, there is sometimes a very fine line between ”cheating” and acceptable and beneficial interaction between peers. Thus, it is essential to fully understand what is and what is not allowed in collaboration with your classmates. We want to be 100% precise, so there can be no confusion.

The rule on collaboration and communication with your classmates is very simple: you cannot transmit or receive code from or to anyone in the class in any way—visually (by showing someone your code), electronically (by emailing, posting, or otherwise sending someone your code), verbally (by reading code to someone) or in any other way we have not yet imagined. Any other collaboration is acceptable.

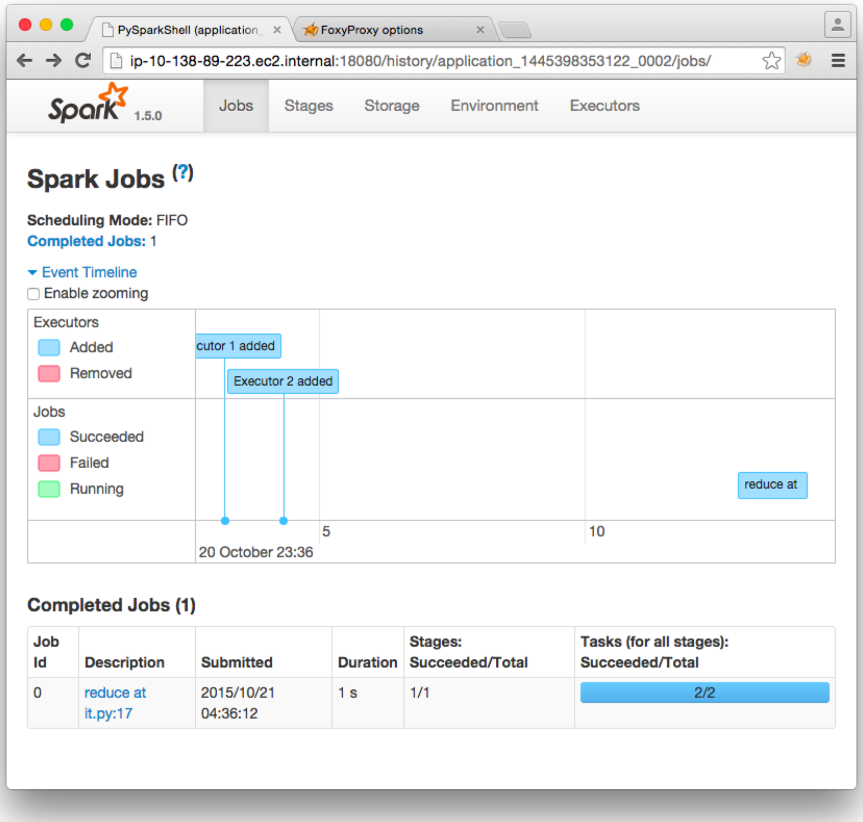
The rule on collaboration and communication with people who are not your classmates (or your TAs or instructor) is also very simple: it is not allowed in any way, period. This disallows (for example) posting any questions of any nature to programming forums such as **StackOverflow**. As far as going to the web and using Google, we will apply the **”two-line rule”**. Go to any web page you like and do any search that you like. But you cannot take more than two lines of code from an external resource and actually include it in your assignment in any form. Note that changing variable names or otherwise transforming or obfuscating code you found on the web does not render the ”two-line rule” inapplicable. It is still a violation to obtain more than two lines of code from an external resource and turn it in, whatever you do to those two lines after you first obtain them.

Furthermore, you should cite your sources. Add a comment to your code that includes the URL(s) that you consulted when constructing your solution. This turns out to be very helpful when you're looking at something you wrote a while ago and you need to remind yourself what you were thinking.

4.3 Turnin

Create a single document that has results for all three tasks.

Also, for each task, for each Spark job you ran, include a screenshot of the Spark History.



Spark Jobs (?)

Scheduling Mode: FIFO
Completed Jobs: 1

Completed Jobs (1)

Job Id	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
0	reduce at it.py:17	2015/10/21 04:36:12	1 s	1/1	2/2

Figure 1: Screenshot of Spark History

Please zip up all of your code and your document (use .zip only, please!), or else attach each piece of code and your document to your submission individually.

Please have the latest version of your code on GitHub. Zip the GitHub files and submit your latest version of assignment work to Blackboard. We will consider the latest version of the Blackboard.