**Machine Learning Algorithms -Part2:**

**Tasks:**

1. Build different models to predict whether a tweet is positive, neutral or negative, and if it is negative, give the type of the reason of it being negative.
2. Base on the run time and accuracy of different model, choose the best model for this particular task.
3. Avoid overfitting.

**Model:**

Naïve Bayes:

Naïve Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. It is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable.

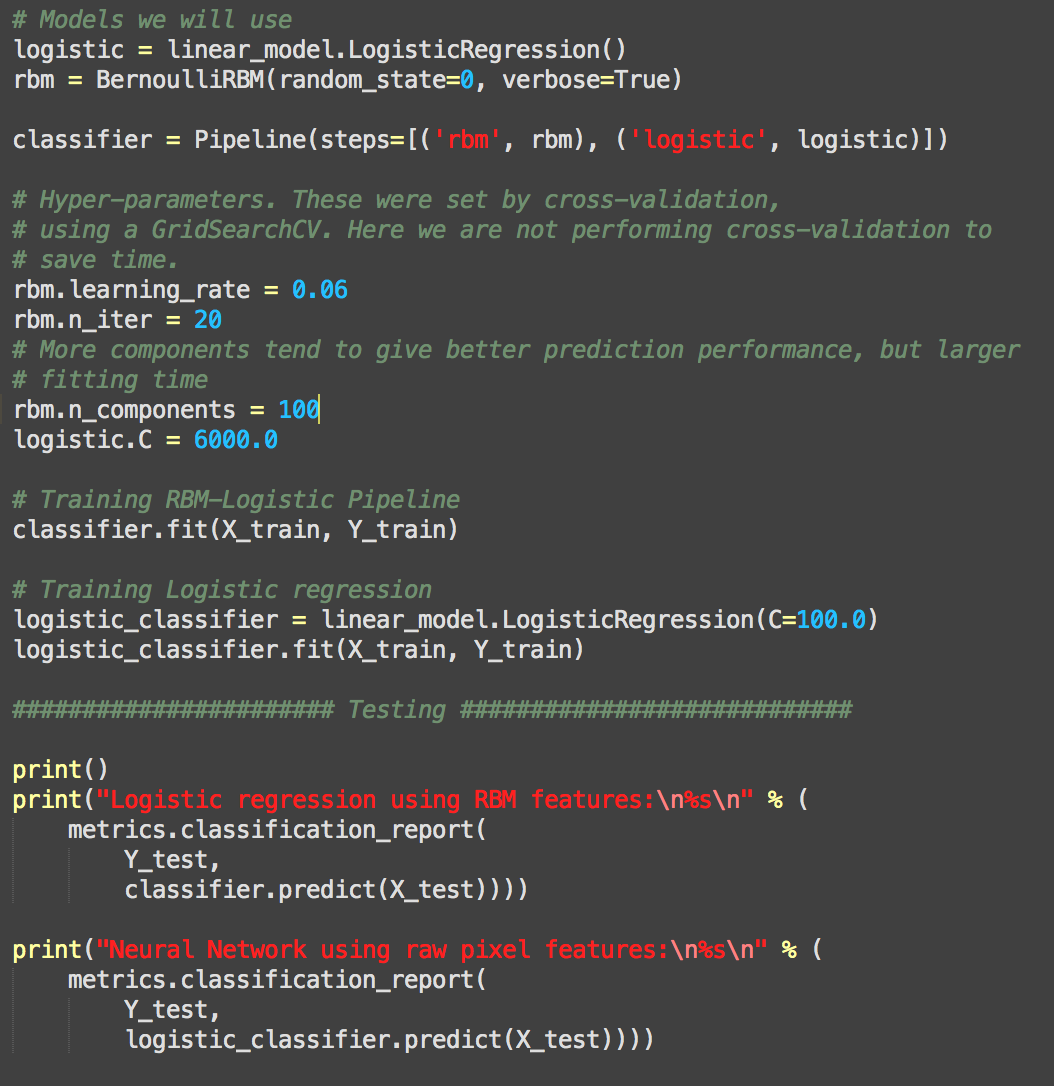
Neural Network:

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurones) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurones. This is true of ANNs as well.

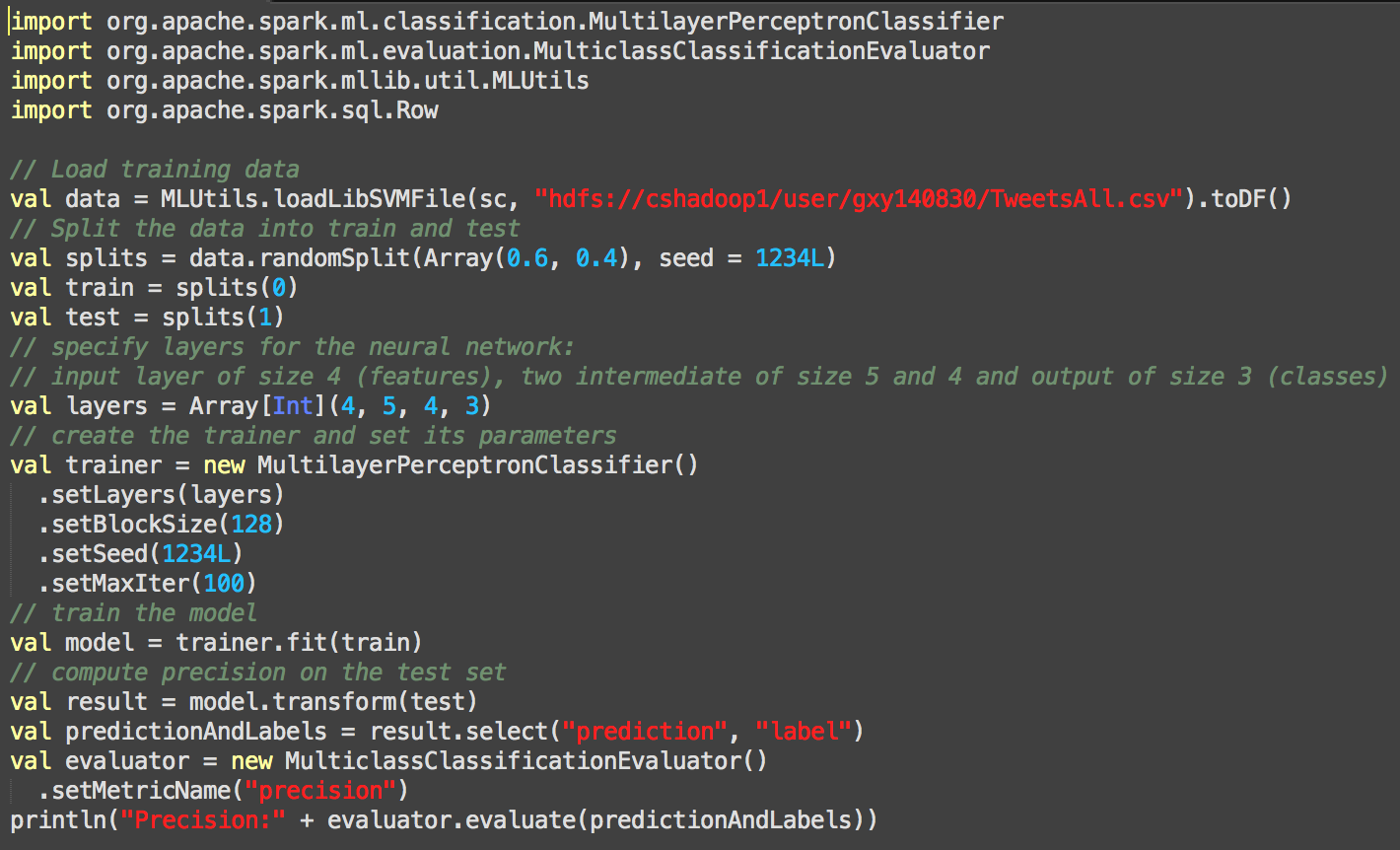
**Technique:**

We used scikit-learn package(sklearn.neural\_network) and Spark Multilayer perceptron classifier to compare the result of neural network.

sklearn.neural\_network package:

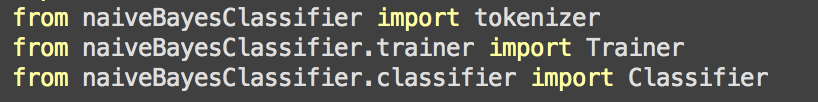


Spark Multilayer perceptron classifier:



We used naiveBayesClassifier package and MLlib Naïve Bayes to compare the result of Naïve Bayes.

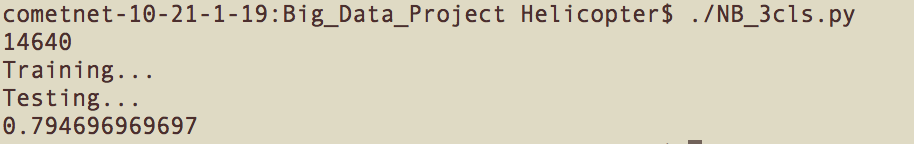
naiveBayesClassifier package:



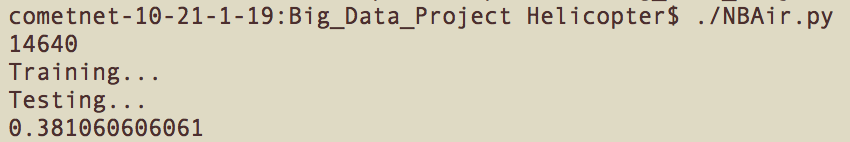
**Accuracy:**

Naïve Bayes:

If we only need to predict three classes(Positive, Negative, and Neutral), it could have the accuracy of 0.794696969697.

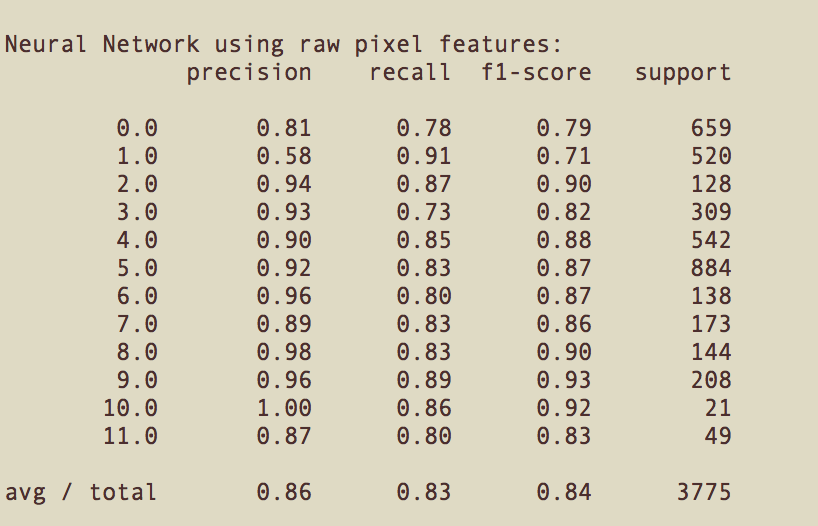


If we only need to predict all twelve classes (including all classification of different reasons of negative), it could have the accuracy of 0.381060606061, which is relatively good because some of the different reasons are very alike.



Neural Network:

Neural network could have the accuracy of 86%, which is the best result we got among all the methods. It is because neural network is extremely well performance with multiple classes output.



This result is soooooo awesome!!

**Overfitting Avoiding:**

For neural network, the biggest problem is overfitting. To avoid overfitting problem, I use python scikit-learn package to compare the accuracy with Spark Multilayer perceptron classifier. The reason I can do this test is scikit-learn package is scikit-learn package has some inner mechanism to avoid overfitting(I used 100 hidden units and 1000 hidden units and same time of iterations—let’s say, 50 times—and the accuracy is the same)

Once the accuracy of Spark Multilayer perceptron classifier has the close result to scikit-learn package, I can assume that it has no overfitting in the model.

And Naïve Bayes do not have the problem of overfitting.