**Luke Walsh**

**May 11, 2016**

## PySpark Streaming and NLTK Text Processing:

## Naïve Bayes Classification for Twitter Sentiment Analysis

**Problem/Purpose:** Build a PySpark Streaming application using Python’s NLTK library that analyzes the sentiment of Tweets about a given topic/search term in near-real time.

**Big Data Set:**

Training data set:

-“Twitter Sentiment Analysis Dataset”

-~1.5 million Tweets already classified as positive or negative

-source: <http://thinknook.com/twitter-sentiment-analysis-training-corpus-dataset-2012-09-22/>

-format: csv

Classification data:

-Twitter live feed (streaming)

-resource URL: https://stream.twitter.com/1.1/statuses/filter.json

-returns specified number of Tweets for every specified time interval

-format: returns JSON

**Hardware:**

* CentOS 6.7 VM running on Windows 10 VMWare Workstation 11.0

**Software:**

|  |  |
| --- | --- |
| **Technology/tools** | **Description** |
| Hadoop 2.6.0-cdh5.7.0 | Distributed file system with 1 live data node (pseudo-distributed environment) |
| Spark version 1.6.0 | Cluster computing software; this app will use PySpark Streaming to analyze streaming data |
| Python version 3.3.3 | Language used to develop the app |
| Python NLTK library | Natural Language Toolkit Library – a Python package that has useful tools for natural language processing such as the Naïve Bayes classifier this app uses |
| D3.js | JavaScript library used for visualizing results |

**YouTube URLs here:**

* 2 min: <https://youtu.be/_DY7L50deqo>
* 15 min: <https://youtu.be/ARP5q4LiMSc>

**Overview of steps:**

1. Train a model (Naïve Bayes) that will classify Tweets as positive or negative

2. Set up Twitter stream and format incoming tweets to run through the model

3. Run each batch of Tweets through the classifier and capture results

4. Visualize

**Summary:**

This application aims to classify the sentiment of Tweets (positive/negative) in near-real time (i.e. batch processing) using PySpark and Python’s Natural Language ToolKit library. The application connects to Twitter’s live data feed and collects batches of Tweets in a time interval specified by the user. A Naïve Bayes classification model (from Python’s NLTK package) is used on these batches of Tweets in the PySpark environment to classify the Tweets as either positive or negative. The total number of positive and negative Tweets for each batch is counted, and D3.js is used to visualize these proportions for each batch over time.

Below are a few brief notes on the pros/cons of using this technology.

Pros:

* Spark has very fast batch processing compared to something like Hadoop MapReduce due to in memory computations
* Spark is well suited for streaming applications and is highly scalable
* Python NLTK library has many built-in tools to assist in NLP, and the Naïve Bayes classification model which is often used in text classification is built into the NLTK package
* Python’s ‘requests’ library makes it easy to set up a REST client to connect to streaming sources like Twitter’s data feed
* Overall, PySpark Streaming and NLTK are very useful when used together

Cons:

* No major concerns raised by this demo project
* Some online research suggests that, since Spark is built on Java/Scala, there are some operations that cannot be performed by Python and need to be done in Java or Scala. In this project I was able to accomplish everything in Python though.
* Scala and Java may potentially perform faster than Python. May depend on the application.