Introduction

This project is to build a model that will determine the sentiment (positive and negative) of the text on Twitter data set. To do this, a model is built and tested on the existing data, and then the model is used to analyze the real-time data using Spark Streaming.

What is Sentiment Analysis

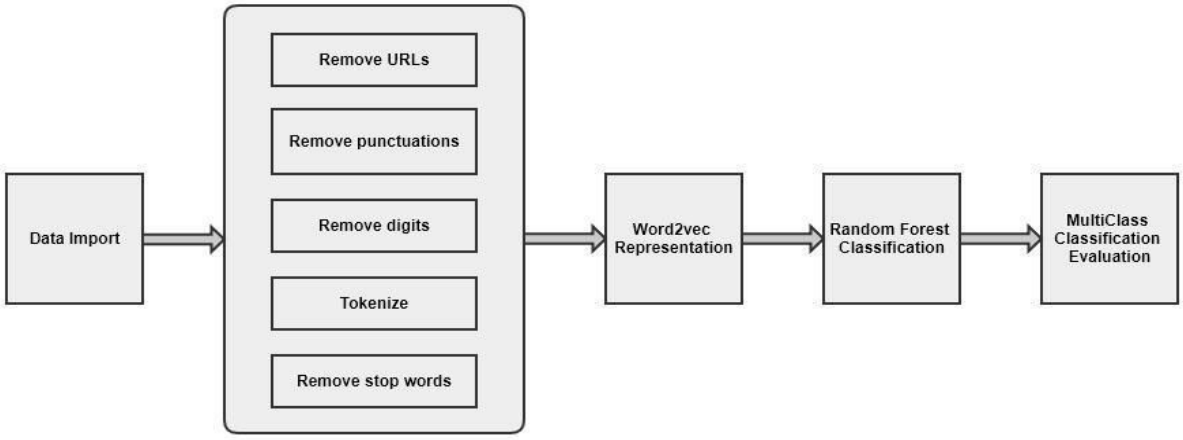
Sentiment Analysis is the method to know whether a piece of writing is positive or negative using Natural Language Processing. It is sometimes also known as Opinion Mining. It not only helps companies understand how they're doing with their customers, it also gives them a better representation of how they stand up against their competitors. For example, if your company has 20% negative sentiment, is that bad? It depends. If your competitors have a roughly 50% positive and 10% negative sentiment, while yours is 20% negative, that merits more discovery to understand the drivers of these views. Knowing the sentiments linked with competitors helps companies assess their own performance and look for ways to improve.

Architecture

There are two parts to the problem:

Training the classifier

Below is the process to build the classification model:



Step 1: Choosing the Data

To obtain training data for sentiment analysis, I downloaded the data from here.

Here are some sample tweets along with classified sentiments:

Sentiment	SentimentText
1	Feeling strangely fine. Now I'm gonna go listen to some Semisonic to celebrate
0	HUGE roll of thunder just now...SO scary!!!!
0	I just cut my beard off. It's only been growing for well over a year. I'm gonna start it over. @shaunamanu is happy in the meantime.

Step 2: Data Preprocessing

Before we start building the analyzer, we first need to remove noise and preprocess tweets by using the following steps:

1. Lower Case - Convert the tweets to lower case.
2. URLs - Eliminate all the URLs via regular expression matching.
3. Punctuations and additional white spaces - remove punctuation at the start and ending of the tweets, e.g: ' the day is beautiful! ' replaced with 'the day is beautiful'. Also, replaced multiple whitespaces with a single whitespace.
4. Digits: Removed digits.
5. Tokenization: Process of converting a sequence of characters into a sequence of tokens.
6. Remove Stop words: Stop words are usually refers to the most common words in a language. Example: “a”, “the”, “is” etc. I removed those stop words for data cleaning.

Below is the sample output of pre-processed data:

Tweets	Sentiment
["sad","apl","friend"]	0
["missed","new","moon","trailer"]	0
["omg","already","o"]	1
["omgaga","im","sooo","im","gunna","cry","ive","dentist","since","suposed","get","crown","put"]	0

Step 3: Vector Representation

Machine learning classification algorithm can only work with the numeric vectors in Spark. In this project, Word2vec method is used to transform the text to vector representation. Word2vec takes as its input a large corpus of text and produces a vector space. In this model each tweet is represented as single vector and each word is a feature. Word vectors are positioned in the vector space such that words that share common contexts in the corpus are in close proximity to one another in the space. Below is the sample output of the word2vec:

label	features
0	[1,10,[],[0.19162334998448688,0.10304862912744284,0.0014406442642211914,0.07609715064366658,0.11180469642082849,0.21096690
0	[1,10,[],[-0.17390286549925804,-0.1760585061274469,0.014200386591255665,0.009363599121570587,-0.01914331503212452,0.2083114
1	[1,10,[],[0.12162707777072986,-0.038270335644483566,-0.0316042856623729,0.1373838298022747,0.056497131784756974,0.385359237
0	[1,10,[],[0.07565960868333395,-0.03558567850492322,0.21174584343456307,0.07709618987372288,0.005194864594019377,0.12514978

Step 4: Classification Model

Machine learning has three types of learning algorithm

1. Supervised learning: This type learning algorithms consists of a target/outcome variable (dependent variable) which is to be predicted from a given set of predictors (independent variables). Names of some of the algorithms are Decision Tree, KNN, and Random Forest etc.
2. Unsupervised learning: This type of algorithm does not have any target or outcome variable to predict/estimate. Example, K-Means Clustering, K-Means++ Clustering and Hierarchical Clustering etc.
3. Reinforcement learning: In this of learning, machine is trained to make specific decision. The machine is exposed to an environment where it trains itself continually using trial and error. Machine learns from the experience and tries to capture the best possible knowledge to make accurate business decisions. Example of such algorithm is Marcov Decision Process.

This project tries to analyze or categories the tweets into positive and negative emotions, which means the type algorithm useful will be classification algorithm (supervised learning algorithms).

The Classification algorithm used in this project is Random forest. The whole dataset is divided into 70% training and 30% test data. Algorithm learns from the training dataset and applies to the test dataset to find the accuracy. To evaluate the model Multiclass Classification Evaluator is used. This evaluator check accuracy of the trained model by classifying the data is test dataset and checking the predicted and actual results. Below is the sample output:

label	prediction	features
1	1	[1,10,[],[-0.03058440429158509,0.05943934358656407,-0.13249878883943894,0.016152805462479592,-0.00791166312992
1	1	[1,10,[],[-0.11831726308446378,0.030592259019613266,-0.1497608891222626,0.09457852377090603,0.153702105395495
1	1	[1,10,[],[0.0028777051096161204,0.025002362672239542,-0.07073737525691588,0.0025700507685542107,0.08025051389
0	0	[1,10,[],[0.03209234229647196,0.1842633063833301,-0.06911044797072044,0.1713677980005741,0.204058907335051,0.0
1	1	[1,10,[],[-0.029241726733744144,0.05691546807065606,0.03823920805007219,0.020737318322062492,-0.0489882733672
1	1	[1,10,[], [0.014108429042001564,-0.026046982035040855,-0.02512898544470469,-0.01163675170391798,-0.02703643621255954,0.
1	1	[1,10,[],[0.0659870295120137,-0.0696711636015347,-0.16383278210248264,-0.16362061551106827,0.3958602824381419,

Showing the first 1000 rows.

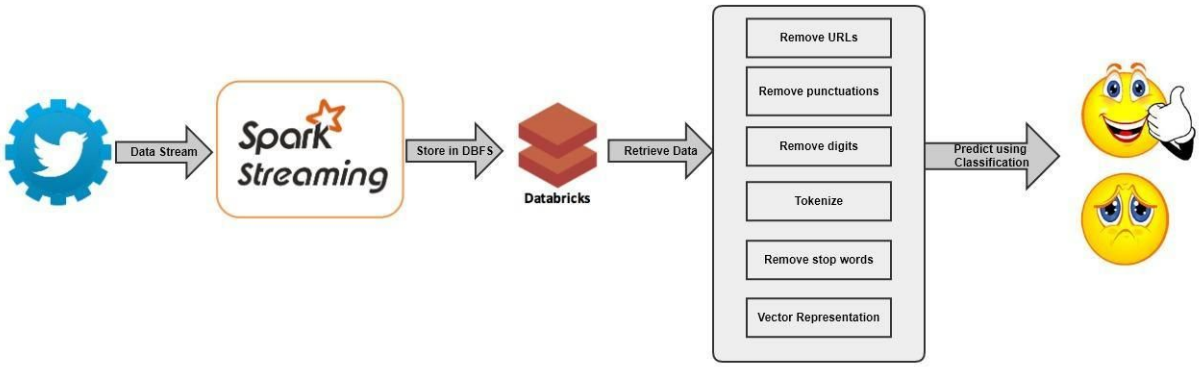
5. Results

As mentioned in step-4, the Multiclass Classification Evaluator is used to evaluate the model. The model is giving almost 65% accuracy for the test dataset.

```
Test Error = 0.351067
```

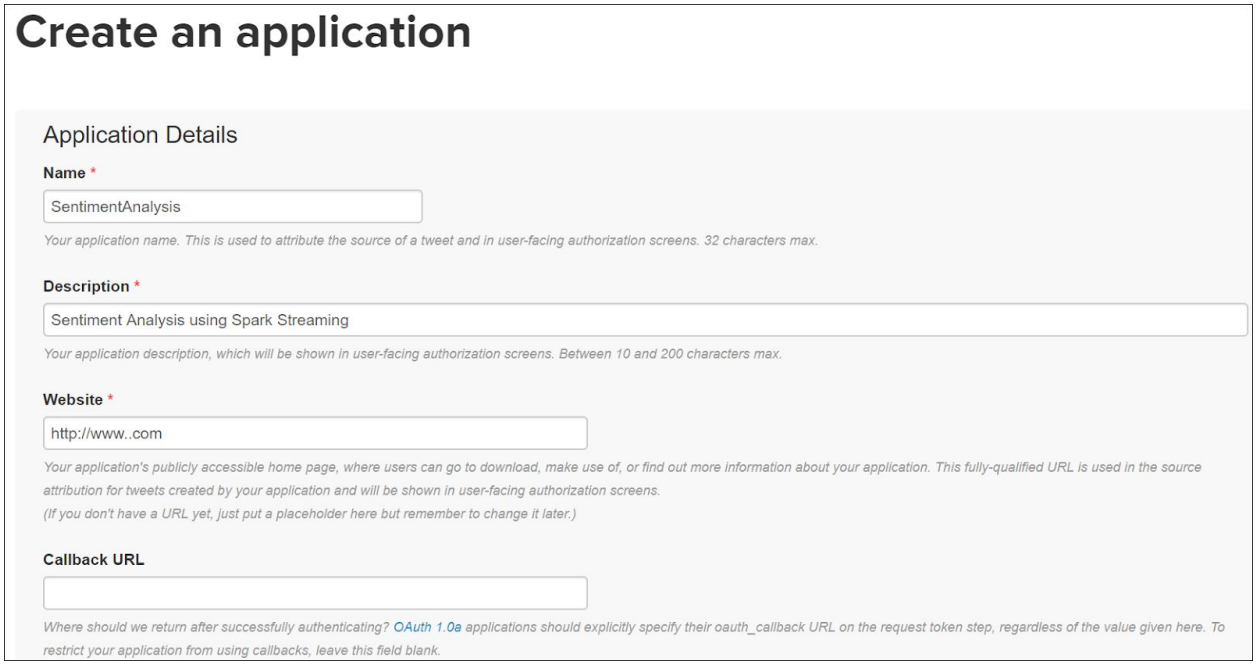
Steaming the real-time data

Here, I built a Spark Streaming application in Scala to fetch live Twitter content and use the previous Python Notebook to analyze the sentiments. Below is the streaming process architecture:

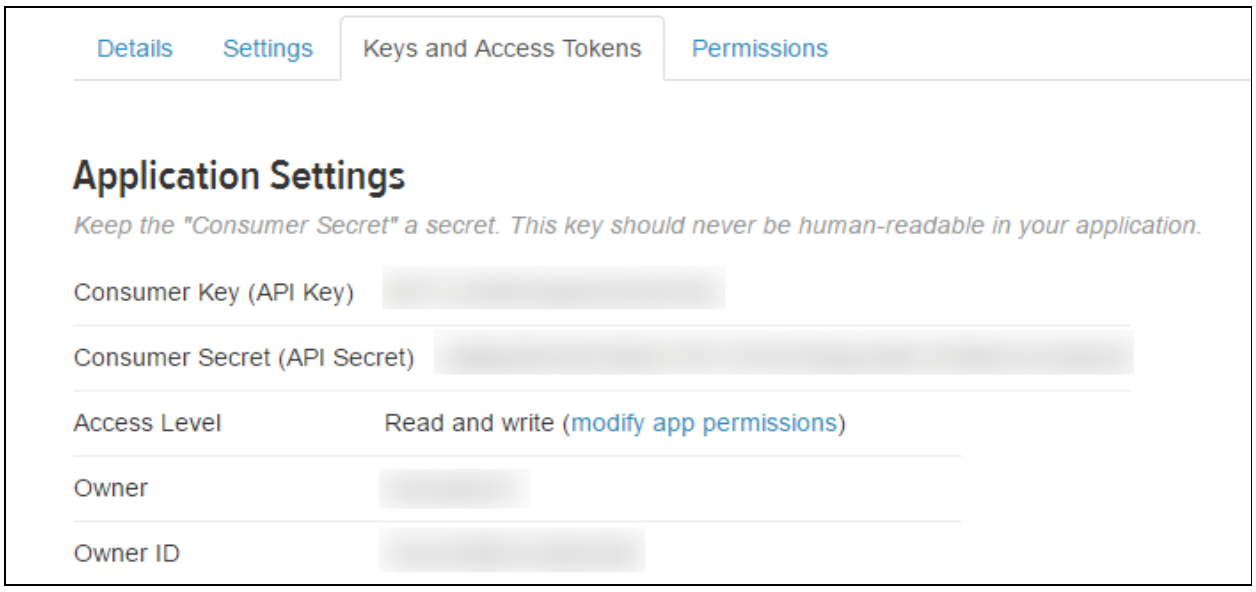


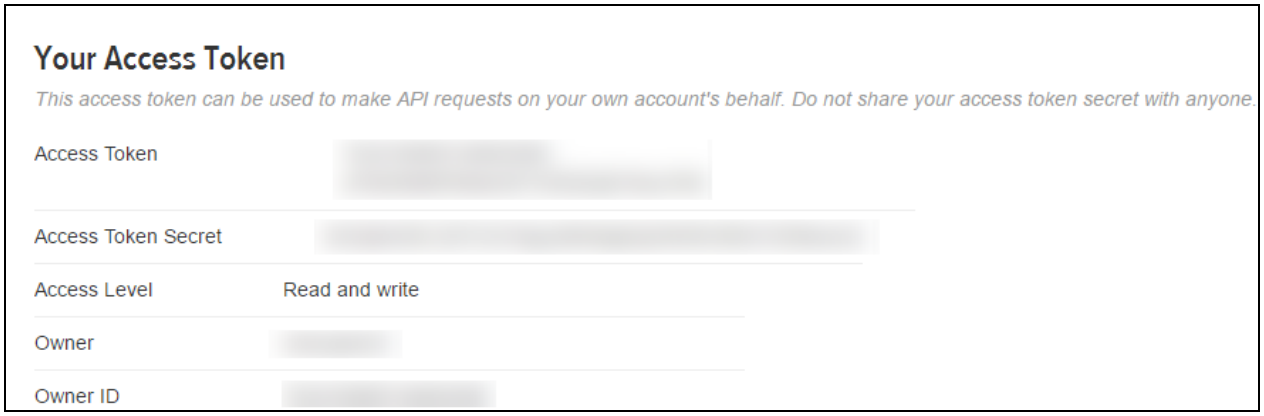
Step 1: Set-up the Twitter App

To have access to the Twitter API, you'll need to login the Twitter Developer website and create an application. Enter your desired Application Name, Description and your website address making sure to enter the full address including the http://.



After registering, create an access token and grab your application's Consumer Key, Consumer Secret, Access token and Access token secret from Keys and Access Tokens tab.





Step 2: Stream the Data Using Spark and Scala

In this project, live tweets from 'sgunjan05' account is downloaded using Spark stream with Scala. Spark Streaming receives live input data streams and divides the data into batches, which are then processed by the Spark engine to generate the final stream of results in batches. Here, batch size taken is for 30 seconds and total stream will run for 180 seconds.

Below, is the screen-shot:

```
-------------------------------------------
Time: 1524444690000 ms
-------------------------------------------
sgunjan05 - Bad attitude staff ignores customer satisfaction. I swear, I was very disappoint
ed at 11:30 this mornin… https://t.co/b2uGouAtBp

-------------------------------------------
Time: 1524444720000 ms
-------------------------------------------
sgunjan05 - Always great. Our life saver whilst abroad, good old xyz, great coffee, great co
okies and donuts, and free WiFi.

-------------------------------------------
Time: 1524444750000 ms
-------------------------------------------
sgunjan05 - Excellent and friendly service!! Every time I go to this xyz, the staff is frien
```

Step-3: Saving the Data and Preprocessing

Once we retrieve the data, tweets are saved in a text file in Databricks DBFS. After that, preprocessing and vector representation methods (Step 2 & 3 from Training the Classifier section) were applied to the streamed data.

Step-4: Prediction

The streamed data is then given to the classification model to sentiment prediction. The results show that out if four, three are predicted as the true result and one is predicted as false positive sentiment.

Tweets	prediction
["bad","attitude","staff","ignores","customer","satisfaction","swear","disappointed","mornin"]	1
["always","great","life","saver","whilst","abroad","good","old","xyz","great","coffee","great","cookies","donuts","free","wifi"]	1
["excellent","friendly","service","every","time","go","xyz","staff","friendly","shop","ver"]	1
["good","value","great","service","always","excellent","standard","another","canadian","great"]	1

Big Data Tools and Technologies

1. **Databricks** – It is a cloud-based big data processing using Spark.
2. **Spark** - Apache Spark is an open-source cluster-computing framework.
3. **Spark Streaming -** Spark Streaming is an extension of the core SparkAPI that enables scalable, high-throughput, fault-tolerant stream processing of live data streams.
4. **Scala -** Scala is a general-purpose programming language providing support for functional programming.
5. **PySpark** - PySpark is the Python spark package.
6. **Spark MLlib** - Spark's machine learning (ML) library.
7. **External Libraries** - requests-oauthlib, streaming-twitter-assembly-1.6 and Streamtwitter.

Code

The code can be found in below links:

1. Train the Classifier: https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/7862332491167843/985754373168602/4348179996931091/latest.html
2. Streaming the real-time data: https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/7862332491167843/3700315452641205/4348179996931091/latest.html

Future Work

Following is the problem of dataset:

1. Tweets are highly unstructured and non- grammatical.
2. Out of vocabulary words.
3. Extensive usage of acronyms like asap, lol etc.
4. Use of emojis like ☺ ☹ •

So, in future data can engineered in such a way that these anomalies can be eliminated.

Conclusion

Sentiment analysis can be applied to many topics. It was interesting to see how different tools make it easy to implement the algorithm. On the other hand, cleaning text and informal data makes it difficult for algorithm to predict with high accuracy. Hence, this project could be extended by applying better data cleaning techniques. Additionally, data pipeline could be experimented with different dataset and algorithms to obtain high accuracy.