Predictive Analysis of #BBNaija Tweets

**INTRODUCTION**

Python programming language has become a mainstream language and its applications cuts across several industries including, Information Technology, Finance, Government, Market research and many others. In this research work, we would examine the application of Python in discovering information from text documents.

**MOTIVATION**

Our motivation is gotten from a Reality TV show competition in Nigeria called Big Brother Nigeria. This show has gathered public attraction and has generated a lot of controversies among Nigerians. Perhaps one thing that makes this show popular is its eviction system, it requires the public to vote through Text message or an online voting platform. Each contestant with the lowest vote during a voting period would be evicted. The interesting thing is that most Nigerians now take to twitter to share their opinions and perception about their favorite contestant. Thus, tons of thousands of tweets are added daily mostly from Nigerians about this reality show. We want to extract this data from twitter, analyze them and check for interesting information about them.

**DATA SET AND DATA COLLECTION:**

We took advantage of the fact that the show is currently ongoing, as a result we were able to gather tweets about the show using the hashtag **#bbnaija**. We connected to twitter using twitter streaming API and we were able to gather **580,000 tweets.**

The good thing is that twitter has a feature that allows developers to download streams of tweets as they are being posted into a database. This features granted us access to twitter using twitter streaming API to collect the tweets.

Data Preprocessing:

Data extracted from Twitter usually come raw and unprocessed. For a data to be useful for analysis it needs to undergo the data cleaning. **Data preprocessing** is a **data** mining technique that involves transforming raw **data** into an understandable format. Real-world **data** is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors1. Data cleaning is an essential aspect of text analytics. A typical raw twitter tweet usually contains Emoticons, Irregular use of upper- and lower-case letters in a sentence, html website links, stop words, lots of punctuations. Etc. Some packages in python such has Lambda functions, Pandas and NumPy were used in the data cleaning process.

Below is a snapshot of an uncleaned data.

Here is a snapshot of the same data which is now cleaned and useful for analysis.

**Our Data Cleaning** Process was in this Stages:

1. Transformed tweets text to lower – here we transformed all the tweet text to lower case to preserve uniformity in the data.
2. Removed emoticons – all emoticons and smileys objects were removed at this stage
3. Removed Stop words – In computing, **stop words** are words which are filtered out before or after processing of natural language data (text)2
4. Removed Url Links

We observed some contestants were pronounced in different ways, we try to conserve data by stemming this names as one. Therefore we:

* Replaced Patterns

In text analysis it is sometimes useful to break text into word tokens, this word tokens can be used to calculate frequencies and perform quantitative analysis. We splitted the whole tweet document into word tokens

**DATA PROCESSING:**

After extracting the raw file from twitter, we performed data cleaning on the specific parameters needed for our analysis like; tweets made within the last few weeks, the locations the tweets came from, dates each tweet was made and username of the tweeters. Hence, we wrote python codes to perform our analysis

**OBJECTIVES:**

We have been able to answer the following research questions at the end of the project:

1. How are tweeters distributed by location?
2. Who are the top contestant by popularity on twitter?

.

1. Number of tweets that contains each contestant?
2. Words with highest correlation with each contestant and generating N-grams?

.

We would generate n-grams using python to group words that have high correlation together.

1. Who are the Top tweeters and their location?
2. What are the sentiments around each contestant?

**Analysis and Methods:**

**Python Modules used**:

The following modules were used for analysis in python

* *matplotlib* and *seaborn* and *wordcloud pylab* for creating charts
* *pandas* and *numpy* for operating with Dataframes and Arrays
* *re* for Regular Expressions
* *textblob* and *nltk* for Sentiment analysis and Natural Language Processing

The implementation was done on Jupyter IDE which sits on the Anaconda environment.

**Reading in Data**

The extracted Tweets from twitter were stored in a Json file. Each tweet's Json file contains information about the tweet, such as creation date, tweet text, location, and many more even retweeters information are also stored in each tweet Json file. We read in the entire Json file and we extracted three information that was needed to answer our research questions. tweet text, tweet location and tweet date

**Creating a Dataframe**

In order to utilize the power of the **pandas data frame** structure, we created a Data Frame of Tweet text (which is the most important component of our analysis), tweet location and tweet date

**User defined Functions**

We defined some python Functions that was useful for our analyis throughout our analysis.

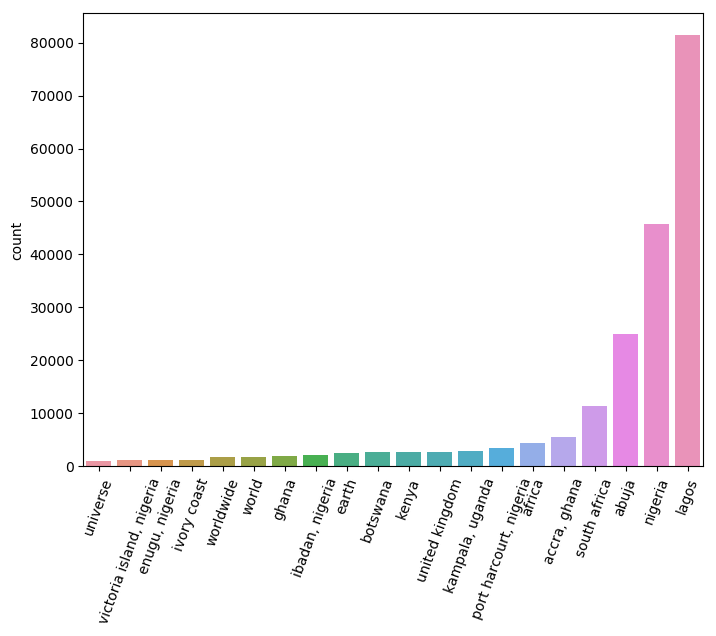
1. **emoji\_pattern** – This function creates a regular expression pattern for emojis to be later removed during the cleaning process
2. **rePart**: This function creates a dictionary of some regular expression pattern that would be used in replacing some words during the cleaning process
3. **reExpr**: This function creates a pattern which would be used in counting word occurrences as part of the *CountWrd* function
4. **RemoveStopWord**: This function receives a string, sentence or a tweet, splits it and returns a new string with no stop word. Stop words are prepositions and conjunctions that have little or no usefulness in Natural Language Processing or Text Mining.
5. **RemoveURL**: This function Removes URL from tweets and returns a string with no URL
6. **ReplacePattern**: Replace similar patterns of words with one single word. Somewhat related to stemming.
7. **CountWrd**: Receives two arguments. A data; could be a list, array or series, and a search pattern. counts the occurrences of that pattern in the data. Then returns the count.
8. **Ngram**: Receives two arguments. A data; could be a list, array or series, and a key word. Creates and returns a list of bi-gram of words around the keyword.
9. **GetTweetSentiment**: Receives a string, here a tweet. the module **textBlob** is used to get sentiment polarity. if polarity is less that 0; it returns a negative sentiment, if equal to 0: it returns a neutral sentiment, if greater than zero: it returns a positive sentiment.
10. **PlotWordCloud**: This function receives a list, concatenates it to a text. Then returns a word cloud plot of the Text

**Data Visualizations**

**RESULT 1: How are tweeters distributed by location?**

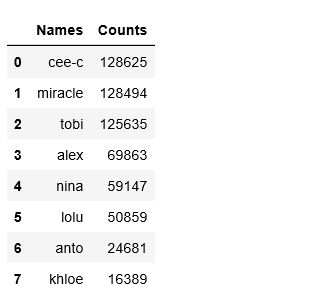
This graph tells us that Lagos has highest contribution on BBnaija based on the top 20 countries we output. Also, we can tell that other African countries like kenya, uganda, ghana, and ivory coast have good online engagement for BBnaija. One of the observations gotten from the data is that most twitter users leave their locations off. We got a count for such users and categorized them as “NONE”, but we had to filter out that from the graph since it carries no useful information.

**Display of BBnaija Engagement**

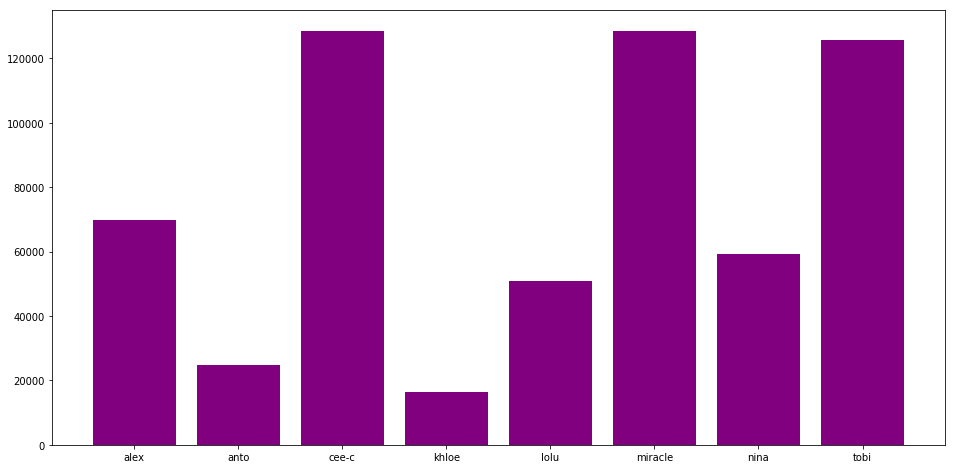


**RESULT 2: Who are the top contestant by popularity on twitter?**

We can tell from this graph that contestants mostly talked about are Cee-c, followed by Miracle and then Tobi.

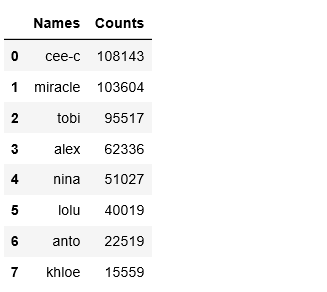


**Graphical Display of Top Contestants by Popularity**

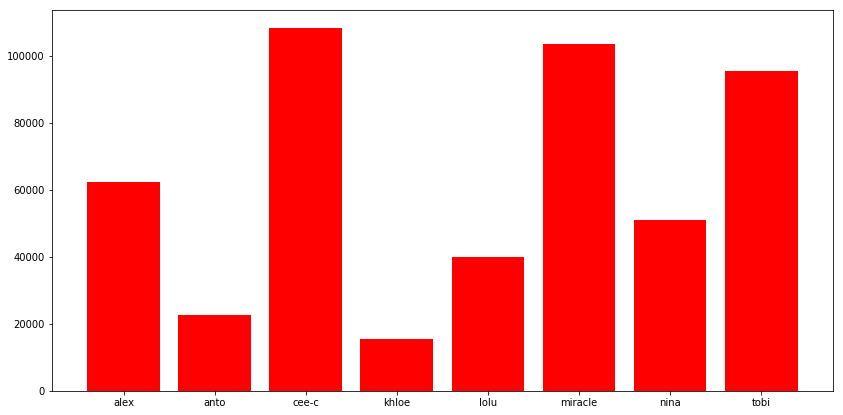


**RESULT 3: Number of tweets that contains each contestant?**

We can tell from this graph that for every tweet made by twitter users, Cee-ce appears most, followed by Mira and then Tobi. The difference between this result and the previous one is that we counted each tweet that contains each contestant regardless the number of times the contestant was mentioned per tweet.



**Graphical Display of Tweets that Contains each Contestants**



**RESULT 4: Words with highest correlation with each contestant and generating bi-grams?**

In text mining, an n-gram is a phrase or combination of words that may take on meaning that is different from, or greater than the meaning of each word individually.

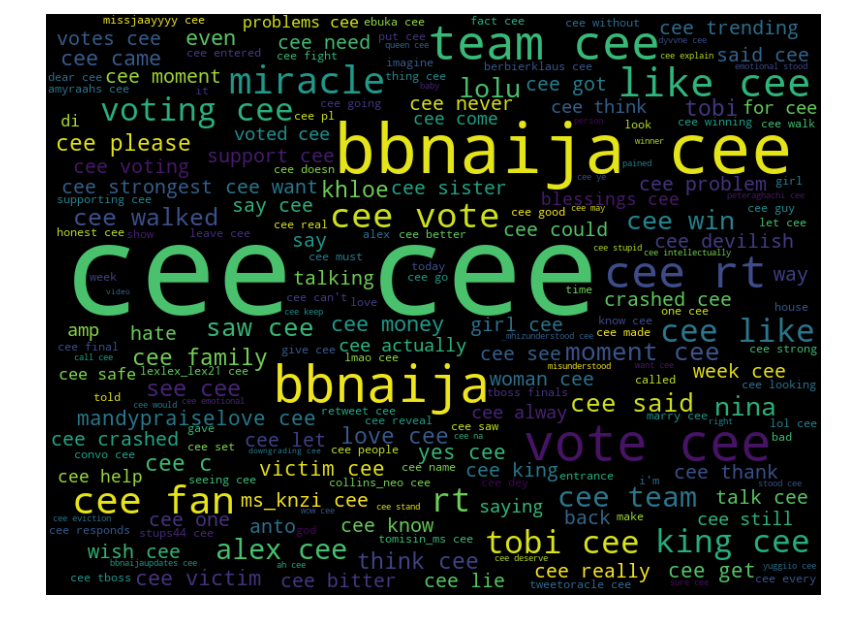
Here we created a bigram on words around each contestant as posted by twitter users to see words that are associated with them the most.

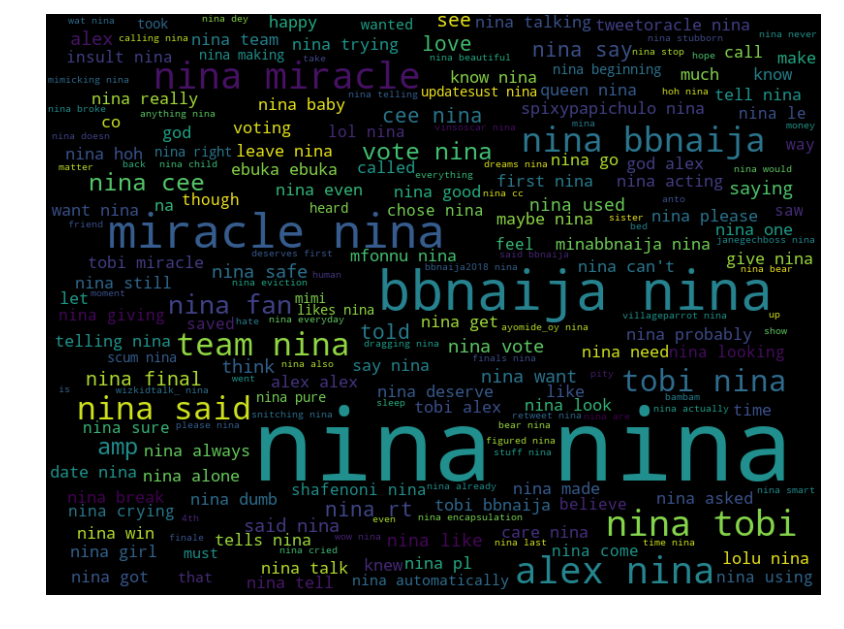
So basically, an n-gram can give us an attribute of a person. For example, words around miracle reveals he is a **people’s person**. As you can see on the Word Cloud; In text mining, a word cloud is a very powerful tool for data visualization. We have created a commonality word cloud. Examples of words that around him that show he seems to be a **people person** are ‘support miracle’, ‘voting miracle’, ‘like miracle’, ‘miracle fan’ and so on.

Words around Cee-c reveals she might be an active person. As you can see on the word cloud; ‘cee-c talked’, ‘cee-c can’t’, ‘cee-c fight’, ‘cee-c crashed’ and so on.

Words around Nina reveals that she might be a **Talking type of person**. As can be seen on the word cloud; ‘telling nina’, ‘nina said’, ‘nina talk’ e.t.c



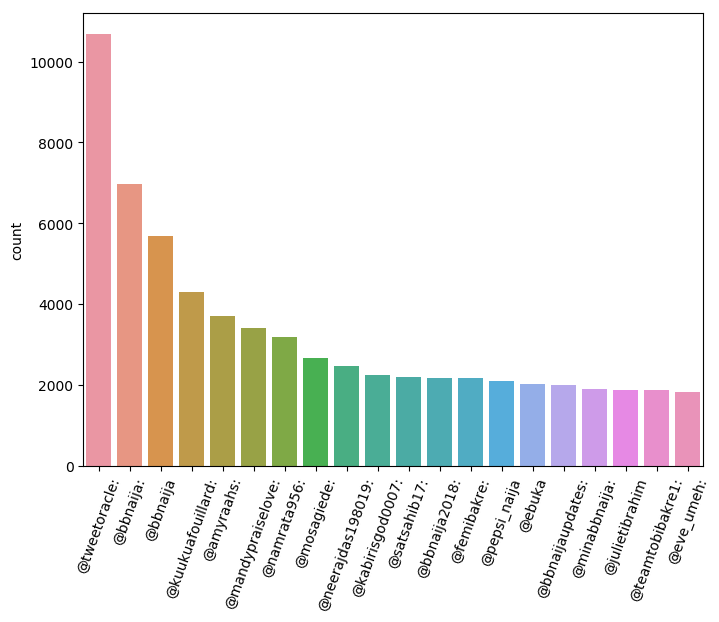




**RESULT 5: Who are the Top tweeters and their location?**

We can infer from our research analysis that @tweetoracle is the highest poster as regards BBnaija followed by @bbnaija:, @bbnaija, @kuukuafouillard and so on. Some said @tweetoracle is the most influential social media personality, well maybe that is true!

**Graphical Representation of Top Twitter**



**RESULT 6: What are the sentiments around each contestant?**

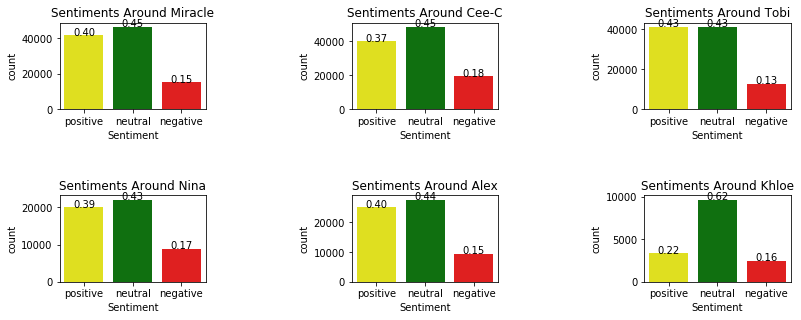
Sentiment Analysis is the process of computationally identifying and categorizing opinions expressed in a piece of text, especially to determine whether the writer's attitude towards a particular topic, product, etc., is positive, negative, or neutral.

In this research we computationally categories every tweet around a contestant into either of positive, negative or neutral sentiments. This helps us determine further those contestants who have little or more love from fans.

Tobi seems to be the one with the highest positive, whereas Cee-c even with a high positive has the highest Negative. However, Miracle has a high positive, low negative and has more tweets compared to Tobi.

As statisticians who do not exactly conclude on a hypothesis, but we can infer and make predictions. We feel Miracle cee-c and Tobi have the top shots in the game

**Graphical representation of the sentiment Analysis on each contestant**



This research work was carried out by:

Makanjuola Ogunleye Miracle Dickson

Graduate Student, Graduate Student,

Predictive Analytics. Data Management and Analysis

APSU, USA. APSU, USA.

[ogunleyemakanjuola@gmail.com](mailto:ogunleyemakanjuola@gmail.com) [dicksonmira@gmail.com](mailto:dicksonmira@gmail.com)

References

1. What Steps should one take while doing Data Preprocessing?. Hacker Noon. https://hackernoon.com/what-steps-should-one-take-while-doing-data-preprocessing-502c993e1caa. Published 2019. Accessed January 10, 2019.
2. Rajaraman, A.; Ullman, J. D. (2011). "Data Mining". Mining of Massive Datasets. pp. 1–17. doi:10.1017/CBO9781139058452.002. ISBN 9781139058452.