**CCT College Dublin**

**Assessment Cover Page**

*To be provided separately as a word doc for students to include with every submission*

|  |  |
| --- | --- |
| **Module Title:** | Big Data Storage & Processing  Advanced Data Analytics |
| **Assessment Title:** | Twitter sentiment analysis |
| **Lecturer Name:** | Muhammad Iqbal  David McQuaid |
| **Student Full Name:** | Sinéad Duffy |
| **Student Number:** | SBA22229 |
| **Assessment Due Date:** | 26 May 2023 |
| **Date of Submission:** | 26 May 2023 |

**Declaration**

|  |
| --- |
| By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution. |

# Group ID - MSc in Data Analytics

Author: Sinéad Duffy

e-mail: [sba22229@student.cct.ie](mailto:sba22229@student.cct.ie)

Student ID: sba22229

## Abstract

*This report outlines the process of acquiring one year of data from twitter, how it was inputted to a distributed file system, namely Hadoop. A comparison of MongoDB and MySQL was carried out using YCSB. Processing of the data set was carried out using Pyspark, and the dataset was filtered to only display tweets relating to the ‘weather’. Sentiment analysis was applied using TextBlob whilst time series analysis used the ARIMA process. The results and output of both methods are discussed before a conclusion is reached on the overall process of the analysis.*

Link to Github - <https://github.com/sinduff-cct/CA2-2022-Twitter>

# Introduction

* 1. This report completes the requirement for the second assignment completed as part of the MSc in Data Analytics. The requirement is to download one year of tweets from Twitter, to insert into a distributed file system, prepare the data before completing some analysis using sentiment and time series algorithms. Once analysis is complete, the author will review the results and outline any learnings and conclusions that they may have uncovered during the analysis.
  2. The chosen topic area for this project is the weather. As part of the data processing requirement, the dataset will be filtered to only show tweets relating to this topic. The weather was selected as a topic due to human interest in it. From experience, the weather is discussed multiple times per day, with different people. The author felt that using the weather as the basis for analysis would give a useable amount of data by which to complete the analysis.
  3. The following section outlines the steps that were taken as part of the data storage and processing requirement.

# Data Storage and Processing

* 1. Distributed file systems organise data across servers or nodes within a network. The nodes work together like a network to ensure that it is possible for the data to be accessible, even if one of the storage notes are not available (B, 2023). The best know distributed file system is Hadoop from Apache, and it is more commonly known as HDFS (Hadoop Distributed Files System) (Bhutanadhu, 2023). For the purposes of this assignment, the author has utilised the Hadoop environment loaded into an Oracle based virtual machine, utilizing the Ubuntu operating system.
  2. **Data Acquisition**

The assignment required that a year’s worth of data from the social media site twitter. Alternatives were offered where historical data was stored in the archive.org site. The author attempted to create a link to the Twitter API but recent changes to the site’s settings would not facilitate the level of data requested. There was a facility to requested access based on an academic need, however no response to the author’s request was received, even using the student’s email address.

The author also made several attempts to download a twitter dataset from the archive.org site using a simple download of individual files, using torrents, a bulk download using a command-line tool row get command. However due to limitations on the available the Wi-Fi connection as well as laptop processing time, an alternative solution needed to be found.

The author did a comprehensive web search and identified twitter datasets available such as Data.World and Kaggle which stored the required dataset. However, these datasets were fully cleaned and processed and deemed not suitable. Sentiment140 is a dataset that was created for academic research and consists of more than 1.6 million individual tweets gleaned over one year (*For Academics - Sentiment140 - A Twitter Sentiment Analysis Tool*, no date). The author felt that this dataset would fulfil the criteria for the assignment and proceeded to use for the remaining sections.

* 1. **Preparation and Processing**

The Sentiment140 dataset was downloaded and inputted to the HDFS on the Ubuntu virtual machine (see appendix one).

Pre-processing was carried out using Apache Pyspark. The full dataset was loaded into a Jupyter notebook and processed by creating a structType to hold the data, with the relevant data types. A function was created to remove ‘stop words’, punctuation, and other marks (hashtags) to allow the data to be processed more efficiently. Finally, all the text was put in the same lower-case format. This data was placed in a new column to differentiate from the original data.

The data for weather related tweets was extracted, and the resulting new dataframe was saved to the local drive (see appendix two).

The author also carried out a sentiment analysis on the data, however the virtual machine started having problems processing the data at this point. The author had an issue with the screen of their laptop, which impacted the performance of other elements of the computer. The author is travelling and must wait to return to Ireland to have the machine fixed as it is under warranty. The rest of the processing was carried out using a windows-based machine.

* 1. **Rationale and Justification**

Pyspark was selected to carry out the pre-processing on the HDFS as it was the package the author was most familiar with, having successfully used the package in a previous assignment.

In respect of the other database issues with the virtual machine began to surface as the project went on. Several of the SQL / NoSQL databases became corrupt and would not load (Cassandra / SparkSQL), if the package did load it would inevitably slow down the virtual machine sending it into a processing loop, resulting in a hard reboot being necessary.

Once processing in the virtual machine became too time consuming, the author moved the analysis to a window’s machine. The saved dataset was moved to windows to allow processing to continue.

* 1. **Comparative analysis using YCSB**

YCSB - *Yahoo! Cloud Serving Benchmark*, was proposed by Cooper et al (2010) as a method to evaluate the performance of cloud bases data systems. The benchmark uses different measurement such as latency, throughput etc as a method of comparing one database against another - regardless of the type of database is in question, as it can be used on hdfs as well as cloud based systems (Cooper *et al.*, 2010).

For analysis, the author compared MongoDB against MySQL as they are a non-SQL and an SQL database respectively. The resultant output file is stored in the submitted folder, and screenshots of the process are included at appendix 3 and 4 respectively.

# Sentiment Analysis

* 1. Sentiment Analysis was initially conducted within Pyspark, but as outlined previously, it was relocated to the windows-based machine for completion. Sentiment analysis allows an analyst to gather insights and / or context on text data such as twitter (Shah, 2020). TextBlob was selected as the library to complete the analysis and uses the Natural Language ToolKit (NLTK). The model returns an expression on the polarity and subjectivity of a piece of text. In this analysis, a sentiment score was assigned to each tweet, this was then generalised into positive / negative or neutral sentiment. In practical terms, Twitter has provided a method of monitoring public opinion in real time (Weller, 2013).
  2. TextBlob was selected for use over Vader as it provided a simple implementation, as well as being a well-tested analysis method. TextBlob allowed for initial interpretation of the data that showed the dataset as being largely positive in nature, indicating that the tweets gathered about the weather are largely positive. Further analysis on weather data from relevant authorities such as Met Eireann could be carried out to determine the specific types of weather during the period.

# Time Series Analysis

* 1. Another method of gaining insight from data such as twitter is implementation of a time series analysis. Such analysis can be helpful in identifying if any trends are showing in the data, it can help with forecasting and predicting outcomes where the data is suitable for such analysis. It can also help identify outliers in the data that may not be apparent using other analysis methods (*A Guide to Time Series Analysis in Python | Built In*, no date). Data can be processed using hourly / daily / weekly / monthly or annual data (*Python Data Analysis : Perform Data Collection, Data Processing, Wrangling, Visualization, and Model Building Using Python*, no date, pp237).
  2. Reviewing the time series data as it stands, the author has determined that it is stationary in nature, i.e. that it has not constants trends or seasonality in it (*A Guide to Time Series Analysis in Python | Built In*, no date). Based on this review, the author has opted to apply the ARIMA forecasting methodology to the data. ARMIA is statistical analysis method that incorporates autoregression and moving averages (*Python Data Analysis : Perform Data Collection, Data Processing, Wrangling, Visualization, and Model Building Using Python*, no date). In simple terms ARMIA allows an analyst to forecast or predict an event based on past values in the dataset (*ARIMA Model - Complete Guide to Time Series Forecasting in Python | ML+*, no date).
  3. The time series analysis is displayed in three graphs using three grouping methods, namely one week, one month and three-month blocks. The graphs largely predict the same outcome, in that the prediction is positive. The variations in the graphs are interesting to see, and vary between being very negative it outlooks, to bordering on the low positive scale of the graph. The predictions based on one weekly data appears to be the most consistent. The ARIMA line rises sharply to positive and does not descend for the five weeks displayed. The positive outlook described in the graph could imply that the weather for this period was stable and ‘good’ i.e., no major variations were displayed in the tweets analysed. The prediction line did not ‘break’ above 0.336.
  4. Looking at the next time frame of one month’s predictions, the weather is more changeable, but still largely positive in nature. The ARIMA predictions are again positive, but there are bigger variations as the month progresses, with the scale reaching 1 and -1 respectively on different dates. The prediction line again hovers around the 0.3 line. There also appears to be periods of stable weather at the start and towards the end of the month respective, with periods of rapid changes in the middle and again at the end.
  5. The three-month forecast has more movement than previous forecasts and is particularly active from mid-May to early June. The stable periods seen in the one-month forecast are clearly visible again in early and late April. Early June seems to have a period of very different weather over a short period of time.

# **Dashboard**

* 1. The interactive Dashboard was created using Dash and Plotly Express. From research, it is common to combine both of these libraries when creating interactive / dynamic dashboards as they allow development of custom visualization, they can be easily integrated with other libraries in python such as pandas and NumPy. The combination of Dash, Plotly Express with the standard python libraries allow development of exciting visualisations and dashboards.

# Results and Discussion

* 1. As outline in earlier sections, exploratory data analysis weas completed to allow the chose dataset be filtered and inputted to the Jupyter environment for analysis. Using ‘feature engineering’, it was possible to apply TextBlob to the filtered tweets in a universal way, having eliminated stop words, punctuation etc. Once sentiment analysis was applied, it is possible to see that the feedback was largely positive in relation to the chosen topic area of weather.
  2. This analysis was borne out by the time series analysis completed in the following section. The forecast that was graphed of the three-analysis confirmed that the weather was due to be largely positive for the three time periods examined.
  3. If the author had more time available for analysis, she would have gone back to the exploratory data analysis stage and reinserted the time column to extract the general location of the tweet. This would have allowed more analysis on the geographical sentiment - was the sentiment localised to a particular area or continent, and was the sentiment positive, negative, or neutral.

# Conclusions

* 1. The use of hdfs as a file system to work with large dataset was largely a positive one. There was an issue with the virtual machine which the author was unable to resolve in time to submit this assignment. An alternative method of processing the data was secured with efforts made to ensure the same dataset was utilised throughout.
  2. Overall, it is possible to say that some level of detail was extracted from the twitter dataset. There are improvements that could be made to make the analysis sharper in terms of geographical location, timing etc. The sentiment analysis carried out using TextBlob was positive, which was confirmed by the time series data analysis.

# Reference

*A Guide to Time Series Analysis in Python | Built In* (no date). Available at: https://builtin.com/data-science/time-series-python (Accessed: 26 May 2023).

*ARIMA Model - Complete Guide to Time Series Forecasting in Python | ML+* (no date). Available at: https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/ (Accessed: 26 May 2023).

B, A. (2023) *Difference between distributed file storage and object-based file storage systems.*, *Medium*. Available at: https://medium.com/@bose8747/difference-between-distributed-file-storage-and-object-based-file-storage-systems-a7e30e2523fc (Accessed: 26 May 2023).

Bhutanadhu, H. (2023) ‘A Dive into the Basics of Big Data Storage with HDFS’, *Analytics Vidhya*, 6 February. Available at: https://www.analyticsvidhya.com/blog/2023/02/a-dive-into-the-basics-of-big-data-storage-with-hdfs/ (Accessed: 26 May 2023).

Cooper, B.F. *et al.* (2010) ‘Benchmarking cloud serving systems with YCSB’, in *Proceedings of the 1st ACM symposium on Cloud computing*. New York, NY, USA: Association for Computing Machinery (SoCC ’10), pp. 143–154. Available at: https://doi.org/10.1145/1807128.1807152.

*For Academics - Sentiment140 - A Twitter Sentiment Analysis Tool* (no date). Available at: http://help.sentiment140.com/for-students (Accessed: 26 May 2023).

*Python Data Analysis : Perform Data Collection, Data Processing, Wrangling, Visualization, and Model Building Using Python* (no date). Available at: https://eds.p.ebscohost.com/eds/ebookviewer/ebook/ZTI1MHh3d19fMjcyNTk5Ml9fQU41?sid=5e9f0a52-717e-4c5e-839f-46e0195c57ae@redis&vid=5&format=EB&rid=9 (Accessed: 26 May 2023).

Shah, P. (2020) *My Absolute Go-To for Sentiment Analysis — TextBlob.*, *Medium*. Available at: https://towardsdatascience.com/my-absolute-go-to-for-sentiment-analysis-textblob-3ac3a11d524 (Accessed: 26 May 2023).

Weller, K. (ed.) (2013) *Twitter and society*. New York: Peter Lang (Digital formations, vol. 89).

# Appendix

## Appendix 1

Save file **twitter\_DS\_1yr.csv** into Hadoop Distributed File System

A screenshot of a computer

Description automatically generated

## Appendix 2

Save processed file **weather\_tweets.csv** into HDFS

A screenshot of a computer

Description automatically generated

## Appendix 3

A screenshot of a computer screen

Description automatically generated with medium confidence

YCSB - Result

A screenshot of a computer screen

Description automatically generated with medium confidence

## Appendix 4

## Appendix 5

## Appendix 6

## Appendix 7

## Appendix 8

## Appendix 9

## Appendix 10

## Appendix 11