**Title to go here.**

**Abstract**

1. **Introduction**

The emergence of the internet can be traced back to the concept of a global computer network traced back to the 1960s. A paper published by Paul Baran, ‘On Distributed Communications’ is thought to be one of the foundational papers providing early inspiration for the design of the APRANET. The paper outlines the idea of a distributed communications network that could withstand partial outages. MIT researchers, including Lawrence Roberts, drew inspiration from Baran’s paper and proposed the initial design of the APRANET. In late 1969, the first APRANET message was sent between two nodes, which acted as a precursor to the modern internet. The internet, or the world-wide web (WWW) as it is sometimes coined, transformed from a communication between two nodes into a massive information exchange introduced by Tim Burners-Lee (T. Berners Lee, R. Cailliau, J. Groff, and B. Pollermann, “World wide web: the information universe,” Internet Res., vol. 20, no. 4, pp. 461–471, Aug. 2010.). The internet has now evolved into a significant medium for information sharing and communication. Communication systems via the internet can be traced back to developments such as Bulletin Board Systems (BBS) (1970s), UseNet (1979), and Internet Relay Chat (1988). These platforms primarily allowed for users to engage with each other via chat rooms, mail systems and the sharing of files. Significant developments emerged from 2000 onwards with regards to social media. Social media is defined as a ‘group of Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of user-generated content’ (Kaplan and Haenlein, 2010). The introduction of Friendster in 2002 proved to be a significant turning point in social media platform structure by giving users control over whom they connect with (Winder, 2007). Facebook and Twitter revolutionised social media websites and became two of the most popular and everlasting platforms since the inception of the internet. This paper places a particular focus on Twitter by analysing the sentiment of a cohort of users during a particular interval. Twitter is defined by Mistry (2011) as being a free social networking tool that allows people to share information, in a real-time news feed through posting brief comments about their experiences and thoughts (Britsol et al 2010). Users broadcast short posts known as tweets which are limited to 280 characters including spaces and punctuation, appearing in followers’ feeds. Users can weave their post into specific topics by adding a hashtag to a keyword in their tweet. Twitter can also be used by businesses, entrepreneurs, journalists etc. to increase brand awareness and create a social media presence that can help; interact with customers, monitor public trends and make public announcements. As of December 2022, Twitter’s audience accounted for over 368 million monthly active users worldwide (Statista, 2024). Due to the high volume of attraction to the social media platform it can be extremely useful to derive analytical insights from the information posted. One such way of extracting useful insight from Twitter is to analyse the sentiment of user posts via Tweets on a particular subject matter. This paper provides a framework for which sentiment can be extracted from Twitter data using advanced data analytics methods and big data storage and processing techniques, including forecasting sentiment for future time periods outside of the original time series.

1. **Methodology**

2.1 Overview

The goal of the research paper was to identify and analyse historic sentiment from a large dataset gleaned from the Twitter API (‘ProjectTweets.csv’) using advanced data analytics techniques. Various sentiment types were explored including polarity, subjectivity and the Valence Aware Dictionary and Sentiment Reasoner (VADER). Time series analysis was applied to the sentiments in Python using various libraries and eventual forecasts were estimated for the period in question.

2.2 Data Storage and Processing

*2.2.1 Hadoop Database (HBase)*

The dataset contained 1,600,000 observations, each representing a tweet posted by a specific user at a particular time. Due to the size of the data and the task to be performed a distributed data processing environment was required to store and process the data. The data was stored in Hadoop Database (HBase). HBase is the Hadoop database. It is used when you need random, real-time, read/write access to your big data. HBase’s goal is the hosting of very large tables-billions of rows X millions of columns-atop clusters of commodity hardware (Apache Hbase). HBase is a NoSQL database built on top of Hadoop. NoSQL databases are databases by which data can be handled and extracted with ease. A NoSQL database is a type of database management system that allows for the storing and extracting of data in ways different to traditional RDBMS methods. HBase utilises Hadoop’s Distributed File System (HDFS) and MapReduce model and contrastingly stores data in columns as opposed to rows (Egan, 2017). HBase was chosen as a mean of storing data for this research paper because of its flexibility.

The ‘ProjectTweets.csv’ file was stored in the HDFS using the below command.

Once the file was correctly stored and uploaded in HDFS an empty table named **INSERT NAME HERE!!** was created in HBase using the below command.

This table was populated with the data from the ‘ProjectTweet.csv’ file using the below command. The below command had to be manipulated to create column names and maintain row numbers for each observation. Following the storage of ‘ProjectTweet.csv’ in HDFS and HBase the dataset was processed using Apache Spark. HDFS and Apache Spark are later utilised to process the text of the tweets for the creation of a Natural Language Processing machine in Python.

* + 1. *Apache Spark*

Apache Spark is a Java Virtual Machine (JVM) based on distributed data processing engine that scales, and it is fast compared to many other data processing frameworks. Spark, as it can be referred to is a framework for distributed computing which depends on Hadoop MapReduce algorithms. Unlike traditional MapReduce algorithms, Spark can store intermediary results using Memory Computing (Aaron N. Richter, Taghi M. Khoshgoftaar, Sara Landset, and Tawfiq Hasanin, “A Multi-Dimensional Comparison of Toolkits for Machine Learning with Big Data”, 2015 IEEE International Conference on Information Reuse and Integration (IRI), San Francisco CA, pp. 1-8, 2015). Spark provides APIs in Java, Scala, Python and R. For this research paper, the Python API is used, namely PySpark. PySpark is an API that allows for the creation of Spark applications using Python. PySpark was used to allow for parallel processing of data preparation and text processing problems to allow for accurate sentiment analysis.

Various user-defined functions (UDF) were created within PySpark to allow for sentiment analysis results with higher degree of accuracy. Notable deficiencies with the text in the ‘tweet’ column included usernames included in the tweet, url websites included in the tweet, special characters included in the tweet and unnecessary whitespace appearing. UDFs were applied on the tweet column using PySpark allowing for quick and efficient data preparation prior to sentiment analysis.

Following data preparation in PySpark, the final Spark table was exported to HDFS. Within HDFS the below command allowed for the Spark output file to be copy to the local drive on the Ubuntu Virtual Machine.

Due to the size of the file exported from HDFS to a local drive on the Ubuntu Virtual Machine, two separate csv files were created. These two files were then shared between the virtual machine and the host machine. Shared folders can be created between the Ubuntu Virtual Machine and the host machine to allow for the transfer of files between respective machines.

* 1. Text Processing and Exploratory Data Analysis

This section details the process of lemmatization, which is applied through Python functions on the tweet column of the dataset. Exploratory Data Analysis (EDA) is also performed to highlight initial observations of the key components of the data prior to sentiment analysis.

*2.3.1 Lemmatization*

Lemmatization is the process of assembling the inflected parts of a word such that they can be recognised as a single element, namely the word’s lemma or vocabulary form (Balakrishnan, Vimala & Ethel, Lloyd-Yemoh. (2014). Stemming and Lemmatization: A Comparison of Retrieval Performances. Lecture Notes on Software Engineering. 2. 262-267. 10.7763/LNSE.2014.V2.134..). Lemmatization follows a similar procedure to stemming however it maps a root word to a relevant word as opposed to a root stem. Lemmatization is utilised in this paper, as the lemma of a word is a real language text format and is good practice to apply to inputs prior to using Natural Language Processors (NLPs) such as sentiment analysis algorithms. The WordNetLemmatizer function is used from the NLTK module in Python to lemmatize text using WordNet’s built-in morphy function. This function returns the input word unchanged if it cannot be found in WordNet.