**Program Title: MSc in Data Analytics**

**Module Tiles: Advanced Data Analytics / Big Data Storage and Processing**

**Assignment Title: MSC\_DA\_BD\_ADAv5**

**Lecturers: David McQuaid / Muhammad Iqbal**

**Student Name: Edmundo Fernandes**

**Student Id: sbs23034**

**Github Link:**[**https://github.com/Young-Jedi79/Integrated\_CA2-MSc**](https://github.com/Young-Jedi79/Integrated_CA2-MSc)

**Abstract**

Social media platforms have gained widespread popularity as channels for individuals to express their thoughts and viewpoints on a wide range of subjects. Monitoring the evolution of public opinion over time has become a valuable tool, serving purposes such as predicting sentiment trends and identifying potential triggers for shifts in sentiment. This understanding of the progression of topics and sentiments empowers businesses and government entities to promptly address negative sentiment.

In this research, we delve into the realm of traditional time series analysis techniques and their applicability to the analysis of trends in topics and sentiments. We leverage data collected from Twitter during 2009.

Among all the parallel programming models, one that gains a lot of popularity is MapReduce.

A section of this paper is to survey apply the MapReduce framework in the context of its open-source implementation, Hadoop, to summarize and report the wide topic area at the infrastructure level. I aim to perform a systematic review based on the prevalent topics dealing with MapReduce in several areas: performance, job/task scheduling, load balancing, resource provisioning, fault tolerance in terms of availability and reliability. I will run my study by doing a quantitative and qualitative evaluation of the research publication ProjectTweets.csv.

I will use PySpark, which gives Python provision on the Apache Spark platform, Spark is the state-of-the-art in large-scale data computing systems nowadays, due to its good properties including generality, fault tolerance, high performance of in-memory data processing, and scalability. Spark adopts a flexible Resident Distributed Dataset (RDD) programming model with a set of provided transformation and action operators whose operating functions can be customized by users according to their applications.

**Introduction**

There are currently 1.3 billion active Twitter accounts, around 528.3 million monetizable monthly active users as of 2023. It is estimated that this number will reach 652.23 million by 2028, where they freely convey their sentiments, encompassing emotions ranging from joy and sorrow to anger and more. Sentiment analysis encompasses the art of deciphering these emotions, opinions, evaluations, attitudes, and insights, mirroring the way in which humans perceive the world. It further categorizes these emotions into positive or negative classes. In the modern landscape, industries demonstrate a keen interest in harnessing textual data for semantic analysis, extracting invaluable insights into public sentiments about their products and services. This in-depth understanding of sentiment proves to be a cornerstone in gauging customer satisfaction, enabling them to refine their offerings accordingly.

To effectively navigate the realm of text data, businesses avidly extract data from diverse social media platforms, and among the notable platforms are the likes of Google Plus, Facebook, and Twitter, which provide a canvas for expressing opinions, viewpoints, and sentiments regarding various topics and events. Twitter Launched in 2006 and has risen to become the epitome of microblogging platforms.

In a single hour of 2017, an astounding 2 million users collectively generated 8.3 million tweets. Twitter users adeptly share their thoughts, emotions, and messages on their profiles, aptly referred to as "tweets," with each tweet restricted to a concise 140-character limit. Twitter sentiment analysis draws upon the field of Natural Language Processing (NLP), leveraging techniques such as word tokenization and the removal of stopwords like "I," "me," "my," "our," "your," "is," "was," and more. The NLP framework also plays a pivotal role in data preprocessing, encompassing text cleansing and the elimination of special characters and punctuation marks. Sentiment analysis proves instrumental in unveiling prevailing emotional trends on specific topics through the lens of Twitter users' tweets.

The usefulness of social media platforms is attributable to their ability to highlight valuable insights on varying perceptions of issues that are happening in real-time. According to research an analysis of Twitter data, people’s emotions especially can be useful in many areas such as stock market, election, vote, elections, war, crime and many more.

Natural Language Processing is a set of theory suggesting computers approach for evaluating and modelling naturally occurring texts at one or more levels of linguistic analysis to achieve human like speech recognition for various activities and applications.

Natural Language Toolkit is a free open-source python package that includes several tools for programming and data classification. NLTK collects text processing libraries for classification, tokenization, stemming, tagging, parsing and semantic reasoning.

**Background and Context**

This study includes levels of sentiment analysis as awe as approaches to sentiment analysis.

The Project Tweets.csv dataset is a collection of tweets from 2009. The dataset contains 1599999 Tweets and 6 columns which are:

ids: unique id of the tweet

date: date of the tweet

flag: refers to the query. If no query exists, then it is NO QUERY.

user: It refers to the name of the user that tweeted.

text: refers to the name of the user that tweeted.

sentiment: polarity of the tweet.

ProjectTweets.csv dataset will be used for a variety of purposes, such as:

Analyzing the sentiment of the tweets to understand how people are feeling about the project.

Tracking the progress of the project over time by using time series analysis.

**Literature Review**

This section serves as an introductory roadmap, orienting the readers to the scope, objectives, and significance of the review while setting the context for the subsequent sections.

“The pen is mightier than the sword” proposes that free communication (particularly written language) is a more effective tool than direct violence. Sentiment analysis is a series of methods, techniques, and tools for detecting and extracting subjective information, such as opinion and attitudes, from language. Traditionally, sentiment analysis has been about opinion polarity, i.e., whether someone has a positive, neutral, or negative opinion towards something. The object of sentiment analysis has typically been a product or a service whose review has been made public on the Internet. This might explain why sentiment analysis and opinion mining are often used as synonyms, although, we think it is more accurate to view sentiments as emotionally loaded opinions.

Therefore, analysis of user-generated data is beneficial for monitoring public opinion and assisting in making decisions. Sentiment analysis, as one of the most popular applications of text-based analytics, can be used to mine people’s attitudes, emotions, appraisals, and opinions about issues, entities, topics, events, and products (Cambria et al. 2022a, b, c, d; Injadat et al. 2016; Jiang et al. 2017; Liang et al. 2022; Oueslati et al. 2020; Piryani et al. 2017). Sentiment analysis can help us interpret emotions in unstructured texts as positive, negative, or neutral, and even calculate how strong or weak the emotions are. Today, sentiment analysis is widely used in various fields, such as business, finance, politics, education, and services. This analytical technique has gained broad acceptance not only among researchers but also among governments, institutions, and companies (Khatua et al. 2020; Liu et al. 2012; Sánchez-Rada and Iglesias 2019; Wang et al. 2020b).

Rambocas et al. examined the application of sentiment analysis in marketing research from three main perspectives, including the unit of analysis, sampling design, and methods used in sentiment detection and statistical analysis (Rambocas and Pacheco [2018](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9816550/#CR172)). Cheng et al. summarized techniques based on semantic, sentiment, and event extraction, as well as hybrid methods employed in stock forecasting (Cheng et al. [2022](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9816550/#CR57) ). Yue et al. categorized and compared many techniques and approaches in the social media domain. That study also introduced different types of data and advanced research tools, and discussed their limitations (Yue et al. [2019](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9816550/#CR235)).

Before machine learning technology became mature, researchers were particularly concerned about feature extraction methods. For example, Feldman summarized methods for extracting preferred entities from indirect opinions and methods for dictionary acquisition (Feldman 2013). Asghar et al. reviewed the natural language processing techniques for extracting features based on part of speech and term position; statistical techniques for extracting features based on word frequency and decision tree model; and techniques for combining part of speech tagging, syntactic feature analysis, and dictionaries (Asghar et al. 2014). Koto et al. discussed the best features for Twitter sentiment analysis prior to 2014 by comparing 9 feature sets (Koto and Adriani 2015). They found that the current best features for sentiment analysis of Twitter texts are AFINN (a list of English terms used for sentiment analysis manually rated by Finn Årup Nielsen) (Nielsen 2011) and Senti-Strength (Thelwall et al. 2012). Taboada sorted out the characteristics of words, phrases, and sentence patterns in sentiment analysis from the perspective of linguistics (Taboada 2016). Besides, Schouten and Frasinar conducted a comprehensive and in-depth critical evaluation of 15 sentiment analysis web tools (Schouten and Frasincar 2015). Medhat et al. (2014) and Ravi et al. (Ravi and Ravi 2015) also analyzed the early algorithms for sentiment analysis.

Sentiment analysis is one of the most promising methods for content analysis in social media, known as emotion AI or opinion mining, leads to natural language processing (NLP) and text analysis to systematically quantify, extract, identify, and study effective states and personal information. Sentiment analysis is widely applied in the voice of the customer materials such as survey responses and reviews.

Analysis of sentimental people can be achieved by millions of likes and tweets, but this interaction with such a post does not reflect the importance of the feelings towards these posts.

Machine learning is one of the common approaches for sentimental analysis. Classified training data is needed when using a machine learning approach. It then train the algorithm on the data structure to predict the classification of previously unseen data. The algorithms I will implement are Logistic Regression, Naïve Bayes and Linear Support Vector. Investigating and testing these algorithms is time consuming. that I will implement

Distributed file systems like Hadoop HDFS and distributed data storage techniques play a vital role in handling large datasets. They provide scalable and reliable storage and processing capabilities, making them ideal for big data applications. The emergence of Hadoop File Systems, Google File Systems and Network File Systems have changed the course of how data is managed in servers and has its own implications on Cloud Computing and Big Data management. Each file system offers its own advantages and challenges in terms of performance, fault-tolerance, consistency, scalability and availability. This opens an open debate on how these can be taken up for implementation. The choice of a feature available with each one of them has their own metrices that differentiates them from other file systems.

The main purpose of the Distributed File System (DFS) is to allow users of physically distributed systems to share their data and resources by using a Common File System. A collection of workstations and mainframes connected by a Local Area Network (LAN) is a configuration on Distributed File System. A DFS is executed as a part of the operating system. In DFS, a namespace is created, and this process is transparent for the clients.

Industries are using Hadoop broadly to analyze their datasets, because Hadoop architecture is based on a MapReduce which makes for scalable, modular, fault tolerant and cost effective computing approach.

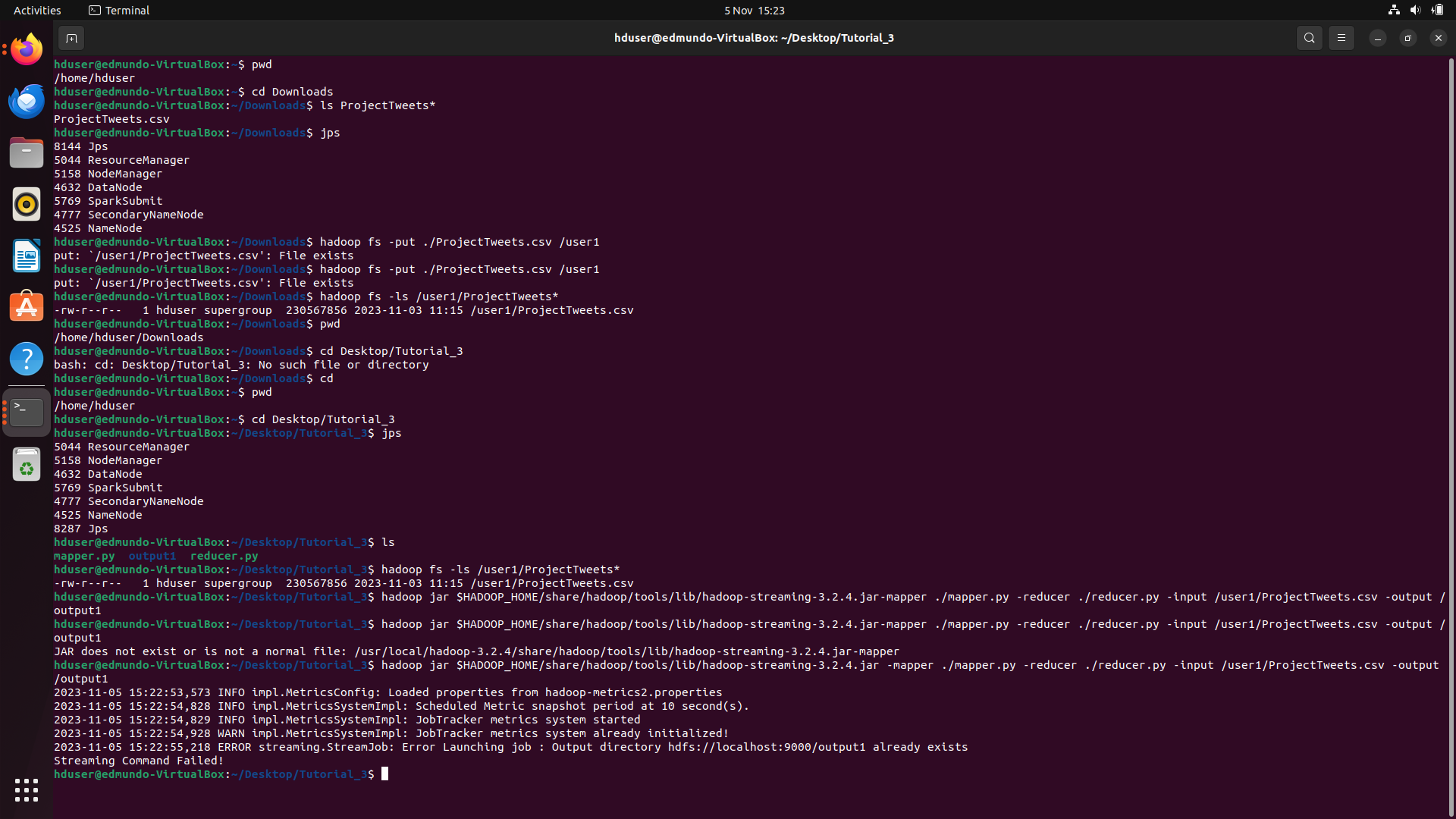
For Big Data Storage I started with MapReduce due to being a processing framework used in Hadoop to process and analyze large datasets stored in HDFS.

MapReduce framework works by dividing the data processing task into two main phases.

I applied Hadoop Streaming to design and execute MapReduce tasks using arbitrary executables as mapper and reducer.

I started by downloading the input csv file from Moodle named ProjectTweets.csv, move the file from Downloads folder into HDFS for MapReduce.

The steps I have used, are demonstrated in the following screenshots:



The output was stored in the output folder called output1.

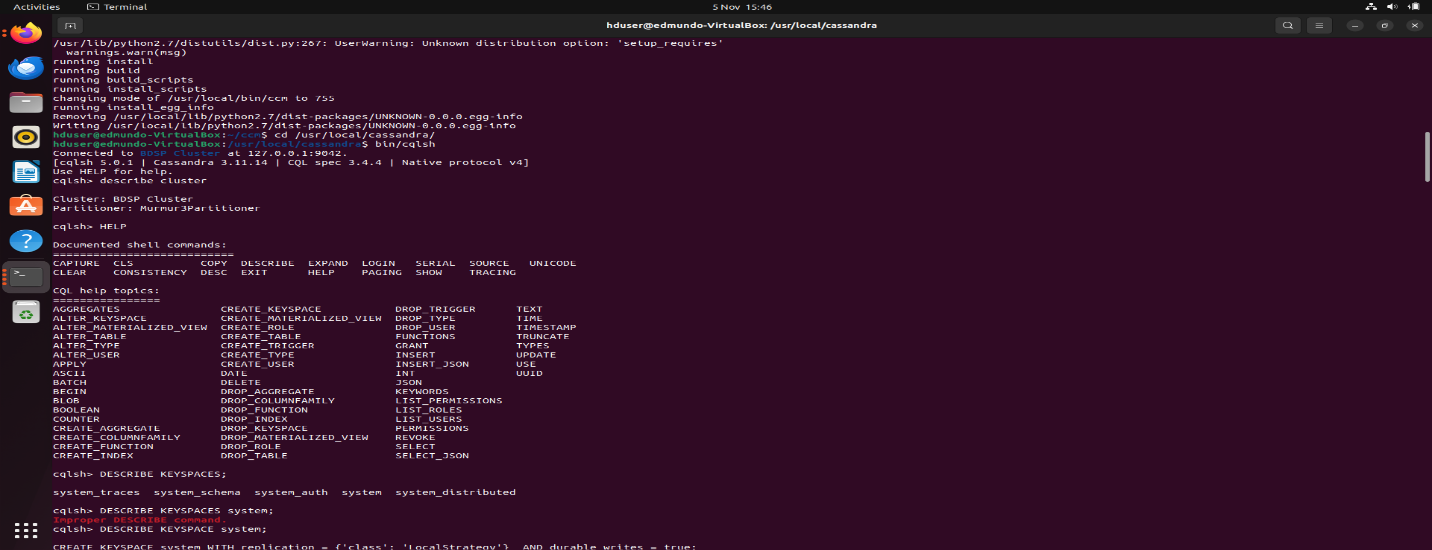
A screenshot of a computer

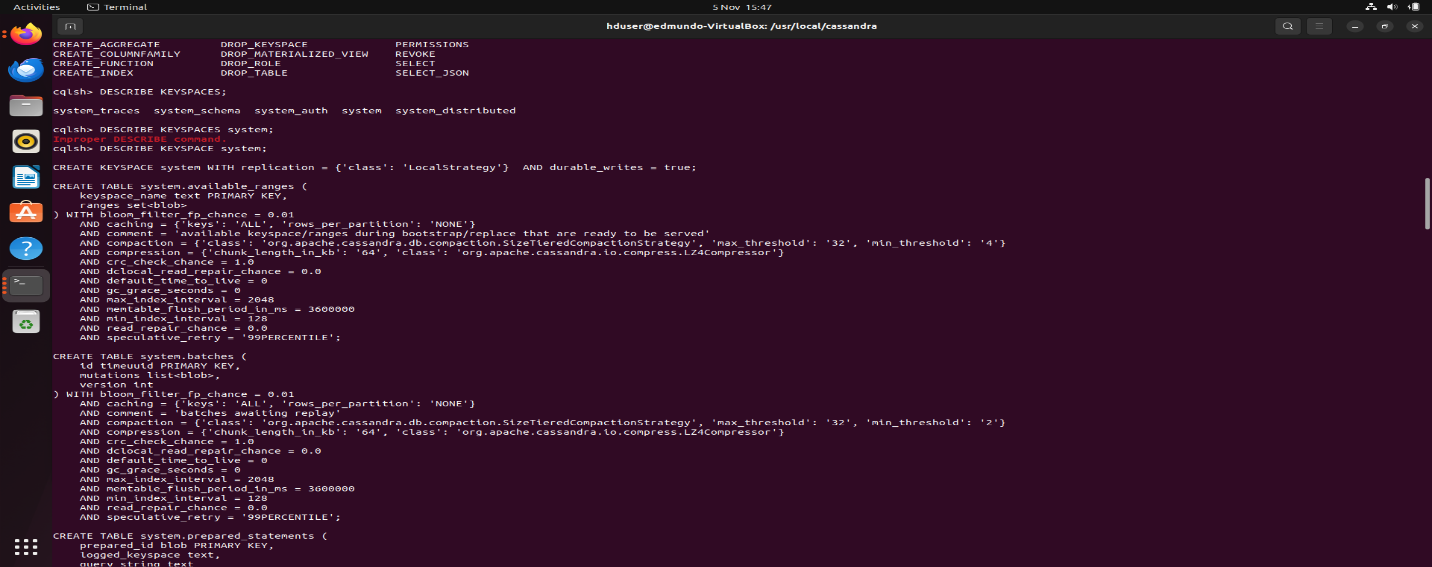
Description automatically generated

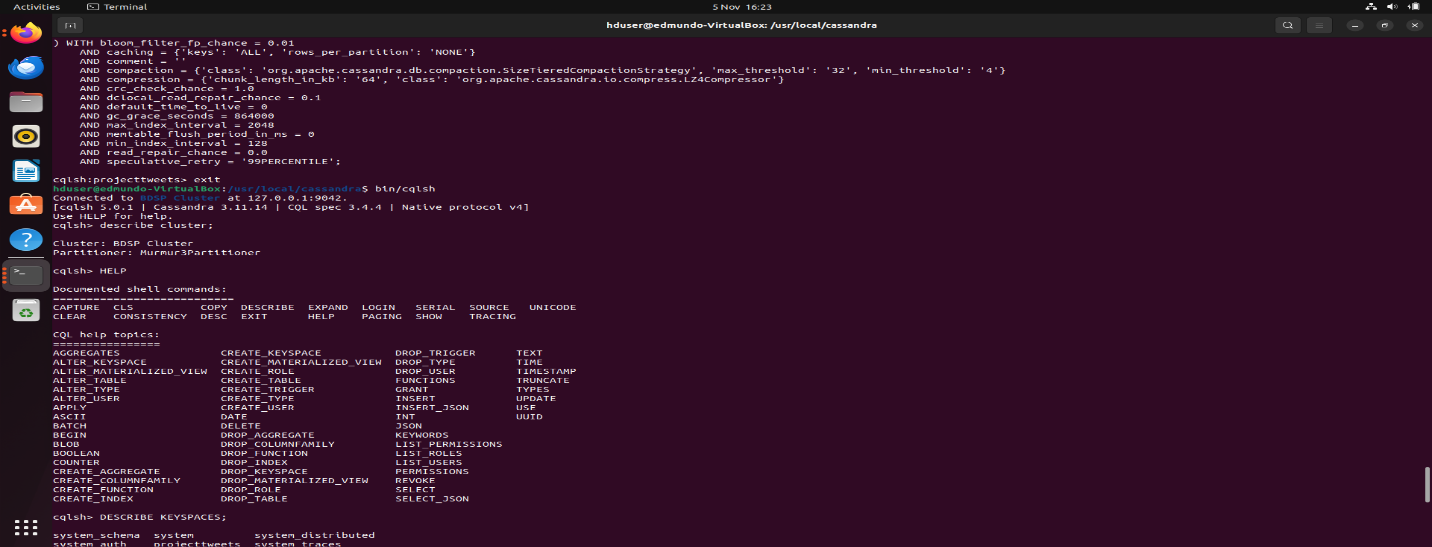
I successfully finished the execution of Hadoop streaming job. The result is stored and available in HDFS.

The next step was to apply Cassandra.

Cassandra is a highly scalable and distributed NoSQL database system designed to manage large volumes of data across multiple nodes and data centers. It supports a variety of functions that can be used to transform data. These functions can be used in Cassandra Query Language (CQL) queries to perform calculations, filtering and so on. It has High availability: Cassandra is designed to be highly available. Data is replicated across multiple nodes in the cluster, so it is still available even if some nodes are unavailable.Scalability: Cassandra is designed to be scalable. New nodes can be easily added to the cluster to handle increasing workloads. Low latency: Cassandra is designed to provide low latency reads and writes. This makes it a good choice for applications that need to handle large volumes of data.In the following screenshots, I will demonstrate the steps followed while using Cassandra:







My next step is to use Pyspark, which is the Python library for Apache Spark, it is a powerful open source data processing framework that provides an open source cluster-computing environment for large scale data processing. It is designed to be efficient and offers support for a wide of data processing tasks. Spark is designed for speed and ease of use and can handle large scale data.

Though Pyspark, I will interact with Spark using Python programming language. Pyspark is built on top of Spark and leverages its distributed computing capabilities.

While Pyspark provides a Python API to interact with Spark, its RDDs are the fundamental data structure in Spark, representing distributed collections of data and are partitioned across the cluster. Pyspark uses DataFrames which are a high level distributed and tabular data structure. Dataframes provide a more intuitive and efficient way to work with structured data compared to RDDs. To speed up the Hadoop computing programme process, Spark was implemented by Apache. Hadoop is used in two ways by Spark: one is storage and the second is computation. Since Spark has its own device for cluster, it only uses Hadoop for storage purposes.

My process while applying Pyspark was to perform Exploratory Data Analysis to understand the dataset, followed by data cleaning, filter and extract hashtags from the list of words, followed by processing text data by cleaning and transforming the text of tweets.

Next stage was to perform Sentiment Analysis which is a method used to carry out an analysis of the sentiment of the text in terms of positive or negative.

This section was centred in analysing sentiments of text from Tweets

The dataset was divided into Training data and testing data. Testing data was then predicted and the accuracy of the model was calculated using three machine learning algorithms – Naïve Bayes, Decision tree Classifier and Random forest. The aim is to evaluate the performance of the algorithms as to which algorithm provides a better accuracy.

In training phase the model starts with preprocessing step, which is a crucial step to give the model valuable results and success.

Preprocessing consists of four primary processes: text cleaning, tokenization, removing stop words, lemmatization and stemming. Then extract specific features from the tweet’s text that should be used as an input to the classification algorithms.

I used TF-IDF to construct a bag of words where tweets can be divided into words and generate a vector.

In the section of Big Data, because the dataset is not labeled by a sentiment , I will use a Vader Lexicon Model, which is a lexicon based approach that considers pre trained model and will be used to identify tweets sentiment. It has the advantage of quickly searching the list of sentiment words and their polarity. One disadvantage of VADER-based approach is that it is domain independent. It works by taking a word and classifying its sentiment to positive, negative or neutral. Which can performed with polarity calculation of sentiment words.

There are three main steps to complete this analysis:

Step 1 – This step involves categorizing words as negative, or positive in two groups, based on the frequency of negative or positive words. The the polarity score for each tweet can be conveniently determined.

Step 2 – The tweet polarity is determined bu summing the tweet value of each selected feature.

Step 3 – to finish a tweets feeling, the following rules are estabilished based on its polarity,

Polarity Score < 0, Negative

Polarity Score = 0, Neutral

Polarity Score > 0, Positive.

Another approach that I implemented on this project was time series analysis, which can also be used to investigate changes in sentiment over time, either to understand the role of sentiment over time, either to understand the role of sentiment in a event or changes in popularity over time.

A time series is a series of data points indexed in time. The fact that time series data is ordered makes it unique in the data space because it often displays serial dependence. Serial dependence occurs when the value of a datapoint is statistically dependent on another data point at another time.

I attempted to analyze temporal trends by specifying the number of tweets at different time periods, such as daily, week 1, month 1, 3 months.

**Results after Pre-processing the data.**

This process reduced the number of words in vocabulary significantly. Considering this result, the preprocessing phase was essential to help me clean and remove unnecessary data.

**Conclusion**

In this project, a systematic literature review has been conducted to identify suitable approach for performing sentiment analysis and time series analysis on ProjectTweets.

There was evidence that VADER is an appropriate technique for sentiment analysis. VADER also follows grammatical and syntactical conventions for expressing and emphasizing sentiment intensity.

According to sentiment values the results show that the KDE distribution regarding each sentiment is positive, negative and neutral. From this study, I can say that people’s reactions vary day to day from posting their feelings on social media, specifically Twitter.

**Future Work**

In this project, sentiment analysis is only done in English. For future implementation, sentiment analysis can be done in other languages as well.

**Challenges**

During the period of this project, I faced many health issues as I was made homeless, by my landlord, thankfully 1 week ago, I found a small place so in between full time working, moving from Wicklow to Dublin, I still faced the issue of internet access. I had to use my mobile internet, which made it extremely difficult to complete this work, especially to run codes or to even open new pages.

But overall, it was a good but though experience with a lot of learning. Or as I call it, a show of resilience. I am grateful to CCT and Springboard for this experience and journey.

**References**

Abiodun, O.I., Jantan, A., Omolara, A.E., Dada, K.V., Mohamed, N.A. and Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. Heliyon, 4(11), p.e00938. doi:https://doi.org/10.1016/j.heliyon.2018.e00938.

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M. and Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, [online] 8(1). doi:https://doi.org/10.1186/s40537-021-00444-8.

Amazon Web Services, Inc. (n.d.). What is Distributed Computing? - Distributed Computing - AWS. [online] Available at: <https://aws.amazon.com/what-is/distributed-computing/>.

Analytics Vidhya. (2021). Integration of Python with Hadoop and Spark. [online] Available at: https://www.analyticsvidhya.com/blog/2021/05/integration-of-python-with-hadoop-and-spark/.

Chai, J., Zeng, H., Li, A. and Ngai, E.W.T. (2021). Deep learning in computer vision: A critical review of emerging techniques and application scenarios. Machine Learning with Applications, p.100134. doi:https://doi.org/10.1016/j.mlwa.2021.100134

Clark, T.H. (2021). 7 Essential Python Skills for Research. [online] Medium. Available at: https://towardsdatascience.com/7-essential-python-skills-for-research-496e1888e7c2 [Accessed 27 Sep. 2023].

Ch, K. and rakant (2020). Fundamentals of Distributed Systems | Baeldung on Computer Science. [online] www.baeldung.com. Available at: <https://www.baeldung.com/cs/distributed-systems-guide>.

Faik, L. (2022). Graph Neural Networks: Merging Deep Learning With Graphs (Part I). [online] blog.dataiku.com. Available at: https://blog.dataiku.com/graph-neural-networks-merging-deep-learning-with-graphs [Accessed 10 Sep. 2023].

GeeksforGeeks. (2018). Association Rule. [online] Available at: <https://www.geeksforgeeks.org/association-rule/>

GeeksforGeeks. (2020). Cosine Similarity. [online] Available at: <https://www.geeksforgeeks.org/cosine-similarity/>.

GeeksforGeeks. (2017). Dynamic Programming - GeeksforGeeks. [online] Available at: <https://www.geeksforgeeks.org/dynamic-programming/>.

GeeksforGeeks. (2023). Distributed Computing System Models. [online] Available at: https://www.geeksforgeeks.org/distributed-computing-system-models/ [Accessed 10 Sep. 2023].

GeeksforGeeks. (2017). Introduction to Convolution Neural Network. [online] Available at: <https://www.geeksforgeeks.org/introduction-convolution-neural-network/>

Google Developers. (2019). Overview of GAN Structure  |  Generative Adversarial Networks. [online] Available at: <https://developers.google.com/machine-learning/gan/gan_structure>.

Jain, A. (2020). What is Auto-Encoder in Deep Learning? [online] Analytics Vidhya. Available at: https://medium.com/analytics-vidhya/what-is-auto-encoder-in-deep-learning-5d668f94651b [Accessed 10 Sep. 2023].

Kumaraswamy, A. (2018). 10 reasons why data scientists love Jupyter notebooks. [online] Packt Hub. Available at: https://hub.packtpub.com/10-reasons-data-scientists-love-jupyter-notebooks/.

Picton, P. (1994). What is a Neural Network? Introduction to Neural Networks, pp.1–12. doi:https://doi.org/10.1007/978-1-349-13530-1\_1.

Saxena, D., Kumar, J., Ashutosh Kumar Singh and Schmid, S. (2023). Performance Analysis of Machine Learning Centered Workload Prediction Models for Cloud. 34(4), pp.1313–1330. doi:https://doi.org/10.1109/tpds.2023.3240567.

Simplilearn.com. (n.d.). Hadoop Architecture in Big Data Explained: A Complete Guide with Its Components. [online] Available at: <https://www.simplilearn.com/tutorials/hadoop-tutorial/hadoop-architecture>.

Tomar, A. (2019). ITEM-ITEM Collaborative filtering Recommender System in Python. [online] Medium. Available at: <https://medium.com/@tomar.ankur287/item-item-collaborative-filtering-recommender-system-in-python-cf3c945fae1e>

Ünal, H.T. and Başçiftçi, F. (2021). Evolutionary design of neural network architectures: a review of three decades of research. Artificial Intelligence Review. doi:https://doi.org/10.1007/s10462-021-10049-5.

Wason, R. (2018). Deep learning: Evolution and expansion. Cognitive Systems Research, 52, pp.701–708. doi:https://doi.org/10.1016/j.cogsys.2018.08.023.

**WORD COUNT**

3467 including References