Sentimental Analysis Using MapReduce and NoSQL databases for Depression

**Chapter 1: Introduction**

We are surrounded by amazing technological marvels today. Streets are filled with autonomous vehicles, drones buzz overhead, and robots dance across our floors. And yet, despite these amazing advancements, we don't appear to be any happy. Most psychological well-being indicators have stabilized, although rates of suicide and depressive disorders have increased. These days, depression frequently contributes to a variety of problems. Diagnoses of depression (and the ensuing therapy) are frequently postponed, hazy, or even forgotten. Around 800,000 individuals die by suicide each year all across the world, and melancholy is the primary cause of disability. Today, deaths of young indivuduals between the ages of 15 and 29 are currently most frequeintly caused by suicide. Treatment is usually required for depressive disorders since they are frequently detected too late, incorrectly, or not at all. [1]

The development of newer, more advanced technologies does not guarantee improved mental health. While modern medicine and the practices that followed have helped individuals live longer, the present issue is mental health. People who struggle with mental health issues typically die 20 years earlier than the average population, according to statistics from the World Health Organization. Physical care has advanced greatly, but there is still much room for improvement in mental care. The online social networking platforms enable us stay isolated while still allowing us to stay connected to the outside world. We are a byproduct of stress thanks to the mechanization that enables us to participate and make efficient use of resources.

Our actions on social media can be seen as one of the markers for early medical intervention, just as your body exhibits symptoms of sadness through increased stress, loss of hunger, heart rate, body temperatures, and so many others. The twitter database estimates that six thousand tweets are published per second, or about 200 billion tweets annually. According to Smart Insights' "Global Social Media Statistics Research" [2], 2 hours and 29 minutes per day is the average amount of time that users spend on social media.

Twitter has improved as a better option because users have more freedom to use pseudonyms and interact with unidentified individuals more regularly. This kind of invisibility can promote open discussion without the worry of criticism; users can express their opinions on a wide range of topics, both favourable and negative.

Tweets are especially useful for analysis because they are frequently produced in daily events and activities. It provides a thorough and logical method for documenting behavioural traits that are crucial for comprehending a person's thoughts, mood, social interactions, communication, and activities.

For processing a huge data collection with unstructured data, Hadoop MapReduce has emerged as the clear choice among researchers and academics in recent years [3, 4]. A distributed parallel programming approach called MapReduce can process a lot of data in a uniform setting.

The paper now emphasizes on parallel processing of Big Data Analytics, reduction of latency, and enhancing computation capabilities by utilizing Hadoop Distributed Framework with Mapper-Reducer Architecture with bulk data storage into NoSQL Hash-key based MongoDB with further encompasses the use of Apache Spark as preprocessing pipeline using Lazy Programming.

**Chapter 2: Literature Review**

[5] asserts that most clients are immune to therapies because they don't understand the cause, resist using drugs like antidepressant medications fail to recognize the need for therapy, are afraid of conveying their feelings, and avoid talking about painful topics in therapy, in spite of the fact that thousands of individuals suffer from depression every day.

According to [6], just 13% of the sample population in their study received counselling and psychotherapy that was at least somewhat effective.

By developing several computational models that accurately predicted depressive disorders and PTSD (Post-Traumatic Stress Disorder) within twitter accounts, the authors [7] have successfully illustrated the advantages of using a data-driven technique for the early diagnosis of depression. Additionally, [8] concentrated on predicting sorrow from a twitter user's tweets in their paper by creating their own dataset. With the dataset, they used RNN, CNN, as well as GRU deep learning algorithms to make predictions.

The study by [9] addresses the problem of separating user and content sentiment.

They put forth a hybrid model that combines characteristics from several social context layers. The model is assessed using a range of datasets. In a publication by [10], a hybrid technique was described that uses sentiment analysis of movie reviews to improve a preliminary recommendation list created by combining collaborative filtering and content-based algorithms.

Scalable computing systems need to perform consistently even under heavier workloads [11]. To address the scalability challenges related to Twitter data, researchers have suggested parallel processing environments, the Apache Hadoop framework, the MapReduce programming approach, programming languages "R" and "Python" [12]. Additionally, they favor using Flume to capture or extract real-time Twitter data, which they then store in centralized databases like Hadoop Distributed Filesystem (HDFS) along with HBase. It makes analyzing the twitter data collection simple.

**Chapter 3: Data and Methodology**

**3.1 Data**

The data has been accessed from Twitter API. Due to its extensive and immediate availability, which gives access to a wide spectrum of public viewpoints, the Twitter API is advantageous for sentiment research. It enables widespread monitoring and analysis of user feelings, trends, and responses, resulting in more precise and current sentiment analysis results.

|  |  |
| --- | --- |
| **Fields** | **Description** |
| Created Date | Date |
| Text | Tweet created by the person |

*Table 1: Data Description*

**3.2 Methodology**

Sentimental Analysis is drawing a wide range of academics from around the world. Several leading MNCs, including Facebook, Twitter, and others, are concentrating on studying the attitudes of their users. Twitter statistics will be the primary topic of this essay. The sentimental analysis process consists mostly of parts. These are what they are:

A screenshot of a computer

Description automatically generated with medium confidence

Figure : Proposed Framework

1. **Live Twitter Data Extraction and Storage:** Live Twitter data extraction is only possible with authorized keys. Create a twitter account first in order to receive authorized keys. Utilise these keys to download live Twitter information and save it to a file. Information from Twitter has been saved in a form of text file for this paper. To stay up to date on current public sentiment, trends, and opinions, businesses and researchers must extract and store live Twitter data. It makes quick decisions possible as well as competitive analysis and crisis management. Organizations can track industry conversations, track new issues, and obtain knowledge about customer preferences and behavior by collecting and storing real data. Additionally, storing real-time data guarantees that it will be accessible for upcoming analyses, comparative studies, and retrospective research. Additionally, it enables companies to track brand reputation, identify influencers, and analyze client comments. All things considered, real-time Twitter data extraction along with storage offer a useful resource for comprehending market dynamics, consumer mood, and business trends, enabling data-driven initiatives and bettering client experiences [13].
2. **Using the Hadoop MapReduce Network:** Due to its support for distributed processing of huge datasets, the Hadoop MapReduce architecture is useful for sentiment analysis. It permits sentiment analysis operations to be carried out in parallel over a cluster of computers, accelerating the analysis. The large amount of data required in sentiment analysis can be handled more easily and effectively because to MapReduce's scalable architecture, which also makes it possible to gain quicker knowledge of sentiment patterns and trends. Due to the sheer quantity and complexity of the data involved, employing the Hadoop MapReduce network as a big data tool is crucial for depression analysis. Processing significant amounts of data from numerous sources, including electronic health records, patient questionnaires, and social media, is frequently necessary for depression analysis. The parallel and distributed processing capabilities of the MapReduce framework make it possible to process and analyse this enormous volume of data quickly and effectively.It enables the fault tolerance required for dealing with hardware failures, the scalability required to tackle big data concerns, and the capacity to run complicated calculations throughout a cluster of devices. The Hadoop ecosystem also provides a number of tools and packages that work well with MapReduce and add functionality such as data manipulation, machine learning, and visualization, all of which are necessary for thorough and intelligent analysis of depression [14].

In general, the Hadoop MapReduce framework is used as

A picture containing text, diagram, screenshot, line

Description automatically generated

Figure : MapReduce Framework [15]

In the diagram, we can see that the input file is added to the data, with a certain list of tasks that can be done. Then the map function is used and then the reduce function below getting the required output.

Map phase as well as Reduce phase are the two key steps that make up the Map Reduce task. As implied by its name, the primary function of a map is to map the incoming data into key-value pairs.A key-value pair with the address id as the key and the value being the information that the map really stores might be the input. Every single one of these input key-value pairs will be subjected to the Map() function's memory repository execution, which creates the intermediate key-value pair that serves as the input for the Reducer or Reduce() function.

The intermediate key-value pairs that serve as the input for the reducer are sorted and then sent to the reduce() method after being shuffled. According to the reducer algorithm created by the developer, reducer aggregates or groups the data depending on its key-value pair.

How the task tracker and job tracker handle MapReduce:

Function Tracker: Since there may be thousands of data nodes accessible in the cluster, the function of Job Tracker primarily to oversee all resources and jobs throughout the cluster as well as to organise each map within the Task Tracker operating on the identical data node.

Task Tracker: It is comparable to the actual slaves carrying out the orders provided by the Job Tracker. Each node in a cluster that is capable of doing the Map as well as Reduce work is equipped with this work Tracker, which is deployed as per Job Tracker's instructions [16].

1. **Usage of MongoDB to save data:** After MapReduce's reduce() function, data can be saved to MongoDB for durable storage, simple sharing, and additional analysis. In order to facilitate cooperation, system integration, and further data processing chores, MongoDB offers a central database for keeping and retrieving the results.An free and open-source NoSQL database management system is called MongoDB. Traditional relational databases can be replaced with NoSQL (Not simply SQL) databases. Working with sizable, distributed data sets makes good use of NoSQL databases. MongoDB is a technology that can manage, store, and retrieve information that is document-oriented. As a NoSQL database, MongoDB uses collections and documents in place of the tables and rows found in relational databases. Key-value pairs, the fundamental data type in MongoDB, make up documents. Document sets are found in collections, which are equivalents of SQL tables. Numerous programming languages, including C#, C++, C, and Go, Java, Python, Ruby, and Swift are supported by MongoDB. Users have access to servers in MongoDB setups so they can utilize those servers to build databases. Data is kept in MongoDB as records, which are composed of records and collections of documents. The information a user wishes to store in a MongoDB database is included in documents. There is field as well as value pairs in documents. In MongoDB, they serve as the fundamental data unit. A main key will be used as a primary identifier for documents. The structure of a document can be altered by adding or removing new or current fields. A main key will be used as a primary identifier for documents. The structure of a document can be altered by adding or removing new or preexisting fields [17].
2. **Using Apache PySpark for reading the data**: Because Apache PySpark offers an extensible and distributed processing framework that enables effective handling of enormous databases, execution in parallel, and integration into other Spark ecosystem elements for sophisticated data analysis and processing, it is utilized here to receive data from MongoDB. Apache Spark and Jupyter notebooks are frequently used together. One of the most popular analytics engines for processing enormous amounts of data is Spark, a general-purpose computational framework that is free source. Spark's core concept is distributed computing, which includes distributing the task among a number of worker nodes to avoid exhausting all of a server's processing resources.This is how the idiom "many hands make light work" is applied technically. Spark is effective and can take in data from many different sources, including relational databases, HDFS file systems, and even MongoDB using the MongoDB Spark Connector [18, 19]
3. **Using Cassandra for storing the .py file:** After reading the.py file from MongoDB using PySpark, Cassandra is used to store it since it is a robust and distributed database which can manage big data volumes and offers a dependable and effective storage solution for a variety of data types, especially files like.py scripts. The purpose of this is to model the data.

With a lack of single point of failures and the ability to handle massive volumes of data across numerous commodity servers, Apache Cassandra is a distributed database that is highly scalable and performant. It belongs to the NoSQL database family. Clusters of nodes make up the core of Cassandra's architecture. The peer-to-peer architecture of Apache Cassandra is quite like that of DynamoDB and Google Bigtable.

It is essential to Cassandra's design that each node be treated equally and with the same degree of importance. Every node is the precise location where a given piece of data is kept. A data center is made up of a collection of connected nodes. A cluster is made up of all the data centers that can store data for processing.

Information is stored and accessed in Cassandra using a partitioning method. Where a data set's primary copy is kept is decided by a partitioner. Direct format for using nodal tokens is supported here. According to a partition key, each node has ownership or is accountable for a particular set of tokens. The location of data storage is determined by the partition key.

Cassandra also functions by replicating data among nodes. Replica nodes are these secondary nodes, and the replication factor (RF) determines how many replica nodes are required for a certain data collection. When three nodes encompass the identical token range and store the same data, the replication factor is 3. The Cassandra system's reliability depends on having many replicas [20].

Additional nodes hold the same data, therefore it is almost never completely lost, even if one node stops working temporarily or permanently. Even better, once a temporally disturbed node is back on course, it gets a notification of any data actions it might have received and then comes up to speed to resume operation.

1. **Taking the relevant information from the saved .py file:** Now the data will be used as a .csv file to model for understanding the sentiment.
2. **Arima for Model:** A well-liked and effective time series forecasting model called ARIMA (Autoregressive Integrated Moving Average) is used to examine and estimate future values based on historical data. Autoregression (AR), differencing (I), and moving average (MA) are the three essential parts of ARIMA. The moving average component considers the dependence on previous error terms, the differencing component handles non-stationarity by distinction the series, and the autoregressive element captures the link among an observation and a specific number of lagged observations. In several fields, including economics, finance, and weather forecasting, ARIMA models are frequently utilised. Through the use of statistical techniques like maximum likelihood estimation, the model's parameters are estimated. ARIMA works well for short-term forecasting because it is good at identifying patterns and trends that emerge quickly. It might, however, struggle to deal with seasonality or long-term tendencies. These restrictions are addressed by additions like SARIMA and ARIMAX, which enable more accurate and thorough time series analysis [21].
3. **Utilizing the Dashboard**: The Dashboard is devised using Python’s Plotly and Dash libraries in a .py file and the seven, thirty- and ninety-days dashboards are plotted.

**Chapter 4: Results and Discussion**

**4.1 Results**

A computer screen shot of a computer

Description automatically generated with low confidence

Figure : Hadoop Installation

A screenshot of a computer

Description automatically generated with medium confidence

Figure : UI Dashboard

A computer screen with white text

Description automatically generated with low confidence

Figure : Hadoop Node Running

A computer screen shot of a computer

Description automatically generated with low confidence

Figure : Local Host Key Creation

The figures of 3, 4, 5, 6 discuss about the installation of Hadoop Framework, in addition images showcasing the Hadoop Node running, Local Host Key Creation etc., to create a Haddop Environment.

**A screenshot of a computer program

Description automatically generated with medium confidence**

Figure : Mongo Connection with PySpark

Figure 7 showcases the code snippet of Mongo Connection with PySpark to read the file.

**A screenshot of a computer

Description automatically generated with medium confidence**

Figure : ARIMA Model

Figure 8 shows the summary statistics of the ARIMA model used to analyse the sentiments of twitter data based on words related to Depression. The NLTK package was utilised prior as a pre-processing tool to remove unnecessary data.

**A picture containing text, screenshot, font, software

Description automatically generated**

Figure : Forecast Prediction for 7, 30 and 90 days

The predicted data in figure 9 showcases the prediction results for 7 days, 30 days and 90 days dataset.

A picture containing screenshot, line, electric blue, rectangle

Description automatically generated

Figure : Time Series Plot

Figure 10 discusses forecasting for 7, 30, 90 days sentiments on depression.

A picture containing screenshot, line, rectangle, design

Description automatically generated

Figure : Dashboard

The dashboard and the figures above show the predicted data

**4.2 Discussion**

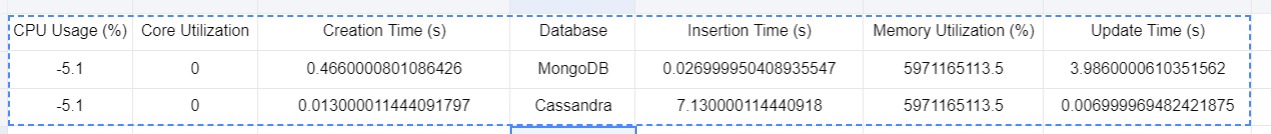


Figure : Database Comparison

In comparison to MongoDB and Cassandra, MongoDB utilises fast insertion in the form of bulk insertion and is much more suitable for raw data transactions. Further, the NoSQL nature of MongoDB allows the raw data to be stored in much rudimentary fashion column types or column lengths in SQL setups.

However Cassandra being a SQL- similar database allows enhanced computation and generic pre-processing pipelines after the pre-processed dataset. It offers a fixed or rigid tabular construct allowing attribute groupings in the form of aggregate summations etc., As in our case, MongoDB was not only able to store the data with very less time-lag, from the Twitter API, but also, the pre-processed data which could be stored in Cassandra was utilised for aggregating the date-wise sentiments necessary for dashboard computation.

Hence, the study concludes the utility of MongoDB before pre-processing and Cassandra after pre-processing.

During the pre-processing pipeline, the paper utilises a Get-Data Python script to save the output as a CSV file- MongoDB which is then loaded as a input to the Mapper and Reducer codes and the pre-processing file with the help of Apache Spark and lazy programming technologies is saved into Cassandra database.

**Chapter 5: Conclusion**

In conclusion, a strong foundation for processing, storing, and analysing massive amounts of data is provided by the combination of a Hadoop MapReduce cluster, MongoDB, Apache PySpark, and ARIMA models in sentiment analysis of depression prediction.

Sentiment analysis activities can be completed more quickly because to the rapid parallel processing of data provided by the Hadoop MapReduce cluster. The capacity to handle semi-structured or unstructured information with flexibility and scalability is made possible by storing the data in MongoDB. Data retrieval and storing in Cassandra are made easier by Apache PySpark's seamless integration with Hadoop. This enables effective data management.

By incorporating temporal patterns and trends in depressive emotion, ARIMA models for time series analysis improve the framework. ARIMA models help with future forecasts by helping to identify underlying trends in historical data. Overall, this integrated strategy equips investigators and statisticians to manage and analyse enormous amounts of data effectively, take advantage of distributed computing, archive information flexible, and record temporal dynamics. It aids in predicting and forecasting depression-related trends and patterns and advances our understanding of depressive emotion.

**Chapter 6: References**

1.     KFF. (2019). Suicide represents 2nd leading cause of death among young people ages 15-29, who notes on World Mental Health Day. Available at: https://www.kff.org/news- summary/suicide-represents-2nd-leading-cause-of-death-among-young-people-ages-15-29- who-notes-on-world-mental-health-day/#:~:text=%E2%80%9CWorldwide%2C%20800%2C000%20people%20die%20by,%E2 %80%9D%20(10%2F9)

2.     Smart Insights. (n.d.). "Global social media statistics research summary 2022." Available at: <https://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/>

3.     Lam, C., Davis, M., & Gaddam, A. (2016). Hadoop in Action, 2nd ed. Manning Publications.

4.     Kenekar, T.V., & Dani, A.R. (2017). "An efficient private FIM on Hadoop MapReduce." In IEEE International Conference on Automatic Control and Dynamic Optimization Techniques, Pune, India, Sept 9–10, 2017.

5.     Dunkin, M.A., "Avoiding depression treatment, making excuses." WebMD. Available at: [https:/www.webmd.com/depression/features/depression-why-arent-you-getting-treatment](https://chat.openai.com/www.webmd.com/depression/features/depression-why-arent-you-getting-treatment) [Accessed August 9, 2022].

6.     Puyat, J.H., Kazanjian, A., Goldner, E.M., and Wong, H. (2016). "How often do individuals with major depression receive minimally adequate treatment? A population-based, data linkage study." The Canadian Journal of Psychiatry, 61(7), 394-404.

7.     Reece, A.G., Reagan, A.J., Lix, K.L., Dodds, P.S., Danforth, C.M., and Langer, E.J. (2017). "Forecasting the onset and course of mental illness with Twitter data." Scientific Reports, 7(1), 1-11.

8.     Singh, D., and Wang, A. (2016). "Detecting depression through tweets." Stanford University CA, 9430, 1-9.

9.     Sánchez-Rada, J.F. and Iglesias, C.A., 2020. CRANK: a hybrid model for user and content sentiment classification using social context and community detection. Applied Sciences, 10(5), p.1662.

10.  Wang, Y., Wang, M. and Xu, W., 2018. A sentiment-enhanced hybrid recommender system for movie recommendation: a big data analytics framework. Wireless Communications and Mobile Computing, 2018.

11.  Ahmad, F., Chakradhar, S.T., Raghunathan, A. and Vijaykumar, T.N., 2014. {ShuffleWatcher}: Shuffle-aware Scheduling in Multi-tenant {MapReduce} Clusters. In 2014 USENIX Annual Technical Conference (USENIX ATC 14) (pp. 1-13).

12.  Leo, S. and Zanetti, G., 2010, June. Pydoop: a python mapreduce and hdfs api for hadoop. In Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing (pp. 819-825).

13.  Sheela, L.J., 2016. A review of sentiment analysis in twitter data using Hadoop. International Journal of Database Theory and Application, 9(1), pp.77-86.

14.  Madani, Y., Erritali, M. and Bengourram, J., 2019. Sentiment analysis using semantic similarity and Hadoop MapReduce. Knowledge and Information Systems, 59, pp.413-436.

15.  Al-Alem, F.A., Big Data Analysis: Implementations of Hadoop Map Reduce, Yarn and Spark.

16.  Malidev, D. (2020) MapReduce architecture, GeeksforGeeks. Available at:https://www.geeksforgeeks.org/mapreduce-architecture/ (Accessed: 15 May 2023).

17.  TechTarget. (n.d.). "MongoDB." SearchDataManagement, Available at: <https://www.techtarget.com/searchdatamanagement/definition/MongoDB> (Accessed: 25 May 2023).

18.  Intellipaat. (n.d.). "PySpark Tutorial." Intellipaat, Available at: <https://intellipaat.com/blog/tutorial/spark-tutorial/pyspark-tutorial/> (Accessed: 25 May 2023).

19.  MongoDB. (n.d.). "Getting Started with MongoDB, PySpark, and Jupyter Notebook." MongoDB Blog, Available at: <https://www.mongodb.com/blog/post/getting-started-with-mongodb-pyspark-and-jupyter-notebook> (Accessed: 25 May 2023).

20.  BMC. (n.d.). "Apache Cassandra: Introduction, Architecture, and Use Cases." BMC Blogs, Available at: <https://www.bmc.com/blogs/apache-cassandra-introduction/> (Accessed: 25 May 2023).

21. McClellan, C., Ali, M.M., Mutter, R., Kroutil, L. and Landwehr, J., 2017. Using social media to monitor mental health discussions− evidence from Twitter. Journal of the American Medical Informatics Association, 24(3), pp.496-502.