A logo for college computing

Description automatically generated

Summer 2023 August Repeat Assessment

**Dovlet REYIMOV 2022341**

Github: <https://github.com/dovletPanda/big_data>

Word: 1304

1. INTRODUCTION

In the era of information and social media, the tremendous amount of data created on a daily basis offers a rare opportunity to uncover understanding into human sentiments and behaviors. This project delves into the realm of sentiment analysis and time series forecasting, harnessing the power of big data processing techniques to glean meaningful patterns from a dataset of 1.6 million tweets extracted using the Twitter API. By unifying distributed data processing systems, NoSQL databases, and cutting-edge machine learning algorithms, we intend to give a comprehensive scrutiny of sentiment alterations over time and precise forecasts for the future.

In a world where user-generated content has become a ubiquitous form of communication, social media platforms serve as a treasure trove of opinions and emotions. Through the lens of sentiment analysis, we seek to decipher the prevailing sentiments within the dataset. Sentiment analysis, a subfield of natural language processing, allows us to quantify emotions expressed in text, giving us a nuanced understanding of the user's perspective. This analysis not only reflects the sentiment associated with a particular event or issue, but also provides a way to uncover the underlying factors that influence public opinion.

In addition, the development of big data technologies has revolutionized how we process, store, and interpret voluminous datasets. By utilizing distributed data processing systems such as Hadoop MapReduce and Spark, we ensure proficient handling of the massive tweet dataset. The NoSQL database, known for its scalability and flexibility, house structured tweet data, facilitating seamless data retrieval and manipulation. These options stem from the need for the scalability, fault tolerance, and fast querying capabilities fundamental to extract meaningful insights from voluminous data. This report delves into the intricacies of these technologies and provides a thorough understanding of their role in analytics.

1. SENTIMENT ANALYSIS AND TIME SERIES FORECASTING OF TWITTER DATA

We started a large-scale project to analyze sentiment and predict time series on a dataset of 1.6 million tweets that were obtained using the Twitter API. Our process included the following crucial steps:

* 1. Information Preparation

- Used the Twitter API to extract tweets, which were then saved in the Hadoop HDFS distributed storage system.

- Converted the unstructured data into a format with the following fields: ids, date, flag, user, and text.

- For effective querying, the transformed data was loaded into a suitable NoSQL database (MongoDB).

* 1. Sentiment Extraction

Natural language processing techniques were used to extract sentiment from the text of the tweets. Assigned sentiment scores to each tweet using a pre-trained sentiment analysis model (like VADER or BERT). Based on sentiment scores, classified tweets as positive, negative, or neutral.

A diagram and a pie chart

Description automatically generated

* 1. Time Series Forecasting

In this section we have examined the ARIMA and LSTM time series forecasting techniques. Divided the data into time periods of daily for one week, weekly for one month, and monthly for three months. Then, created ARIMA and LSTM models using StatsModels and TensorFlow libraries. However, here we have a different approach, that should be explained. We have aggregated data points and spreaded over the time horizon. Then, we have taken the proportion of aggregated data points to provide a robust system for forecasting. Historical sentiment data to train the models to predict sentiment scores for the given time intervals.

* 1. Dynamic Dashboard Creation

In the dynamic dashboard creation section, we have created an interactive dashboard using tkinter and visualization libraries of Python. We have incorporated the results of the sentiment forecasting, displaying the predicted sentiment scores for the predetermined intervals as well as sentiment trends over time.

1. RATIONALE AND JUSTIFICATION FOR CHOICE
   1. Data Processing and Storage:

We have selected a distributed environment, here it will be Hadoop HDFS for initiating data storage due to its scalability and fault tolerance. Moreover, MongoDB was selected as the NoSQL database, because its document-oriented sake and beneficial query capabilities

* 1. Programming Language

We have used Python in the whole project for data processing, sentiment analysis, and model development because of its extensive libraries and ease of use.

* 1. Sentiment Analysis Model

The pre-trained BERT model was chosen for sentiment extraction due to its effectiveness in handling short and informal text like tweets.

* 1. Time Series Forecasting Models

We chose ARIMA for its simplicity and LSTM for its ability to capture complex temporal abstracture in the data.

1. COMPARATIVE ANALYSIS OF DATABASES

In this section, in order to compare database systems, we deployed the Yahoo Cloud Serving Benchmark tool and then we recorded metrics such as throughput and latency for MongoDB and moreover Cassandra. Our analysis in comparison of databases revealed that MongoDB showed a better performance in read-heavy scenarios, while Cassandra especially excelled in intensive workloads.

1. CHANGE IN SENTIMENT ANALYSIS

Here, we have a forecast of 1week, 1 month and 3 months, then analyzing sentiment changes over time horizon. An important throughput aspect is observed fluctuations because of real-world events, such as holidays, global events, natural disasters. The sentiment analysis of ‘ProjectTweets.csv’ indicated that there is a huge impact of these events on user sentiments.

1. SENTIMENT FORECASTING

As we clarified in the above section, for forecasting, we chose ARIMA and LSTM. Our models predicted sentiment scores for the next week, month, and three months. These forecasts could be applied based on historical sentiment trends and showed robust accuracy.

1. PRESENTATION OF RESULTS

After processing and analyzing twitter data, we have performed forecasting and in this section, finally we will present our results through visualizations on a dynamic dashboard. The dashboard exhibited sentiment trends, forecasts, and comparisons between the real or actual and forecasted sentiment scores, providing users with an interactive experience. Briefly, our project has involved extensive data processing and storage activities, sentiment analysis utilizing pre-trained models, time series forecasting deploying ARIMA and LSTM models, afterwards comparative database analysis, and a dynamic dashboard presentation of results. Our selection were driven by the need for efficiency, accuracy, leading to insightful sentiment trends and forecasts.

1. CONCLUSION

In the course of this project, we embarked on a multifaceted exploration of sentiment analysis and time series forecasting using a Twitter dataset comprising 1.6 million tweets. By carefully preparing and transforming raw data, we harnessed the power of distributed computing environments and employed Hadoop MapReduce for efficient processing. The process of extracting sentiment from the tweet texts was made easier by utilizing a pre-trained sentiment analysis model, which in turn enabled a granular categorization of sentiments as positive, negative, or neutral.

Our project also addressed time series forecasting, which is an crucial factor in predicting future sentiment trends. Using two distinct approaches, ARIMA and LSTM, we can now model historical sentiment patterns and predict sentiment for the next week, the next month, and the next three months. These models, designed and implemented using Python libraries, showcased our commitment to accuracy and predictive capability.

In accordance with our objective to present the outcomes in an interactive and insightful manner, a dynamic dashboard was created. This dashboard included sentiment trends over a period of time, displayed forecast sentiment scores and provided a convenient comparison between actual and predicted values. Additionally, we conducted a comparative analysis of two prominent NoSQL databases, MongoDB and Cassandra, employing benchmarking tools to gauge their performance under different workloads.

This project outlined the dynamic relationship between data analysis, sentiment analysis, forecasting techniques, database performance analysis, and data visualization. Our rationale-driven choices, coupled with meticulous execution, culminated in a project that

not only unraveled the emotions hidden in our data, but also cast foresight into the mood landscape of the future.

visionary glimpse into the sentiment landscape of the future. This journey enabled us to bridge the gap between the complex world of data and the valuable insights that lie within, demonstrating the tremendous potential of data-informed decision-making in the era of information overload.