Big Data for COVID-19 Insights: From Analysis to Visualization

Team Members:
Emily Nguyen, Riddhi Athreya
Khoury College of Computer Sciences
Data Science Program
nguyen.emily@northeastern.edu, athreya.ri@northeastern.edu

December 10, 2024

Contents

1 Introduction

The COVID-19 pandemic highlighted the importance of data in understanding and managing global health crises, but the sheer volume of data generated posed challenges in storage, processing, and interpretation. Our project, COVID-19 Data Insights: From Analysis to Visualization, addresses these challenges by leveraging Big Data technologies—Apache Hadoop for scalable storage, Apache Spark for rapid data processing, and Apache Hive for structured querying—to build a robust analytics platform. With a focus on ensuring data accuracy and creating dynamic, interactive visualizations, the platform provides actionable insights into COVID-19 trends across regions and over time. Designed for scalability and efficiency, this project serves as a prototype for utilizing Big Data to address real-world challenges.

2 Literature Review

Existing research on COVID-19 data analysis has highlighted the use of platforms such as Dashboard and COVID-19 Maps to track and visualize pandemic statistics. These tools effectively communicate key metrics during the pandemic's peak. However, many of these platforms have since ceased updates. While prior studies demonstrated the potential of Big Data technologies for healthcare analytics, there remains a gap in utilizing these tools to create interactive and dynamic dashboards that provide actionable insights. Our project leverages technologies like Hadoop, Spark, and Hive to build a scalable platform with advanced visualizations, including correlation heat maps and pie charts, to uncover deeper trends and relationships within COVID-19 data.

3 Methodology

The project employs a structured approach to data collection, preprocessing, and analysis, leveraging Big Data technologies and advanced visualization tools to derive actionable insights from COVID-19 data. collection, preprocessing, and analysis.

3.1 Data Collection

Data is gathered from two primary sources:

- Daily CSV Files: Historical COVID-19 data spanning the past three years is sourced from publicly available datasets. Each CSV file contains columns such as Country_Region, Last_Update, Lat, Long_, Confirmed, Deaths, Recovered, Active, Combined_Key, Incident_Rate, and Case_Fatality_Ratio.
  - Link
- News API: COVID-19-related news is fetched in JSON format.
  - Link

3.2 Data Preprocessing

Raw data is cleaned and transformed to ensure consistency, accuracy, and usability:

- Handling Missing Data: Numerical fields with missing values are filled with 0, while rows with empty or incorrect categorical fields (e.g., country names) are either corrected or removed.
- Removing Duplicates: Repeated entries for the same country and date are identified and dropped to maintain data integrity.
- Standardizing Formats: Dates are converted to a uniform format, and country names are standardized (e.g., "United States" vs. "US") using a reference list to ensure consistency across the dataset.

3.3 Analysis Techniques

1. The processed data is stored in Apache Hive tables, enabling SQL-like querying and structured analytics. Hive, built on the Hadoop framework, ensures scalability and efficient access to large datasets.
2. The data is analyzed using Apache Spark, which enables fast and efficient processing of large datasets.
3. The processed data is visualized on an interactive dashboard, developed using:
    - Panel Library: The FastListTemplate is used to create a user-friendly layout, organizing dashboard components and widgets intuitively.
    - Plotly Library: Interactive visualizations, such as line charts, heatmaps, and pie charts, are created to present data dynamically, allowing users to explore trends, relationships, and distributions.

4 Results

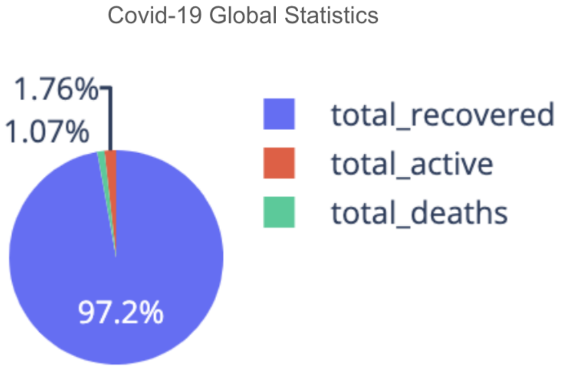

Figure 1: COVID-19 Global Statistics Pie Chart

Recoveries dominate at 97.2%, reflecting the significant recovery efforts worldwide. Active cases account for 1.76%, while deaths represent 1.07% percent, highlighting the pandemic's ongoing impact alongside the large number of recoveries.

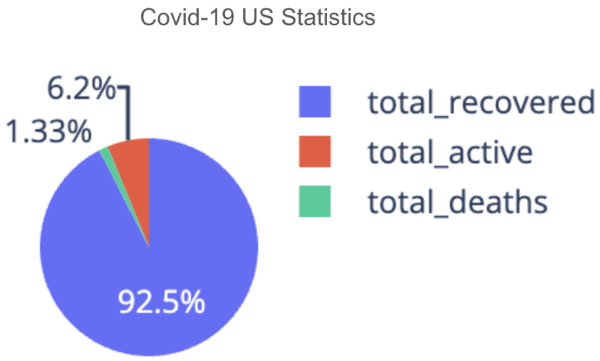

Figure 2: COVID-19 US Statistics Pie Chart

This pie chart illustrates the pandemic's impact within the US country. The majority, 92.5%, are recoveries, showcasing the effectiveness of treatment efforts. Active cases make up 6.2%, while deaths account for 1.33%, emphasizing both the challenges and progress in managing the pandemic domestically.

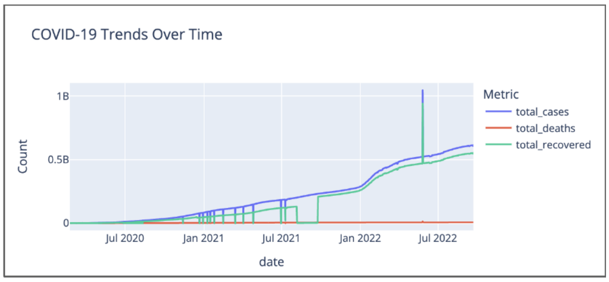

Figure 3: COVID-19 Metrics over Time

These trend lines shows COVID-19 trends over time, tracking total cases, recoveries, and deaths. The blue line reflects the steady rise in total cases, while the green line mirrors this growth, highlighting recovery progress. The red line, representing deaths, remains significantly lower but continues to trend upward.

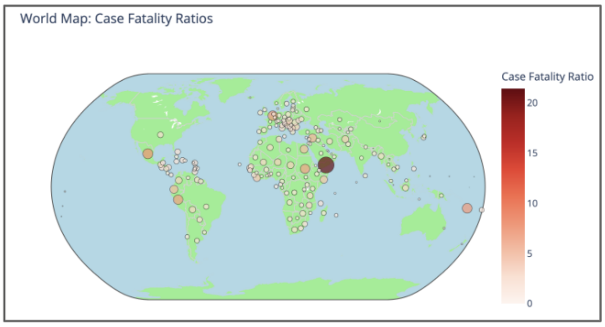

Figure 4: Case Fatality Ratios Globally

This map shows global COVID-19 case fatality ratios, with darker red areas indicating higher fatality rates. The dark red point which stands out is Yemen, which seems to be the country that was most affected. The map highlights significant global variations, reflecting disparities in healthcare and pandemic response.

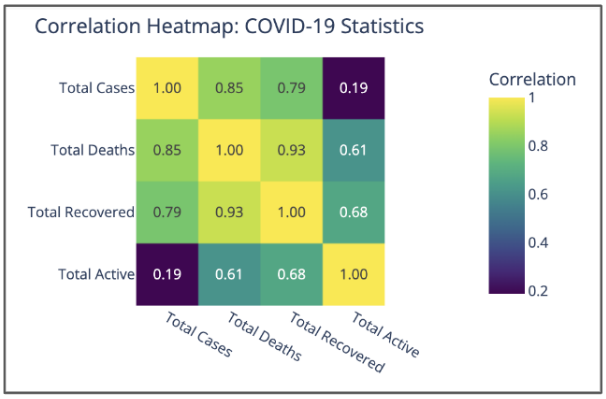

Figure 5: Correlation Heat Map of COVID-19 Statistics

In this heat map, we can see that when total cases increase, deaths rise in tandem with a strong correlation of 0.85. Interestingly, active cases don't follow this pattern, showing a much weaker relationship with total cases at just 0.19. This tells us that while deaths and recoveries tend to move together with total cases, the number of active cases at any given time seems to follow its own independent pattern.

5 Discussion

The results of this project highlight the effectiveness of data-driven tools in understanding and addressing global health crises like COVID-19. Globally, recoveries dominate at 97.2%, while deaths represent 1.07% and active cases account for 1.76%, showcasing both the progress and ongoing challenges in managing the pandemic. Geospatial analysis reveals stark disparities, with countries like Yemen showing significantly higher fatality rates, reflecting gaps in healthcare infrastructure. Correlation analysis indicates a strong relationship between total cases and deaths (0.85) but a weaker correlation with active cases (0.19), suggesting distinct influencing factors. These findings emphasize the importance of prioritizing healthcare resources, such as vaccines and medical supplies, in regions with high fatality rates, while also showcasing the potential of interactive dashboards to inform policy and healthcare planning.

The findings confirm trends highlighted in prior research, such as the strong correlation between total cases and deaths and the disparities in healthcare across regions. This project emphasizes recoveries (97.2%) and provides deeper insights through interactive and dynamic visualizations. Discrepancies, including under-representation of certain regions due to inconsistent reporting, underscore the need for standardized global data collection to improve analysis and decision-making.

6 Conclusion

The project demonstrates the effectiveness of Big Data technologies in analyzing and visualizing COVID-19 data, offering actionable insights through an interactive dashboard. The ability to process large datasets and present complex information in a user-friendly format enables policymakers and researchers to better understand trends and disparities across regions. This approach highlights the importance of leveraging data-driven tools for addressing global health crises and guiding strategic healthcare planning.

Despite its strengths, the project has limitations. Regional differences in testing and reporting standards can affect the consistency of data analysis, and some countries or time periods may be over- or under-represented. Sparse data from certain regions also creates gaps, limiting the ability to draw comprehensive conclusions. Future research should focus on integrating predictive modeling, enhancing real-time monitoring, incorporating geospatial data, and developing bias mitigation strategies to expand the platform's capabilities and apply it to other global challenges.

7 References

- https://www.mass.gov/info-details/covid-19-reporting
- https://data.who.int/dashboards/covid19/cases
- https://coronavirus.jhu.edu/map.html
- https://www.worldometers.info/coronavirus/coronavirus-death-rate/

A Appendix A: Code

Relevant code used in the project:

```python
# ------------------------------------------------------------
def create_session_hive():
    # Initialize Spark session with Hive support
    spark = (
        SparkSession.builder
        .appName('Athena')
        .config('spark.sql.catalogImplementation', 'hive')
        .config('spark.sql.hive.metastore.uris', HIVE_METASTORE_URI)  # Metastore URI
        .config('spark.sql.warehouse.dir', '/user/hive/warehouse')  # HDFS or local path
        .enableHiveSupport()
        .getOrCreate()
    )
    return spark

# ------------------------------------------------------------



    # Read CovidData table from Hive
    covid_data_df = spark.sql("SELECT * FROM database_name.CovidData")

    # Calculate Total Cases, Deaths, Recoveries, and Active Cases
    total_cases = covid_data_df.select(
        sum("Confirmed").alias("Total_Confirmed"),
        sum("Deaths").alias("Total_Deaths"),
        sum("Recovered").alias("Total_Recovered"),
        sum("Active").alias("Total_Active")
    )
```

```
# ---------------------------------------------------------------


    # Read CovidData table from Hive
    covid_data_df = spark.sql("SELECT * FROM database_name.CovidData")

    # Calculate Total Cases, Deaths, Recoveries, and Active Cases by Country
    total_cases_by_country = (
        covid_data_df
        .groupBy("country_id")  # Assuming 'country_id' corresponds to countries
        .agg(
            sum("Confirmed").alias("Total_Confirmed"),
            sum("Deaths").alias("Total_Deaths"),
            sum("Recovered").alias("Total_Recovered"),
            sum("Active").alias("Total_Active")
        )
        .orderBy("Total_Confirmed", ascending=False)
    )
# -----------------------------------------------------------------


    # Read CovidData table from Hive
    covid_data_df = spark.sql("SELECT * FROM database_name.CovidData")

    # Calculate average case fatality ratio by country_id
    cfr_by_country = (
        covid_data_df
        .groupBy("country_id")
        .agg(avg("Case_Fatality_Ratio").alias("Avg_Case_Fatality_Ratio"))
        .orderBy("Avg_Case_Fatality_Ratio", ascending=False)
    )
# -------------------------------------------------------------------
    # Read Country table to map country_id to country name, latitude, and longitude
    country_df = spark.sql("SELECT * FROM database_name.Country")
    country_map = {
        row["id"]: {
            "name": row["Name"],
            "latitude": row["latitide"],
            "longitude": row["longitude"]
        }
        for row in country_df.collect()
    }
# -------------------------------------------------------------------

    # Read CovidData table from Hive
    covid_data_df = spark.sql("SELECT * FROM database_name.CovidData")

    # Convert 'Last_Update' to date type and aggregate total cases over time
    total_cases_over_time = (
        covid_data_df
        .withColumn("date")
        .groupBy("date")
        .agg(
            sum("Confirmed").alias("Total_Confirmed"),
            sum("Deaths").alias("Total_Deaths"),
            sum("Recovered").alias("Total_Recovered")
        )
        .orderBy("date")
    )
# -------------------------------------------------------------------
```

Listing 1: Data Agregation and Analysis

```
import requests
from config import BASE_URL

# Access total cases endpoint
response = requests.get(f"{BASE_URL}/total_cases")
global_json = response.json()

# Access total cases by country endpoint
response = requests.get(f"{BASE_URL}/total_cases_by_country")
country_totals_json = response.json()

# Access case fatality ratio endpoint
response = requests.get(f"{BASE_URL}/case_fatality_ratio")
world_map_json = response.json()

# Access total cases over time endpoint
response = requests.get(f"{BASE_URL}/total_cases_over_time")
trends_json = response.json()

# Access news endpoint
response = requests.get(f"{BASE_URL}/news")
news_data = response.json()
```

Listing 2: Data Retrieval