RAJALAKSHMI
ENGINEERING
COLLEGE

Department of Artificial Intelligence and Data Science

Epidemic spread Monitoring Dashboard

Professor
Dr. Suresh Kumar S M.E., Ph.D.,

S Tanisha (231801178)
S Sushmitha (231801176)
R Roshini (231801140)

Abstract

In recent years, epidemic outbreaks have underscored the urgent need for real-time monitoring, data visualization, and informed decision-making to control disease spread effectively. This project introduces an interactive web-based epidemic monitoring dashboard designed to transform static CSV data into dynamic, actionable insights.

The system leverages Streamlit for building an intuitive front end and pandas for efficient data manipulation. Plotly Express powers the interactive charts and hotspot maps, enabling users to filter and visualize epidemic trends by region, disease, and time period. Integration with DuckDB allows users to perform on-the-fly SQL queries for advanced analysis. Additionally, a live data simulation module mimics real-time updates, demonstrating the system's potential for live epidemic surveillance.

This solution is scalable, adaptable, and user-friendly, making it suitable for health agencies, researchers, and policymakers to monitor outbreaks, identify critical regions, and enable data-driven decision-making for public health preparedness.

Introduction

Epidemic outbreaks pose significant challenges to global health systems, demanding **timely data analysis**, real-time monitoring, and **strategic decision-making** to minimize their impact. Traditional data analysis methods often rely on static reports, making it difficult to track disease progression dynamically. To address this gap, this project introduces a **real-time epidemic monitoring dashboard** that converts raw CSV data into **interactive and insightful visual representations.**

Built using **Streamlit**, the platform offers an intuitive interface where users can filter data by **region**, **disease type**, and **time period**. Visualizations powered by Plotly Express highlight key trends, hotspots, and case distributions. A built-in **SQL query panel**, backed by **DuckDB**, enhances analytical flexibility for users with advanced data needs. Furthermore, a live data streaming simulation demonstrates how the system can be extended to handle **real-time epidemic feeds,** making it ideal for health authorities, researchers, and policymakers to **respond faster and more effectively.**

Probelm Statement And Motivation

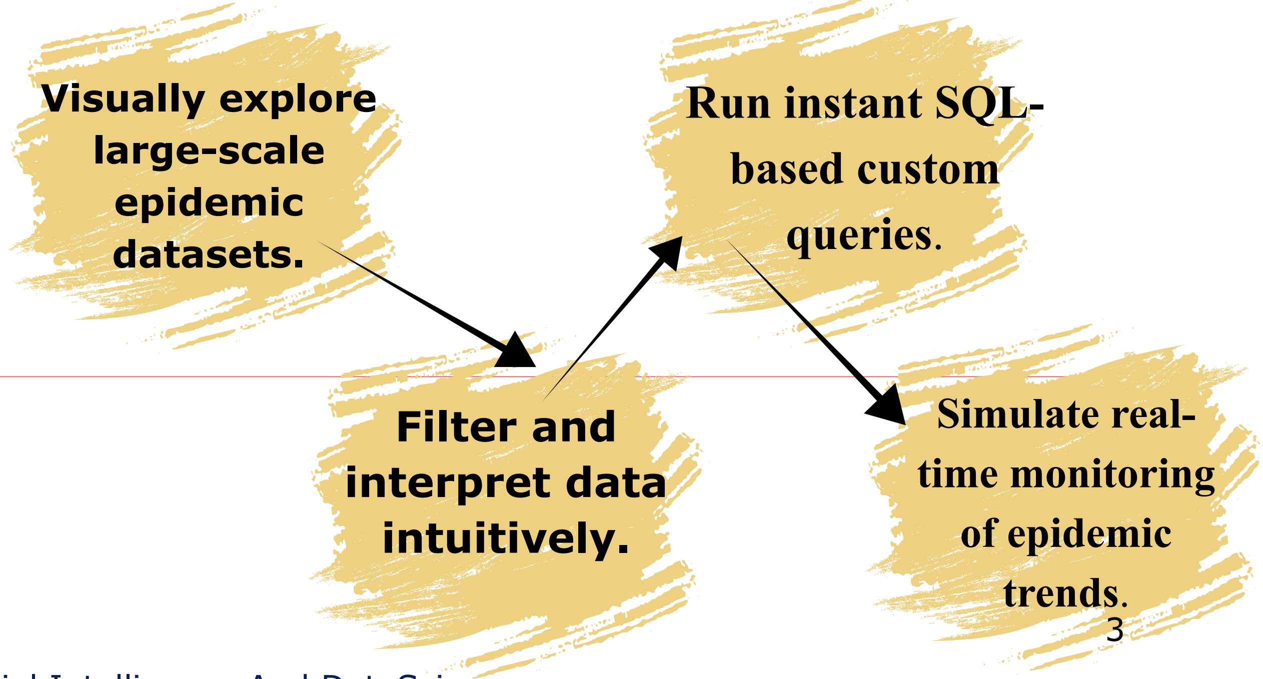
Problem Statement

Monitoring epidemic outbreaks and their progression across regions is often difficult due to fragmented data sources, static reports, and **delays** in analysis. Public health researchers, policymakers, and data analysts **struggle** to quickly identify patterns, track disease spread, and make informed decisions in real time.

There is a need for an **interactive, unified dashboard** that enables **efficient visualization, querying**, and dynamic analysis of epidemic data from a single source.

Motivation Of PS

The motivation behind this project is to **bridge the gap between raw epidemic data** and **actionable insights.** Traditional data reports lack interactivity and real-time exploration features, making it challenging to detect emerging hotspots or understand disease dynamics quickly.



Visually explore large-scale epidemic datasets.

Filter and interpret data intuitively.

Run instant SQL-based custom queries.

Simulate real-time monitoring of epidemic trends.

Existing Systems

Existing System

Most public health dashboards before 2020 were static. They displayed epidemic statistics in simple charts or tables but required manual from health authorities. Systems such as WHO’s dashboard or national COVID dashboards provided valuable insights but lacked local-level drill-down capabilities.

Traditional System Used

- Excel or Power BI for static data visualization.
- Relational databases like MySQL for storage.
- Manual refresh cycles, typically once per day.

Limitations on Existing System

- **Static Data Updates:** Users couldn’t view the latest numbers
- **Poor Interactivity:** No dynamic filtering by region or disease.
- **Limited Visual Range**: Maps were often fixed, lacking customization.
- **High Infrastructure Cost:** Real-tim dashboards required complex server

Proposed Improvement

- A **simulated streaming engine** for live updates.
- · **Plotly visualizations** for zoomable, interactive charts.
- **Integrated DuckDB** for instant SQL querying.
- A **lightweight Python-based** dash board deployable on any machine.

Modules And Description

1 Dashboard UI Module

Database Handling Module

3 Data Streaming Module

Launcher Module

5 Bigdata Processing Module

Dataset (Project)

7 Database file

This is the **main user interface of the project**. It loads epidemic data, applies filters, and generates interactive visualizations. It also connects with the SQL query panel and live streaming module.

This module **manages data storage and querying**. It uses DuckDB to run custom SQL queries for advanced analytics.

This **module simulates real-time streaming** by sending data in intervals, showing how live epidemic data can be handled.

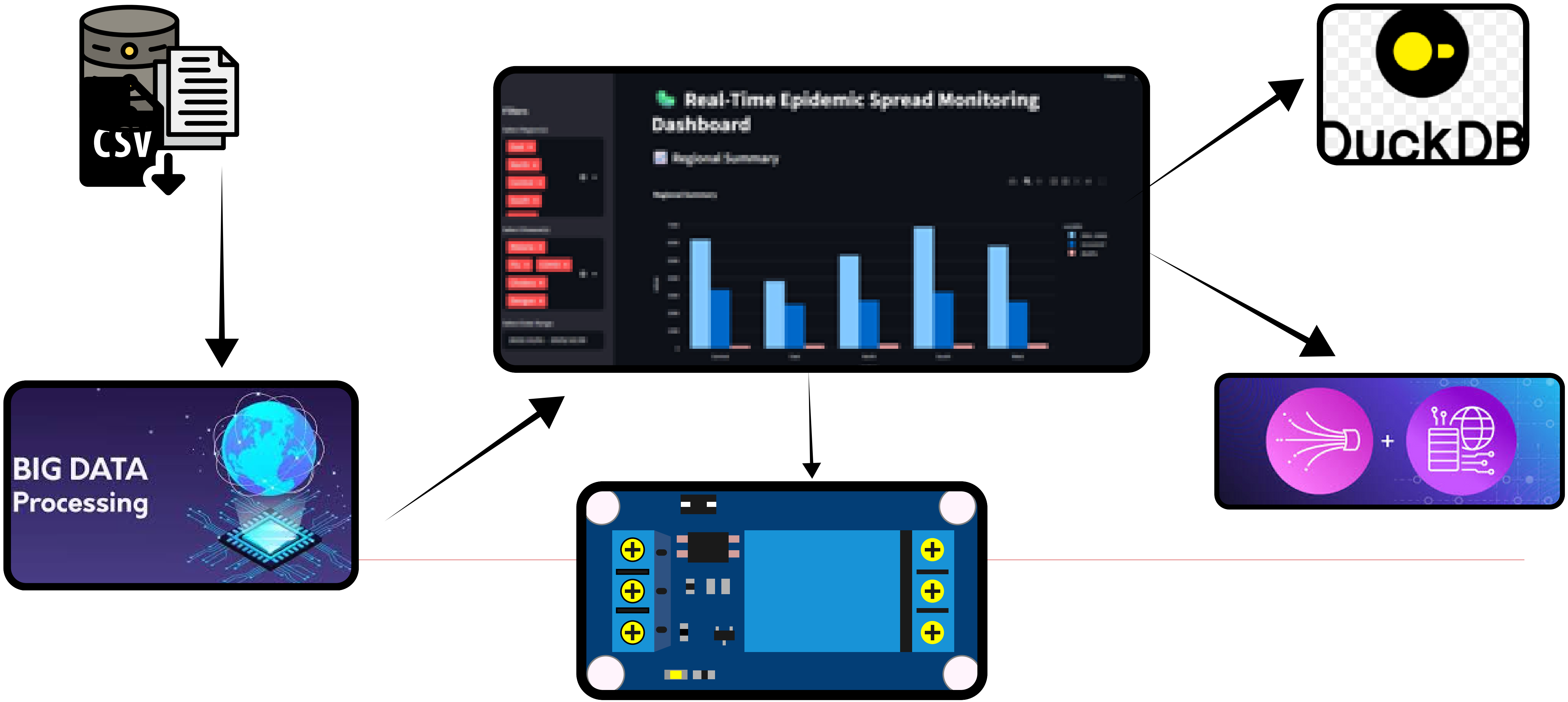
This module **preprocesses large CSV files with Spark**, transforming and scaling data before converting it to Pandas for visualization.

This **script automates launching the Streamlit** dashboard with a single command using a subprocess call.

This **file stores epidemic case** data with key fields like region, disease, cases, and dates, serving as the main data source for all modules.

A **persistent database file that stores the registered dataset**, allowing fast SQL query execution without reloading the CSV file

Architecture Diagram



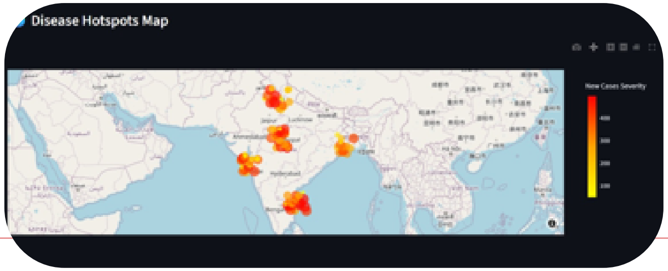
Result And Conclusion

Result

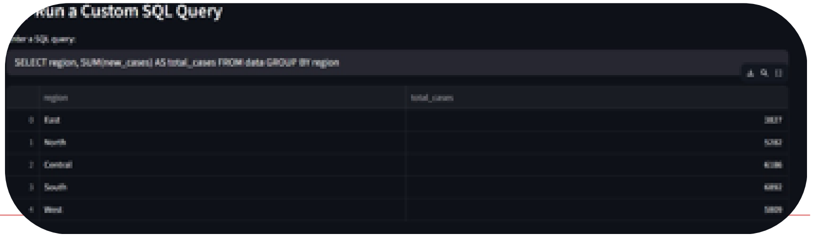
- Provides **instant visual feedback** when filters are applied.
- Displays **real-time charts** and map updates.
- Runs SQL queries smoothly **without lag**.
- Simulates **live epidemic data** streaming.
- Handles large datasets **quickly** with a clean UI

Conclusion

- Demonstrates a **Big Data–driven solution** for epidemic monitoring.
- Enables **interactive visualization**, **filtering**, and **analysis**.
- Integrates **Streamlit**, **DuckDB**, **Pandas**, and Plotly for
- efficiency.
- Proves **open-source tools** can replace heavy platforms.
- Useful for **health organizations,** researchers, and student


Disease Hotspots Map

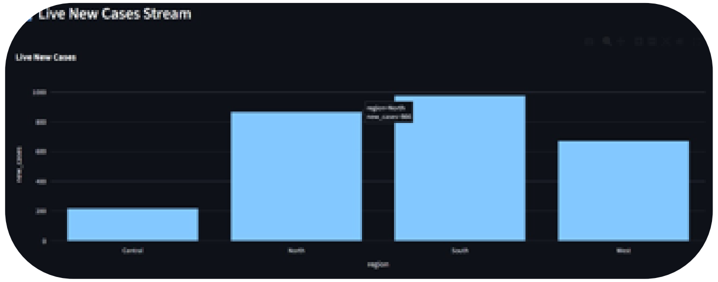
OUTPUT

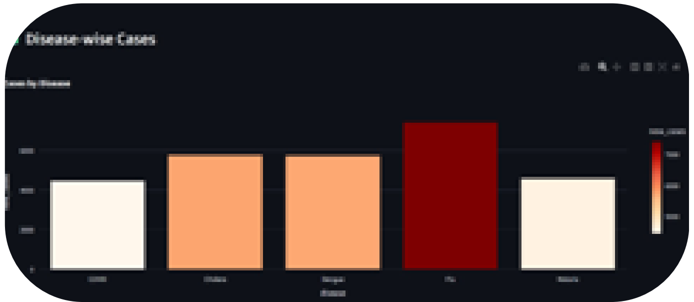

Run a Custom SQL Query

nter a SQL query:

```
SELECT region, SUM(new_cases) AS total_cases FROM data GROUP BY region
```

	region	total_cases
0	East	3827
1	North	5282
2	Central	6136
3	South	6892
4	West	5809


Live New Cases Stream


Disease-wise Cases

Thank You