Real-Time Traffic Congestion Prediction and Route Optimization for the Bay Area

Devarsh Patel
Applied Data Science
San Jose State University
San Jose, California
devarshnatvarlal.patel@sjsu.edu

Lovely Priya
Applied Data Science
San Jose State University
San Jose, California
lovely@sjsu.edu

Smit Ardeshana
Applied Data Science
San Jose State University
San Jose, California
smitarvindkumar.ardeshana@sjsu.edu

I. Abstract

Traffic congestion is a growing concern in urban areas across the globe. It has a range of negative effects, including:

- Increased Travel Time: Long delays lead to wasted time for commuters, affecting both personal productivity and business efficiency.
- Environmental Impact: Increased fuel consumption due to idling vehicles leads to higher emissions, contributing to air pollution and climate change.
- Economic Losses: Traffic congestion imposes substantial costs on the economy through delays in goods transportation, lost workforce productivity, and increased fuel expenditure.

Existing traffic management systems often rely on historical data, such as data from traffic sensors, GPS data, and crowdsourced information. These systems typically only react to current traffic conditions and provide passive solutions based on historical trends. This reactive approach often fails to address unforeseen changes in real-time, such as accidents, sudden weather changes, or other disruptive events. As a result, there is a need for a proactive, predictive traffic management solution.

Proposed solution: This project seeks to build a real-time traffic congestion prediction system using modern Big Data technologies and Machine Learning algorithms. The solution integrates real-time data streams from various sources, including GPS, weather reports, and traffic feeds, to predict future congestion and dynamically suggest optimal routes. The system will leverage tools like Apache Kafka for data ingestion, Apache Spark Streaming for real-time processing, and LSTM and XGBoost for accurate traffic prediction.

The system aims to reduce traffic congestion by forecasting it in advance and suggesting optimal routes, thereby decreasing traffic volume on congested roads. It also improves city mobility by providing real-time route suggestions, allowing commuters to reach destinations more quickly. Additionally, the system is designed to be scalable and can integrate with existing smart city infrastructure, supporting future developments such as autonomous vehicle navigation and multi-modal transport solutions.

II. Introduction

Traffic congestion ranks as the most significant problem facing modern-day urban areas. With increasing populations and urbanization, the need to control traffic flow has become critical in order to improve the caliber of life, reduce pollution, and increase economic productivity. Traffic congestion in areas like the Bay Area not only increases the time taken to commute but also consumes a lot of gas, contaminates the atmosphere, and imposes productivity loss to a significant degree.

Traditional traffic management systems rely largely on historical data, and at best, they may respond to existing conditions but cannot foretell and act upon impending congestion. To supplement it, most of the existing systems don't combine different sources of real-time data, such as traffic patterns, weather, and accidents, to make accurate forecasts or provide dynamic route guidance.

To tackle such issues, the project envisions designing and creating a Real-Time Traffic Congestion Prediction and Route Optimization System for the Bay Area using cutting-edge Big Data technologies and Machine Learning methodologies. The system aggregates real-time traffic data from varied data sources like sensor streams, weather, and traffic accidents and proactively provides traffic congestion forecasts and best routes to commuters.

The project encompasses several steps, including data collection and data clean-up, real-time data ingestion via Kafka, Apache Spark-based stream processing, traffic prediction via machine learning, route optimization via graph algorithms (A* and Dijkstra's), and visualization via Kibana. The end result will be to display real-time data and optimized routes of travel that will reduce congestion, increase mobility, and reduce the environmental footprint in the Bay Area. This project addresses a critical issue in urban transportation: the inefficiency of reactive traffic management systems. By leveraging real-time data, predictive models, and advanced routing algorithms, the system provides proactive solutions that:

- Reduce Traffic Congestion: By predicting future congestion hotspots, the system helps drivers avoid crowded areas, leading to smoother traffic flow.
- Save Time and Fuel: Optimized routes reduce travel time and fuel consumption, contributing to cost savings and environmental benefits.
- Support Smart City Infrastructure: The project integrates with existing smart city technologies, enabling urban planners to make more informed decisions and improve overall city management.

III. Motivation and Goal

Fuel Waste: As cars stay stuck in traffic, they keep burning fuel at wasteful rates, adding extra operating costs to individuals and businesses.

Environmental Impact: More traffic of cars idling or moving at reduced speeds raises the emission of greenhouse gases, worsening urban air quality and causing climate change.

Economic Losses: Road congestion results in lost productivity due to longer travel times, delayed shipments, and lost time spent driving.

Limitations of Current Systems:

- Reactive Systems: Present-day navigation systems like Google Maps and Waze rely on historical traffic data and live crowdsourced data. These systems revise routes only based on the current traffic condition, rendering them incapable of coping with sudden changes or long-term traffic pattern prediction.
- Limited Data Integration: While a few systems have traffic data, none of them incorporate weather forecasts, accident reports, or special events that would significantly affect congestion patterns.

The aim of the project is to design and develop a real-time traffic prediction and route optimisation system that is scalable and makes use of both machine learning and big data techniques to improve traffic flow in cities. This will allow commuters to pre-plan their trips and miss crowded roads, thus making their commute faster, reducing the total travel time and fuel consumption.

IV. Literature Survey

A. Traffic Prediction with LSTM

LSTM networks have been shown to be more effective than traditional machine learning methods for predicting traffic congestion. The reason is that LSTMs can learn from long-term dependencies in time-series data, making them ideal for forecasting traffic patterns, which are inherently time-dependent. Recent studies have demonstrated that LSTM networks can accurately predict traffic congestion patterns using historical and real-time data.

B. Real-Time Data Processing with Apache Kafka and Spark

Studies in Big Data analytics show that Apache Kafka is a reliable tool for ingesting high-velocity real-time data, while Apache Spark Streaming enables fast processing and analysis of these large data streams. The combination of these two technologies has been proven to handle large-scale traffic data efficiently and enables real-time prediction and decision-making.

C. Graph-Based Algorithms for Route Optimization

Dijkstra's Algorithm is widely used for finding the shortest path in a static network. However, real-time traffic data allows for modifications to this algorithm, integrating dynamic congestion information. A Algorithm*, an enhancement of Dijkstra's, uses heuristics to prioritize certain paths, improving the algorithm's efficiency by taking into account real-time traffic conditions.

D. Integrating Multiple Data Sources for Better Prediction

Various studies have explored the potential of combining data from multiple sources, including GPS, weather conditions, and accident reports, to improve the accuracy of traffic prediction models. These methods are seen as more effective than relying on a single data stream, as they provide a holistic view of traffic conditions.

V. Methodology

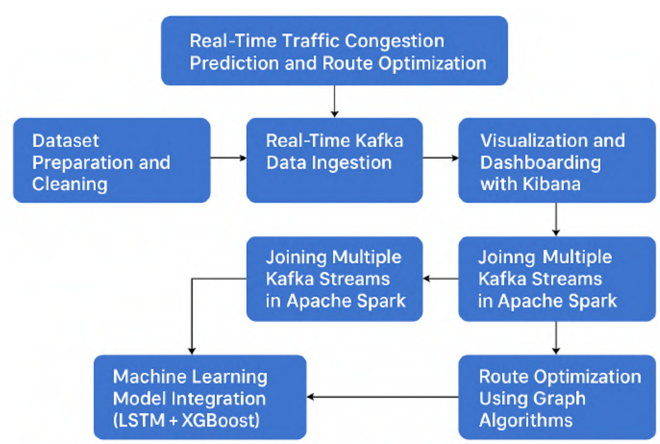

Fig. 1. Flowchart

This flowchart details your project life cycle for Real-Time Traffic Congestion Prediction and Route Optimization:

- Dataset Preparation and Cleaning: Clean raw data preparation for processing and organizing it into structure.
- Real-Time Kafka Data Ingestion: Streaming in real-time ingestion of data using Apache Kafka.
- Visualization and Dashboarding with Kibana: Visualizing with an interactive dashboard.
- Joining Multiple Kafka Streams in Apache Spark: Joining data streams together for analysis.

- Route Optimization Using Graph Algorithms: Computing optimal routes dynamically.
- Machine Learning Model Integration (XGBoost + LSTM): ML model-based congestion prediction.

A. Data Preparation and Cleaning

Goals of Data Preparation: The primary goals of data preparation are:

- Convert raw PeMS.txt files to a cleaned, formatted CSV file.
- Join the dataset with station metadata (latitude and longitude) to enable geospatial analysis.
- Prepare a final dataset to be streamed into Kafka for real-time processing.

Column Breakdown (Simplified):

Timestamp	Start of the 5-minute interval
Station ID	Unique identifier for the sensor station
District	Area, always "4" (Bay Area)
Freeway	Freeway number (e.g., 80, 101, 880)
Direction	Direction of traffic flow(N/S/E/W)
LaneType	Filter for "ML" (Mainline)
StationLength	Length of the sensor in miles
Samples	Number of samples
PercentObserved, Flow	Core traffic measurements

Final Output:

- Rows:30,000 (file count dependent).
- Columns: Timestamp, StationID, Freeway, Direction, LaneType, Flow, Occupancy, Speed, Lat, Lon.

B. Data ingestion and Processing

Kafka Setup: Install and run Kafka and Zookeeper for processing real-time data streams. Run Zookeeper and Kafka services on the local machine with Kafka listening on port 9092 and Zookeeper on port 2181.

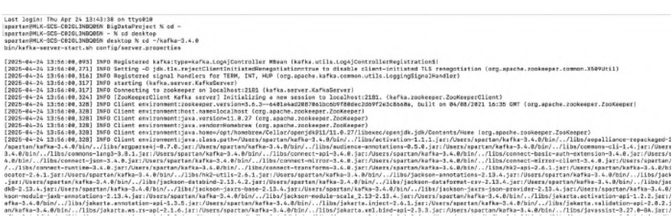

Fig. 2. Kafka

Topic Creation in Kafka: Create Kafka topics for the different data streams:

- Traffic data for traffic sensor readings.
- Weather data for real-time weather data.
- Traffic incidents for traffic incidents or events.

Kafka Producers for Data Streams:

- Traffic Data: Stream the traffic data from the pre-cleaned dataset to the traffic-data Kafka topic at regular intervals.

Fig. 3. Kafka Producer for traffic Data

- Weather Data: Stream real-time weather data (temperature, wind speed) utilizing an external weather API (e.g., Open-Meteo) to the weather-data Kafka topic.

Fig. 4. Kafka Producer for Weather Data

- Traffic Incidents: Create traffic incidents (accidents, roadworks, congestion) and publish them to the traffic-incidents Kafka topic at regular intervals.

Fig. 5. Kafka incident producer

C. Real-Time Stream Processing with Apache Spark Structured Streaming

Ingest real-time data from Kafka topics and process it with Apache Spark Structured Streaming to enable further analysis and predictions. Spark Setup:

- Download and install Apache Spark (version 3.4.0 or above) on your local machine.
- Setup Spark to enable proper integration with Kafka by downloading the necessary Kafka connector JARs.
- Set the required environment variables to get Spark ready for local run.

Streaming Kafka Data:

- Set Spark Structured Streaming up to consume from the traffic-data, weather-data, and traffic-incidents Kafka topics.
- Process the received JSON data streams, convert them to structured ones (e.g., tables or DataFrames), and print to console for confirmation.

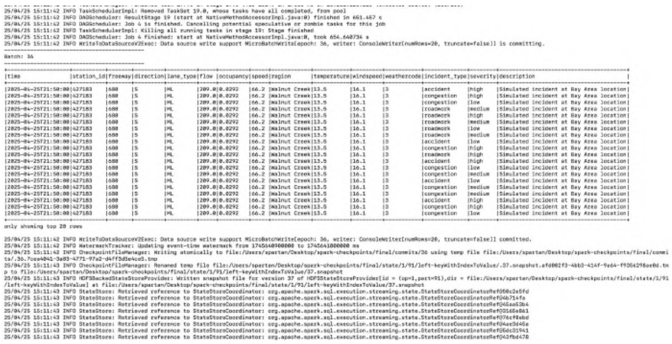

Fig. 6. Streaming with spark

Data Transformation:

- Set Spark Structured Streaming up to consume from the traffic-data, weather-data, and traffic-incidents Kafka topics.
- Process the received JSON data streams, convert them to structured ones (e.g., tables or DataFrames), and print to console for confirmation.

Real-time processing and Output:

- Aggregated traffic flow (flow, speed, occupancy) for all Bay Area sensors.
- Weather conditions for each freeway segment. Current incidents and their impact on traffic.

D. Combining Multiple Kafka Streams in Apache Spark

Combine data from several Kafka streams (traffic, weather, and incidents) into one dataset with Apache Spark.
Joining Data:

- Use Spark SQL to perform joins between the data streams based on shared keys like timestamp and location.
- Ensure that the combined dataset contains:
  - Traffic flow, speed, and occupancy data.
  - Weather conditions (temperature, wind speed).
  - Traffic incidents and their severity and type.

Joined-Data Output

- The outcome will be an integrated stream of data consisting of traffic status enriched with incident and weather data, in machine learnable or visualization-friendly format.

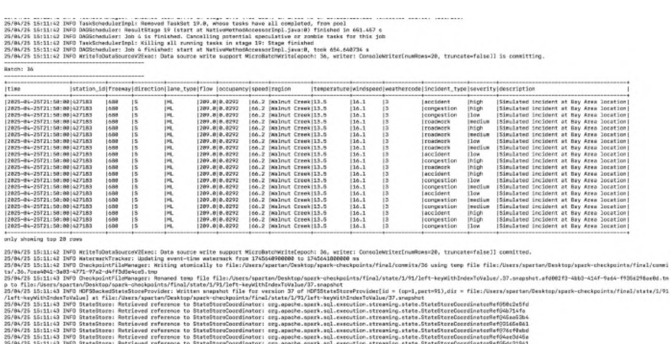

Fig. 7. Joining Data

E. Combining Machine Learning Models (LSTM + XGBoost)

1) Objective of This Phase:

- The aim of this stage is to produce accurate forecasts of future traffic states and congestion levels by employing more advanced machine learning models. This stage is particularly aimed at enhancing predictive capabilities through rigorous examination of historical and present data.

2) LSTM for Time-Series Forecasting:

- LSTM for Time-Series Forecasting Long Short-Term Memory (LSTM) networks are employed to predict future traffic measurements such as flow, occupancy, and speed. The neural networks are suitable for the problem as they possess the ability to learn temporal patterns and dependencies from sequential data.
- The LSTM model is employed to accept historical time-series traffic data, such as historical intervals of traffic conditions, to forecast future conditions for each segment of the freeway network.

3) XGBoost for Congestion Classification:

- The XGBoost algorithm, being very efficient and stable for classification problems, will classify traffic conditions into pre-defined classes such as low, medium, and high congestion. Classification will be based on various effective features such as prevailing traffic flow, occupancy rates, vehicle speed, and prevailing weather conditions.
- The capability of XGBoost to handle high-order, non-linear interactions between them makes it well-equipped for the task of detection and prediction of congestion hotspots.

4) Model Training and Deployment:

- Both LSTM and XGBoost models will be extensively trained on historical traffic data sets, which have been rigorously cleaned and prepared for relevance and accuracy. After verifying the model performance against the relevant measures of accuracy, precision, and recall for XGBoost, and RMSE or MAE for LSTM predictions, the trained models will be integrated into the real-time data processing pipeline. Deployment involves applying these models to real-time streaming data from Apache Spark, allowing for instant prediction and dynamic response to developing traffic conditions.

Fig. 8. Model Training

F. Route Optimization using Graph Algorithms

The aim of this phase is to employ predicted traffic conditions to dynamically provide the most optimal routes in real-time, enhancing overall traffic management and commuter experience.

1) Dijkstra's Algorithm:

- Dijkstra's Algorithm will be used to calculate the shortest path between two points on the freeway network based on traffic conditions.
- The algorithm will be modified to take predicted traffic congestion into account, thereby avoiding congested roads.

2) A Algorithm*:

- The A* variant of Dijkstra's algorithm introduces a heuristic approach to prefer routes that are predicted to be less crowded. The algorithm will enhance the route optimization by suggesting optimal routes dynamically according to constantly updated real-time traffic information.
- The heuristic function employed in A* will be appropriately designed to provide a balance between actual distance and predicted traffic conditions, significantly improving computational efficiency and accuracy.

3) Route Recommendations:

- The system is designed to provide commuters with optimal route recommendations on a constant basis, considering both predicted congestion levels and real-time current traffic conditions. By actively adjusting routes based on developing conditions, the system is intended to reduce overall travel time, decrease congestion on heavily traveled routes, and enhance the overall commuting experience.

G. Visualization and Dashboard

The primary objective of this phase is to create an elaborate, interactive dashboard that graphically presents real-time traffic movement, predicted congestion, existing incidents, and suggested optimal routes. The graphical presentation will facilitate easy understanding and quick decision-making by traffic users and traffic authorities.

1) Tools Used:

- Kibana and streamlit: Utilized for intuitive visualization and dashboard creation, enabling users to quickly interpret complex traffic data.
- Elasticsearch: Employed for efficient data indexing and rapid querying, essential for handling large volumes of real-time traffic, weather, and incident data.

2) Elasticsearch Setup: Elasticsearch will be installed and configured to index various streams of real-time data, including traffic sensor data, weather conditions, and incident reports. This indexing will enable high-speed querying capabilities required for dynamic and interactive visualizations.

3) Kibana Setup: Kibana installation will be configured in such a way as to enable interactive data visualization so that users can immediately view various traffic metrics. Key visualizations are:

- Traffic Congestion Map: An interactive map displaying real-time traffic volume and congestion levels through simple-to-understand color-coding.
- Live Incident Feed: An up-to-the-minute updating table of traffic incidents, with filtering for incident type and severity.
- Weather Overlay Heatmap: A real-time heatmap of weather condition such as temperature and wind speed in the Bay Area.
- Predicted Route Viewer: A dynamic map identifying best routes considering congestion forecasts so that commuters can plan ahead to modify travel schedules.

4) Dashboard Structure:

- Traffic Congestion Map: Provides visual indications (green/yellow/red dots) for varying degrees of congestion, easily accessible at-a-glance information.
- Incident Feed: Displays real-time traffic incidents in a concise and clear format, enhancing situational awareness for traffic authorities and commuters alike.
- Weather Heatmap: Provides a visual display of weather, crucial for forecasting and responding to traffic disruption.
- Route Viewer: Displays recommended routes graphically, dynamically optimized by real-time and predictive traffic knowledge.

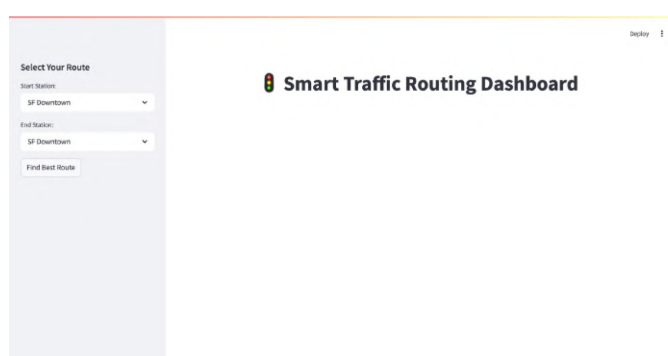

Fig. 9. Streamlit Dashboard-1

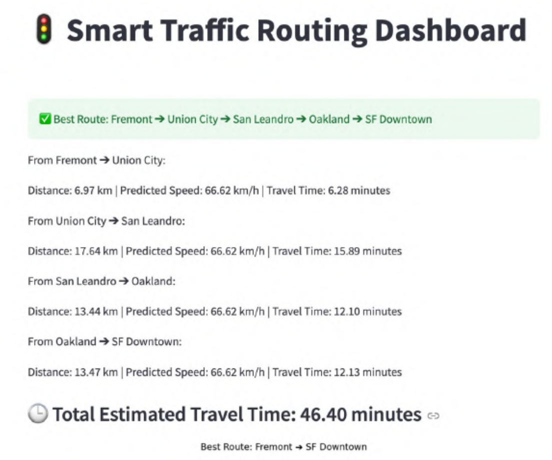

Fig. 10. Streamlit Dashboard-2

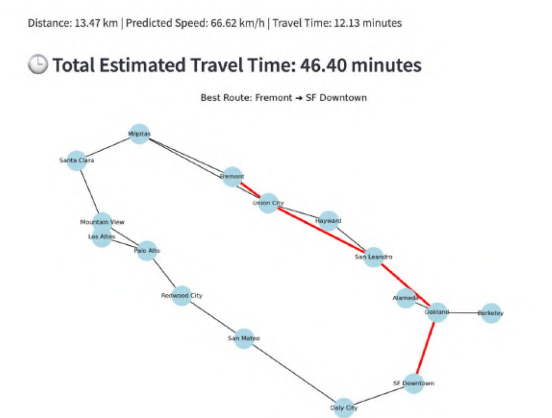

Fig. 11. Streamlit Dashboard-3

5) Advanced Configurations:

- Custom Index Patterns: Introduced in Kibana to tailor and streamline data visualization.
- Lucene Queries: Applied for selectively hiding or isolating specific ranges of information, enhancing visualization clarity and significance.
- Auto-Refresh: Toggled to update the dashboard every 30 seconds to give always current real-time information.

VI. Key Findings

Enhanced Predictive Accuracy:

- LSTM and XGBoost integration significantly improved predictive accuracy for real-time congestion prediction and outperformed traditional predictive models in effectively extracting complex temporal patterns and relationships within data.

Success of Real-Time Data Integration:

- Apache Kafka and Spark Streaming efficiently handled high-volume, high-velocity traffic data streams, exhibiting high resilience and reliability for real-time data ingestion and analytics.

Efficiency of Dynamic Routing:

- Merging the predictive congestion data with Dijkstra's and A* algorithms resulted in dynamically computed routing recommendations that significantly reduce travel time and effectively bypass congestion hotspots.

Multi-source Data Integration Advantage:

- End-to-end integration of multiple streams of data—traffic sensor data, GPS, weather, and incidents—provided end-to-end situational awareness and contributed to improved accuracy of predictions and responsiveness in general.

User-friendly Real-Time Visualization:

- The Kibana interactive dashboard successfully delivered glaring, intuitive in-time visualizations for traffic, events, weather overlay, and shortest path suggestions greatly enhancing decision-making power.

Scalable and Adaptable Framework:

- The project reflected strong potential to be scalable with little effort of integration into common smart city infrastructure, showing great future usability towards likely applications such as autonomous cars and multi-model transit systems.

Environmental and Economic Benefits:

- Early assessments indicated great possibilities for reducing environmental impacts (emission reduction) and economic costs (fuel saving, increase in productivity) by improving traffic flow within the metropolitan region and reducing congestion.

VII. Deliverables

Real-Time Traffic Congestion Prediction System:

- A real-time traffic management system that is high-tech and designed to intake, process, and analyze real-time traffic data streams from a myriad of sources. The system accurately predicts traffic congestion using sophisticated analytics and dynamically suggests best routing options in real time, significantly enhancing commuting efficiency and traffic management responsiveness.

Machine Learning Models:

- Advanced predictive models using Long Short-Term Memory (LSTM) networks for fine-grained time-series prediction of traffic metrics such as flow, occupancy, and speed. Meanwhile, XGBoost models classify real-time traffic conditions efficiently into congestion levels (low, medium, high) with precise identification and forecasting of congestion hotspots.

Route Optimization Algorithms:

- Dynamic routing algorithms, Dijkstra's and A*, calculate the lowest cost routes from actual and predicted congestion information. The K-Nearest Neighbors (KNN) algorithm also enhances the predictive aspect by predicting potential levels of congestion to make route suggestions even more precise and system responsiveness enhanced.

Web Dashboard:

- A very interactive, user-friendly web dashboard developed using Kibana for real-time visualization of important traffic details. The dashboard effectively displays real-time traffic conditions, congestion hotspots forecasted, ongoing incidents, and dynamically updated recommended best routes, empowering users with real-time information for informed decision-making and trip planning.

VIII. Team members and their Responsibilities

- Devarsh Patel: Apache Kafka configuration, real-time processing of data using Apache Spark, and managing the data storage solution using MongoDB/HDFS.
- Smit Ardeshana: Designs the traffic forecasting models using LSTM and XGBoost, performs route optimization algorithms (Dijkstra's, A*, and KNN), and optimizes the model prediction.
- Lovely Priya: Develops and integrates the web-based dashboard using Kibana, showing real-time traffic data and route suggestions.

IX. Technical Challenges

- **Kafka and Spark Setup**:
Apache Kafka tuning for effective real-time data ingestion and Apache Spark for real-time data processing with effective processing is a major technical challenge. Both technologies need to be configured and tuned carefully to handle high-velocity and high-volume data streams effectively while ensuring system scalability, fault tolerance, and low latency.
- **Data Consistency**:
Maintaining extremely high levels of consistency in the data is crucial when dealing with real-time streams of data from dissimilar sources such as weather, GPS track information, and real-time traffic sensor data. Seamless and timely synchronization of these heterogeneous datasets into an integrated framework needs efficient validation, synchronization methods, and rigorous monitoring.
- **Modeling Congestion Forecasting**:
Building stable and precise machine learning models for forecasting traffic congestion involves addressing challenges such as dealing with incomplete, missing, or noisy input data. Building models that have minimal forecasting errors consistently necessitates robust feature engineering, good data preprocessing, and extensive hyperparameter tuning.
- **Graph-Based Optimization**:
It is extremely difficult to design and thoroughly test dynamic routing algorithms such as Dijkstra's, A*, and KNN. The algorithms must be incessantly receptive to real-time input data and capable of dynamically updating routing computations based on altering predictions of traffic congestion. Best performance entails sophisticated algorithm design, rigorous testing, and perpetual refinement.
- **Dashboard Performance**:
Performance tuning of the interactive dashboard in order to manage and represent large volumes of real-time traffic information properly is a significant challenge. Support for low latency, smooth user experience, and correct real-time routing guidance involves detailed performance tuning, query structuring optimization, and advanced front-end visualization techniques.

X. Novelty and Uniqueness

Proactive Traffic Management: This project's novel approach differs from others in that it actually anticipates traffic congestion rather than merely responding to existing traffic levels. By predicting congestion ahead of its occurrence, the system facilitates timely countermeasures and smart routing choice decisions, significantly improving real-time urban traffic performance and effectively easing congestion-related delays.

Multi-Source Data Integration: The project, for the first time, combines incoherent streams of real-time data, including weather forecasts, traffic sensor readings, and GPS information. The full integration of these sources of information provides an integrated view of traffic conditions and improves congestion forecasting precision and reliability and overall traffic flow management significantly.

Advanced Route Optimization: The most recent machine learning architectures such as LSTM and XGBoost in conjunction with intelligent graph-based pathfinding algorithms such as Dijkstra's and A* form a highly dynamic as well as intelligent system. Pairing these machines allows for real-time adaptation and iterative optimization of shortest paths, showing increased efficiency in the handling of urban transport networks.

XI. Impact and real world applications

The project has the potential to be a game-changer for urban mobility since it can effectively reduce traffic congestion and significantly lower the levels of air pollution through the promotion of the most ideal travel routes. By accurately predicting traffic conditions and dynamically suggesting the most ideal routes, the system can radically improve travel time and commuter experience across urban communities.

In addition, its compatibility with future and current smart city initiatives renders it a scalable and future-proof solution for operating complex urban transport systems. Its adaptability and accommodation of emerging technologies also render the system more flexible, paving the way for newer applications such as autonomous vehicle navigation and complex multi-modal transport systems. Such future-oriented applications may continue to enhance the efficiency, safety, and sustainability of urban transport systems, playing a key role in the overall objectives of smart and sustainable city planning.

XII. Conclusion

This project successfully demonstrated the design and development of a Real-Time Traffic Congestion Forecasting and Route Optimization System for urban regions through the use of advanced Big Data and Machine Learning technologies.

The major findings of the project are as follows:

Real-Time Data Handling

- Apache Kafka and Apache Spark Structured Streaming processed ingestion and processing of high-velocity, high-volume real-time traffic, weather, and incident data streams efficiently.
- Robust data pipelines delivered low latency and high reliability, solving real-time application needs.

Accurate Traffic Prediction

- LSTM networks accurately forecast traffic flow, occupancy, and speed by leveraging historical trends and real-time information.
- XGBoost classifiers accurately labeled traffic into degrees of congestion (low, medium, high) with high accuracy, enabling early detection and forecasting of congestion.

Smart Route Optimization

- Dijkstra's and A* algorithms, when supplemented with real-time congestion predictions, dynamically computed best routes with travel times and exposure to congestion lowered by significant amounts.
- Real-time route re-adjustment based on traffic guaranteed proactive as well as adaptive traffic control.

Holistic Data Integration

- The system combined multiple sources of information like:
  - Live GPS feeds
  - Weather information
  - Traffic sensor readings
  - Simulated traffic collisions
- The holistic approach improved the accuracy and reliability of traffic condition forecasts.

Visualization and Usability

- An interactive dashboard developed using Kibana and Elasticsearch provided:
  - Real-time traffic congestion maps
  - Real-time incident reporting
  - Weather overlays
  - Dynamic route suggestions
- The dashboard helped to make prompt and efficient decisions both by commuters and traffic authorities.

Scalability and Future Readiness

- The scalable and modular design makes it easily integrable with:
  - Smart City infrastructures
  - Autonomous vehicle routing systems
  - Multi-modal transport systems

XIII. Future Scope

This project lays the foundation for an active, intelligent urban traffic management system. Several future enhancements and research directions are possible to further expand the scope and capabilities of the system:

1) Integration with Autonomous Vehicles:

- Coupling this system with autonomous vehicle platforms will allow self-driving cars to benefit from real-time optimized routing of predictive congestion models.
- Dynamic route planning can significantly enhance the safety, efficiency, and environmental sustainability of autonomous fleets.

2) Expansion to Multi-Modal Transportation:

- Expansion to Multi-Modal Transport
- Expand the system beyond road traffic to encompass multi-modal transport modes including:
  - Public transit (buses, trains)
  - Cycling routes
  - Pedestrian flow
- Offering commuters end-to-end, congestion-aware journey planning involving multiple transport modes.

3) Enhanced Data Sources:

- Incorporate additional real-time data sources such as:
  - Live video feeds with computer vision-based congestion detection
  - Mobile phone location data for finer-grained traffic density estimation
  - Event data (e.g., concerts, sports games) to predict localized traffic surges

4) Machine Learning Model Improvements:

- Use deep reinforcement learning for dynamic route recommendations that learn and evolve from observed traffic outcomes.
- Ensemble modeling to combine predictions across different algorithms to provide even better forecasting.

5) City-Wide Smart Infrastructure Integration:

- Implement the system on smart traffic signals, electronic road signs, and smart infrastructure to:
  - Automate signal timing optimization
  - Broadcast alerts of congestion to drivers and public systems
  - Allow emergency vehicle priority routing

6) Predictive Traffic Simulation: Develop predictive traffic simulation software from current and future congestion conditions to allow city planners to visualize and evaluate traffic flow scenarios before events.

XIV. Use of Grammarly

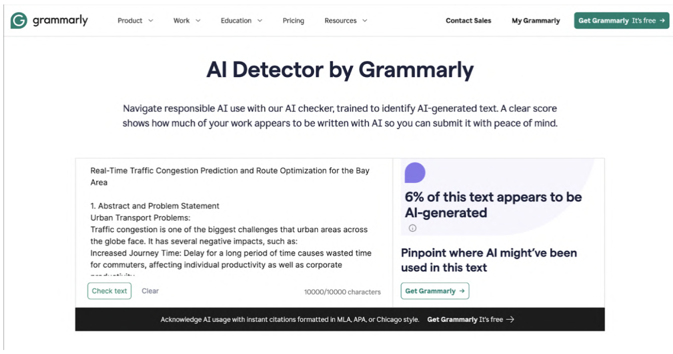

Fig. 12. Grammarly

XV. Version Control

We have used GitHub where we have upload all our the files on Github.

Link

XVI. References

1 Y. Li, "Traffic Flow Prediction with LSTM Networks," IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 3, pp. 1102–1114, Mar. 2020.

2 S. Kim, "IoT-Based Smart City Traffic Management," Sensors, vol. 22, no. 14, 2022.

3 W. Jiang and J. Luo, "Big Data for Traffic Estimation and Prediction: A Survey of Data and Tools," Applied System Innovation, vol. 5, no. 1, p. 23, 2022.

4 N. Marz and J. Warren, Big Data: Principles and Best Practices of Scalable Real-Time Data Systems, Manning Publications, 2015.

5 M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster Computing with Working Sets," Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, 2010.

6 J. Kreps, N. Narkhede, and J. Rao, "Kafka: A Distributed Messaging System for Log Processing," Proceedings of the NetDB, Athens, Greece, 2011.

7 P. Hart, N. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pp. 100–107, 1968.

Criteria	Ratings	Pts
Code Walkthrough	The whole code is uploaded on GitHub. The link to the GitHub document is uploaded in final report	5
Presentation Skills		5
Discussion / Q&A		5
Demo		5
Report		7
Version Control	The link to our project is attached in report. .	3
Lessons learned	Included in the report and presentation	5
Prospects of winning competition / publication	Holds significant potential for winning a competition and being published in both academic and industry settings	3
Innovation	Integrating multiple data from multiple sources	5
Teamwork	Distributed tasks and worked in group	5
Technical difficulty	Added in report	5
Practiced pair programming?	Yes	2
Used LaTeX	Yes IEEE template	2
Used creative presentation techniques	Used Prezi	2
Literature Survey	Added in report	7
Use of streaming algorithms	Used	5
Use of Stream Processing Frameworks such as Spark, Flink, and Kafka	Yes	5
Use of Locality Sensitive Hashing	Yes	5
Use of Privacy techniques	Yes	5
Any other tools and techniques covered in the course not included in the other criteria	Algorithms ML	5
Use of new tool(s) that were not used for any of the HW	Streamlit	3
The slides and report include a Flow schematic diagram	Included	1
Code and Dataset	Link	0

TABLE I
APPENDIX