

# Explainable AI in Traffic Prediction

Han Juin Wong  
Auckland University of Technology  
mkh6113@autuni.ac.nz

**Abstract**—This dissertation presents an end-to-end pipeline that connects traffic prediction, explainability, and policy generation to simulation evaluation for urban traffic management in Auckland CBD. An XGBoost model predicts lane-level hourly volumes, SHAP provides global and local explanations to identify actionable drivers. Using these explanations, a large language model generates SUMO-compatible signal timing policies. Two iterations are studied: Version 1 (local retiming, addition traffic signal) and Version 2 (green-wave). In simulation, Policy-V2 increases total arrivals by +38.2%, eliminates teleports, increases average mean speed by +68.2%, and reduces mean travel time by -25.3% relative to baseline. The study demonstrates that Explainable AI can bridge forecasting and control by turning interpretable features into deployable signal plans. Limitations include data quality and road network scope. The framework suggests a practical path toward auditable, adaptive signal control combining ML prediction, XAI, and LLM-assisted design.

**Index Terms**—XGBoost, Explainable AI, Large Language Models, Policy Generation, SUMO

## I. INTRODUCTION

In recent decades, transportation has evolved rapidly, progressing from wagons and sailing ships to steam engines and modern automobiles. As travel times have decreased, its importance to people worldwide has grown. However, the surge in transportation services now overloads many systems, causing negative effects and undermining the original goal of efficient travel. This makes traffic planning important, it not only accommodates more vehicles but also keeps traffic flowing smoothly. With current technological advances, traffic prediction is now possible, enabling smarter and more efficient traffic management systems.

The aim of an Intelligent Traffic Management System (ITS) is to reduce traffic congestion, enhance road safety, optimize fuel consumption and emissions, and enable real-time traffic management and traveler information [1]. However, the growing complexity of ITS has increased the need for accurate traffic prediction models [2]. These models forecast future traffic states, such as speed, flow, or congestion levels by using historical and real-time data [3]. Reliable traffic prediction enables authorities to optimize signal control, manage congestion, and keep traffic flowing smoothly [4].

There are two types of traffic prediction, short-term and long-term prediction. Short-term traffic prediction focuses on forecasting traffic conditions over a brief period, ranging from a few minutes to an hour. This type of prediction is crucial for real-time traffic management, enabling dynamic adjustments to signal timings, route guidance, and congestion mitigation strategies [5]. In contrast, long-term traffic prediction aims to

forecast traffic patterns over extended periods, such as days, weeks, or even years. Long-term predictions are essential for urban planning, infrastructure development, and policy-making, as they help authorities anticipate future transportation needs and design systems that can accommodate growth [6]. To effectively manage traffic, both short-term and long-term predictions are necessary, as they address different aspects of traffic management and planning, and there are different techniques that can be used to perform traffic prediction.

Traffic prediction models can be broadly categorized into statistical models, Machine Learning (ML) models, and Deep Learning (DL) models [2]. Statistical models, such as Autoregressive Integrated Moving Average (ARIMA), use historical data to identify patterns and forecast future traffic states [7]. ML models, like Extreme Gradient Boosting (XGBoost), can capture nonlinear relationships in traffic data, improving prediction accuracy [8]. DL models, such as Long Short-Term Memory (LSTM), are designed to handle sequential data and can learn complex temporal dependencies in traffic patterns [9]. However, many traffic prediction models operate as "black boxes," providing accurate predictions without explanations. This lack of transparency can hinder trust and adoption by traffic authorities [10].

Explainable Artificial Intelligence (XAI) techniques can help address this issue by providing insights into how models make predictions [11]. Methods like Shapley Additive Explanations (SHAP) and Local Interpretable Model-Agnostic Explanations (LIME) can identify the most influential features contributing to traffic predictions, enhancing model interpretability [12]. By understanding model behavior, traffic authorities can make informed decisions and trust the predictions provided by Artificial Intelligence (AI) systems.

Large Language Model (LLM) have shown promise in generating human-like text and reasoning about complex problems [13]. Feeding appropriate data into an LLM can yield accurate and reliable results. In traffic management, LLMs can potentially translate model outputs into actionable policy recommendations [14]. For example, LLMs could analyze traffic prediction results and suggest signal timing adjustments or route diversions to alleviate congestion [14]. However, the effectiveness of LLM-generated recommendations in real-world traffic scenarios remains underexplored. To validate these LLM generated recommendations, simulation environments like Simulation of Urban Mobility (SUMO) can be employed.

Simulation environments like SUMO provide a platform to evaluate traffic management policies in a controlled set-

ting [15]. By implementing traffic policies in SUMO, policy makers can assess their impact on traffic flow and congestion before real-world deployment. There are many simulation environments that pursue the same objective, such as Verkehr In Städten – SIMulation (PTV microscopic traffic simulator) (VISSIM), Advanced Interactive Microscopic Simulator for Urban and Non-urban Networks (AIMSUN), and Multi-Agent Transport Simulation (MATSim). There are advantages and disadvantages for each of them, which should be discussed to determine the most suitable option for this research.

### A Research Question

This paper aims to explore the integration of XAI techniques with traffic prediction models and evaluate the effectiveness of LLM-generated policy recommendations using the SUMO simulation environment. The research questions guiding this study are as follows:

**RQ 1:** What explainable prediction models are best suited for traffic prediction?

**RQ 2:** LLM facilitates interpretable traffic policy suggestions based on explainable prediction models outputs?

**RQ 3:** How effective are LLM-generated policy recommendations on real-world traffic performance when tested in a simulation environment like SUMO?

### B Method Overview

This study employs a mixed-methods approach, combining a Systematic Literature Review (SLR), model development, and simulation-based evaluation. The literature review synthesizes existing research on traffic prediction models, XAI techniques, and LLM applications in traffic management. Based on insights from the review, an explainable traffic prediction model will be developed using XGBoost and SHAP. LLMs will then be utilized to generate policy recommendations based on model outputs. Finally, the effectiveness of these recommendations will be evaluated in the SUMO simulation environment. Figure 1 illustrates the overall workflow.

This study focuses on the Auckland Central Business District (CBD), a dense urban network with recurrent peak-period congestion, using it as the empirical context for developing and evaluating the proposed pipeline.

### C Document Section

This paper consists of five main sections: Literature Review, which reviews related work and identifies the research gap. Research Methodology, which details the methodology. Implementation and Results, which reports the implementation and findings. Conclusion and Future work, which presents conclusions and outlines future work.

## II. LITERATURE REVIEW

To address the research questions, two structured search string was developed and applied in IEEE and ACM database. The formulated search string is presented below:

*(“traffic prediction” OR “traffic forecasting”) AND (“prediction model” OR “predictive model” OR “machine learning” OR “deep learning” OR “statistical*

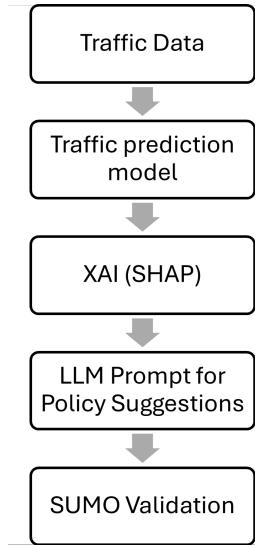


Fig. 1. Workflow Overview

*model”) AND (“road traffic” OR “vehicle traffic” OR “urban traffic” OR “highway traffic”) NOT (“network traffic” OR “computer network” OR “internet traffic”)*

*(“large language model” OR LLM) AND (“traffic prediction” OR “traffic forecasting” OR “traffic flow”) AND (road OR highway OR vehicle OR “road network” OR “urban mobility”) AND (policy OR simulation OR planning) NOT (“network traffic” OR “internet traffic” OR “computer network” OR “telecommunication” OR “5G” OR “6G”)*

The two search strings were designed to systematically capture literature relevant to the dual focus of this study: (1) traffic prediction using data-driven methods, and (2) the emerging integration of LLM into traffic simulation and policy generation.

Table I presents the criteria considered for the literature review, while Figure 2 illustrates the study inclusion process. A total of 35 articles were selected for analysis in this paper.

The literature review was structured into four subsections to align with the main research objectives and to provide a logical progression from foundational models to the emerging integration of XAI and LLM in traffic management. The first subsection discussed the types of models for traffic prediction, establishing the technical background and highlighting the evolution from traditional statistical approaches to ML and DL techniques. The second subsection focuses on the role of LLMs in traffic prediction, addressing the recent rise of Generative Artificial Intelligence (GAI) and its potential to influence predictive modeling and policy generation. The third subsection discusses XAI in traffic prediction, which directly supports this study’s aim of producing interpretable, policy-relevant insights from predictive models. Finally, the fourth subsection reviews simulation environments for traffic prediction evaluation, such as SUMO, which form the experimen-

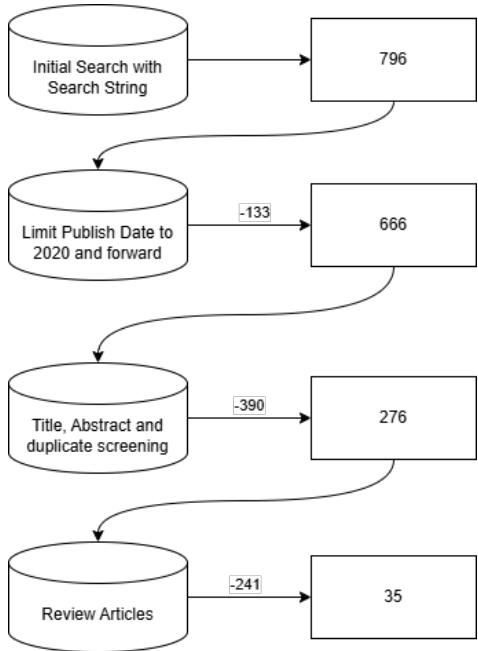


Fig. 2. Study Inclusion Process

TABLE I  
INCLUSION AND EXCLUSION CRITERIA

Criteria	Inclusion	Exclusion
Publication Type	Peer-reviewed journal articles, conference papers, systematic reviews	Non-peer-reviewed articles, blogs, web pages
Time Frame	Publications from the last 5 years (2020 onwards)	Publications before 2020
Focus	Studies evaluating traffic prediction models based on LLM, ML, or DL techniques, and using simulation environments for traffic simulation	Studies without experiments, models, or explainability focus
Domain	Smart cities, intelligent transportation systems, industrial traffic management	Aviation, maritime, telecommunication

tal foundation for testing and validating traffic management strategies in this research.

#### A Types of Models for Traffic Prediction

Traffic prediction methods can generally be categorized into three main types, statistical models, ML models, and DL models. The following section provides a more in-depth discussion of each category.

These three papers used a variation of LSTM and ARIMA to predict traffic volume [7], [16], [17]. ARIMA is a statistical

method that uses historical traffic data to find traffic patterns. After the traffic pattern is found, it is used to forecast future traffic patterns. Wang and Zhang demonstrated that ARIMA can accurately predict short-term traffic patterns using only historical data [17]. Meanwhile, Verma, Pandey, and Rup improved the workflow by integrating LSTM into their model [7]. They used LSTM to make the initial prediction, computed the residual value between the actual and predicted values, and used the residual to train the ARIMA model. The final forecast was the combination of the LSTM prediction and the ARIMA residual forecast, resulting in a more effective model compared with standalone LSTM and ARIMA. Lim, Connie, and Goh investigated ARIMA further and used Seasonal Autoregressive Integrated Moving Average (SARIMA) in their study [16]. SARIMA is an extension of the ARIMA model that can handle seasonal patterns in time series data. This accounts for the seasonality that ARIMA does not naturally handle. Although ARIMA works well in predicting traffic, it assumes that traffic is linear, whereas in reality, traffic is not, and it cannot capture sudden spikes from external factors such as accidents and weather. Furthermore, these three papers did not explore the explainability of the model, which is crucial for the reliability and trustworthiness of a real-time system.

To capture the nonlinearity in traffic data, ML models can be used to address this challenge. XGBoost and LSTM are among the ML and DL models that can utilize nonlinear data for traffic prediction. XGBoost is utilized in Zheng, Ye, Yin, *et al.*'s paper to conduct a comparative analysis between XGBoost and Sparrow Search Algorithm (SSA)-XGBoost. [8] XGBoost is a ML algorithm based on decision trees. It makes accurate predictions by combining many simple trees into one strong model. The SSA is an optimization algorithm that automatically searches for the best combination of hyperparameters to maximize prediction accuracy. Zheng, Ye, Yin, *et al.* compared both models and concluded that the SSA-XGBoost model only improved their model accuracy by 0.17% in terms of R-squared value [8].

XGBoost is built on decision trees, which are generally interpretable, and its ensemble nature makes it less transparent than a single tree. Each prediction is the cumulative result of hundreds of trees, but with using explainable tool such as SHAP, it can trace making trace how specific features influence the outcome. Nevertheless, XGBoost achieves strong predictive performance and often provides a balance between accuracy and interpretability when combined with appropriate explanation methods.

LSTM is a deep learning model designed to remember recent information for a long time and forget less relevant past data when processing sequences. It consists of three gates: the forget gate, which decides what data to discard, the input gate, which decides what new data to store, and the output gate, which decides what information to use for the next step. LSTM can capture patterns over time, handle nonlinear relationships effectively, and remember long-term effects, which is significant in traffic prediction. In Zhang's paper, they developed an LSTM model to forecast three taxi operational metrics: trip

distance, trip duration, and trip fare [9]. Zhang also combined meteorological data and demonstrated that weather factors can improve model robustness. Zhang further improved the model's performance by performing K-means clustering to group pickup and drop-off locations into regions with similar patterns and applied random search to tune LSTM parameters. Li, Xu, Cheng, *et al.* took a similar approach to Zhang but used K-means clustering first to find similar day-pattern data and then used the filtered data to train the LSTM model [18]. This helped avoid training with irrelevant or abnormal data.

Despite their strong predictive capabilities, most machine learning and deep learning models function as black-box systems, offering limited visibility into how individual features influence their outputs. This lack of transparency presents major challenges in traffic prediction, where model outcomes can directly impact real-world operational and policy decisions. Without interpretability, stakeholders are unable to validate or explain the model recommendations, leading to trustworthy problem. Although existing studies have effectively captured temporal dependencies in traffic data, few have examined model interpretability in depth. Even when external factors such as weather were included, their influence on model decisions was rarely analysed. This underscores a critical research gap, the need to identify and explain the key influential features driving traffic prediction models to ensure that their results are both accurate and explainable, thereby supporting transparent and accountable decision-making.

## B LLM in Traffic Prediction

With the rise in the usage of LLM, their applications have started to integrate into different industries to help with automation processes. In this section, the paper is going to discuss different types of LLM implementations in the traffic prediction sector.

Chenxi, Sun, Qianxiong, *et al.* proposed a Spatio-Temporal Large Language Model (ST-LLM) with partially frozen attention for traffic prediction [13]. As mentioned in the paper, existing LLM-based models focus on the temporal aspect of data and often ignore the spatial aspect in traffic prediction. However, spatial data such as lane number and direction are highly relevant for traffic prediction. Chenxi, Sun, Qianxiong, *et al.* proposed the ST-LLM framework to address this research gap. The model defines timesteps at a location as a token, which is then transformed into a unique spatial-temporal embedding layer. This helps emphasize the spatial and global temporal patterns. The proposed model outperformed the baseline models in both full-shot and zero-shot settings.

Yiqing, Xingyuan, and Yisheng introduced an LLM-assisted traffic signal control approach based on the traffic green wave method [14]. The green wave method, commonly used in adaptive traffic signal control systems, creates a continuous sequence of green lights by coordinating signals across a series of intersections. In this study, Yiqing, Xingyuan, and Yisheng generated a scenario using synthetic data and fed it into GPT-4. The evaluation metric employed was average road speed. Compared with random scripted traffic flow and conventional

control methods, the LLM-assisted approach demonstrated superior performance. Although the paper serves primarily as a proof of concept, since it does not incorporate real-world data, actual road networks, or external influencing factors, it highlights the feasibility of applying LLMs to traffic prediction tasks.

Mohammad and Juyeong explored the use of LLM for adaptive traffic signal control, focusing on reasoning and planning tasks [19]. Mohammad and Juyeong proposed two LLM-based controllers: a Zero-Shot Chain of Thought (ZS-CoT) controller and a Generally Capable Agent (GCA)-based controller. The ZS-CoT controller relies on the LLM's pre-trained knowledge to generate traffic phase strategies without additional feedback, while the GCA framework introduces a critic agent that continuously updates text-based knowledge from previous interactions to refine decisions. Both controllers were implemented and tested in a simulated environment using SUMO, and their performance was compared against traditional traffic controllers. The results showed that the GCA-based controller significantly reduced halted vehicles, improved average speed, and increased the proportion of vehicles arriving on green lights. Although the study was limited to a single isolated intersection with simulated traffic demand, it demonstrates the feasibility of LLM-based controllers in enhancing adaptability and reasoning in traffic signal control, suggesting potential for broader real-world application.

Zhonghang, Lianghao, Jiabin, *et al.* proposed UrbanGPT, a LLM designed for spatio-temporal traffic prediction [20]. The model addresses the absence of explicit spatial encoders by integrating extensive geographic information and points of interest data into the textual input. This design allows the model to reason about functional similarities between regions, such as identifying comparable commercial districts in different cities, without requiring a predefined graph structure. UrbanGPT demonstrated strong performance in zero-shot settings by pretraining on other urban datasets and applying instruction tuning to capture both spatio-temporal structure and semantic knowledge, thereby leveraging general world knowledge and cross-city patterns. However, the model suffers from high computational cost and resource requirements, making it impractical in scenarios with limited resources. Furthermore, it lacks built-in explainability, and in cases with abundant clean, long-term labeled data, graph neural network models may achieve superior performance with lower computational demands. Despite these limitations, the study highlights the potential of LLMs to generalize across diverse urban contexts for traffic prediction tasks.

Overall, these studies illustrate the growing role of LLMs in traffic prediction and control, demonstrating their capability to handle complex reasoning, spatio-temporal understanding, and decision-making tasks. However, most existing works remain at the conceptual or simulation level and lack integration with interpretable, data-driven prediction models. This gap aligns with the objective of this research, which aims to leverage LLM-assisted reasoning together with explainable predictive models to support policy generation and simulation.

### C XAI in Traffic Prediction

There is a growing number of XAI techniques available today, with SHAP and LIME being among the most widely adopted. Other notable frameworks include Integrated Gradients, XAI-Toolbox, XAITK, and InterpretML. SHAP is a game theory based method that assigns contribution values to individual features, thereby explaining how each input influences the model's prediction. In contrast, LIME provides local explanations by approximating the behaviour of a complex model with a simpler, interpretable surrogate model. Numerous studies have already investigated the use of SHAP and LIME for enhancing interpretability in traffic prediction, as reviewed below.

Waqas et al. employed an LSTM model to predict traffic congestion levels for autonomous vehicles [10]. The model classified congestion into four categories, blocked, mild, moderate, and severe by using autonomous vehicle sensor readings, weather conditions, and historical traffic logs. SHAP was applied to interpret the model's output, revealing that time of day, traffic volume, weather and holiday indicators were the most influential features. Although the model achieved a high training accuracy (99.8 %) and validation accuracy (99.1 %), this near-perfect performance raises potential overfitting concerns. Moreover, the lack of evaluation on unseen data limits its generalizability to real-world scenarios. Nevertheless, the study demonstrates how SHAP can validate model reasoning and provide transparent insight into predictive factors.

Kong et al. proposed a custom framework named Trafexplainer [11], which integrates interpretability directly into a Graph Neural Network (GNN)-based traffic prediction model. Instead of using existing XAI tools, they developed a Perturbation-Based Hierarchical Interpretation Generator comprising spatial and temporal masks. The spatial mask perturbs the adjacency matrix to learn the influence of first- and second-order neighbours across GNN layers, while the temporal mask identifies the most significant time steps in the traffic series. The enhanced model outperformed the baseline GNN, achieving a lower Root Mean Squared Error (RMSE) of 15.504 compared to 15.561 without the interpretation layer. Interpretation results revealed that traffic conditions approximately 55–60 minutes earlier were the most predictive of future congestion, and that second-order neighbours had greater influence than first-order ones. Although prediction and explanation were trained separately, it does not create a framework that is interpretable from start to finish. Nevertheless, The research shows that adding an interpretation mechanism to a model can improve its predictive accuracy.

Vijaya et al. investigated Random Forest (RF) and k-Nearest Neighbours (kNN) models for predicting public transport delays [12]. While the focus was on bus and train delays rather than road congestion, the work remains relevant due to its detailed comparison of XAI methods. RF, an ensemble approach that aggregates multiple decision trees, achieved a lower Mean Absolute Error (MAE) (3.39) compared with kNN (4.35). Through SHAP, the most influential factors were

identified as time of day, day of the week, and route type, indicating strong correlations between weekday peak hours and delays. Meanwhile, LIME provided instance-level explanations, revealing that evening rush-hour delays were mainly due to route congestion. However, the study's scope was limited to public transit and did not address road traffic flow or dynamic routing, thereby restricting its general applicability to broader traffic networks.

Lee proposed the High-Importance Links with XGBoost (HI-XGB) framework for short-term traffic speed prediction, integrating SHAP for feature-level explainability [21]. The framework operates in two stages: first, feature selection using SHAP to identify high-impact road links, and second perform prediction using XGBoost with the selected features. This approach outperformed baseline models including ARIMA, Support Vector Machine (SVM), LSTM, Gated Recurrent Unit (GRU), and standard XGBoost. Despite its success, the framework excluded external contextual factors such as weather and special event, and relied on SHAP as a pre-processing step rather than as an integrated interpretability layer. Hence, it does not constitute a fully end-to-end explainable model.

In conclusion, the reviewed studies demonstrate that XAI techniques bridge the gap between high predictive accuracy and human interpretability, making AI-driven traffic models more transparent and trustworthy. However, most existing work applies explainability post-hoc, treating interpretation as a separate phase rather than an embedded design element. This motivates the approach of this research. Having established the value of XAI in understanding traffic prediction models, the next chapter explores how a simulation environment can help in traffic policy evaluation.

### D Simulation Environments for Traffic Policy Evaluation

Traffic simulation is a widely used approach to optimize road design and assess the impacts of different infrastructure configurations. It is often confused with traffic prediction, although the two are distinct. Traffic prediction focuses on estimating future traffic conditions by leveraging historical data, whereas traffic simulation evaluates traffic behavior under specified conditions [15]. Simulation environments play a crucial role in representing and assessing traffic performance, ensuring that smart mobility solutions are implemented effectively. Furthermore, traffic simulation can support policymakers by providing evidence insights for decision-making. Depending on the level of detail required, simulation can be applied at varying levels of complexity and city scales, thereby offering valuable contributions to traffic analysis.

Traffic simulation can be classified into three categories, marcoscopic, microscopic and mesoscopic [15], [22], [23]. Macroscopic models describe traffic flow at an aggregate level by formulating the relationships among flow, speed, and density, where simulation occurs on a segment basis rather than tracking individual vehicles. These models typically employ routing algorithms that optimize travel time, making them effective for analyzing network-wide travel demand patterns. In contrast, microscopic models capture vehicle-level dynamics

by incorporating car-following and lane-changing algorithms. This level of detail allows for the evaluation of complex scenarios such as rush-hour congestion or intricate roadway geometries. Hence, microscopic simulations are often computationally intensive, time-consuming, and pose significant calibration challenges. Mesoscopic models serve as an intermediate approach, combining aspects of both macroscopic and microscopic perspectives to balance computational efficiency with behavioral realism [15].

For the purpose of this study, the focus will be limited to microscopic simulation, as it aligns with the research objective of capturing vehicle-level dynamics and evaluating detailed traffic behaviors, such as congestion formation, intersection performance, and the impact of policy interventions on individual driver interactions.

On the other hand, Diallo, Lozenguez, Doniec, *et al.* introduced a structured two-step evaluation framework, combining weighted criteria with a case study of intermodal mobility, to compare academic and industrial simulators [23]. Their analysis concluded that SUMO offers stronger multimodal traffic interaction modeling, whereas MATSim, with its agent-based and activity-based design, is more suitable for large-scale intermodal studies.

TABLE II  
COMPARISON OF COMMONLY USED TRAFFIC SIMULATORS [15], [22], [23]

Simulator	Strengths	Limitations
<b>SUMO</b>	Open-source, flexible, supports multimodal transport, strong API integration (Python/TraCI), widely used in research.	Steeper learning curve; less user-friendly GUI compared to commercial tools.
<b>MATSim</b>	Activity-based design, scalable for large cities, supports intermodal mobility, open-source.	Requires high expertise; complex setup and calibration.
<b>VISSIM</b>	High realism, strong visualization (2D/3D), widely validated in industry, good for signal and intersection studies.	Commercial license, expensive, limited customizability.
<b>AIMSUN</b>	Multi-resolution modeling (micro, meso, macro), strong visualization, widely adopted in industry.	Commercial, costly, less extensible than open-source tools.
<b>CORSIM</b>	Established, good for freeway and arterial studies, U.S. standard tool.	Outdated interface, limited extensibility.
<b>Paramics</b>	Realistic driver behavior models, commercial support.	Expensive, less flexible for research integration.
<b>TRANSIMS</b>	Open-source, supports regional-scale modeling, activity-based.	Complex setup, limited community support.

Ullah, Khattak, H. Khan- P. Eng, *et al.* conducted a systematic comparative review of 29 traffic simulators, categorizing

them into microscopic, mesoscopic, macroscopic, and hybrid approaches [22]. They emphasized that no single simulator is universally optimal, each choice depends on the research objectives. Microscopic tools excel in detailed traffic control analysis, mesoscopic tools in large-scale demand modeling, and hybrid approaches in balancing scale with behavioral realism. In conclusion, these comparative studies highlight that simulator selection must align with the intended application, the required level of detail, and the trade-offs between computational efficiency, cost, and extensibility.

Table II shows the comparison of commonly used traffic simulations models. Following a comprehensive review and comparison of various traffic simulation tools, SUMO was selected as the primary simulation environment for this research. This choice is justified by several key considerations. First, SUMO is extensively utilized in academic and applied research, offering strong validation and a substantial body of published studies to reference. Second, its open-source and free availability eliminates licensing constraints, thereby enhancing accessibility and flexibility in academic contexts. Third, SUMO is supported by an active community and comprehensive documentation, including tutorials, GitHub resources etc, which collectively facilitate effective implementation and troubleshooting. Additionally, SUMO provides extensive integration capabilities through its Python API (Traffic Control Interface (TraCI)), enabling dynamic simulation adjustments such as traffic signal control, speed regulation, and vehicle rerouting. Finally, the platform's high degree of customizability allows for the importation of road networks from OpenStreetMap (OSM) and the modification of layouts, signals, and policy configurations, ensuring alignment with the specific requirements of this research.

TABLE III: Systematic Literature Review

Title	Methods & Metrics	Key Findings & Contributions	Limitations	Database
A Comparative Study on Traffic Modeling Techniques for Predicting and Simulating Traffic Behavior [15]	Comparative review of simulators (SUMO, MATSim, VISSIM, AIMSUN, Corridor microscopic Simulation (CORSIM), Paramics, Transportation Analysis and Simulation System (TRANSIMS)); performance-evaluation considerations.	- Combining spatial and temporal factors improves prediction quality. - Clarifies strengths/weaknesses of non-parametric models and how to mitigate them. - Provides guidance for choosing a simulator based on intended use.	- Survey paper (no single benchmark dataset evaluated). - Does not validate causal impact of policy via micro-simulation.	N/A
Vehicular Traffic Simulation Software: A Systematic Comparative Analysis [22]	- SUMO	- Maps simulator strengths/limitations - Helps structure a comparison table of tool capabilities	- Literature-only synthesis - Absent explainability and AI policy generation	N/A
Traffic Speed Prediction of Urban Road Network Based on High Importance Links Using XGB and SHAP [21]	- HI-XGB - SHAP	- Proposes HI-XGB: XGBoost with high-importance links selected via SHAP. - Directly aligns with XGB+SHAP; shows HI-XGB (SHAP-selected links) yields Mean Absolute Percentage Error (MAPE)=0.015 at 1-step. - Demonstrates global/local SHAP use to interpret dominant features (target speed t, delta speed t-1).	- Stops at interpretation - No traffic policies suggested	- TOPIS data
A Federated Learning Framework for Automated Decision Making with Microscopic Traffic Simulation [24]	- FedMMTS - SUMO	- used experiment as their research method as well. - Validates SUMO choice and outlines TraCI variables for behavior	- Lacks explainability	US highway101 traffic data
Design of Urban Road Traffic Induction Algorithm based on DL Algorithm [25]	- Comparison of several DL	- conducted comparative modeling experiments, train and test several DL technique against traditional baselines on loop detector time series, and report error such as RMSE, MAE and MAPE.	- Lacks explainability - Accuracy emphasized, but policy outcomes untested	N/A
Support Vector Machines (SVM) in Traffic Prediction for Intelligent Transportation Systems- A comprehension Review [26]	- SVM	- A comprehensive review of key ML and DL techniques used in traffic forecasting, accompanied by a discussion of the challenges associated with their implementation. - Provides theoretical baseline to contrast tree-based models (XG-Boost) with SVM/SVR on tabular traffic data.	- Review focuses on algorithms, not on explainability, driven interventions	N/A
ML Applications in Vehicular Traffic Prediction and Congestion Control: A Systematic Review [27]	- ML - DL - Convolutional Neural Network (CNN)	- Reviews AI/ML/IoT for congestion control and the ITS pipeline. - Aligns with my end-to-end approach (data → model → policy → signals)	- Lacks explainability	N/A
Towards Data Empowered Intelligent Transportation Systems: Use Cases and Future Opportunities [28]	- ITS	- conducted a case study using k-means clustering to demonstrate how big data can improve ITS performance.	- Lacks explainability	N/A
Intelligent Transportation Systems Utilizing UAVs: Integration with IoT and ML [29]	- ML	- examines the integration of UAVs, IoT, and ML in ITS, highlighting their potential to transform transportation management.	- Lacks explainability	N/A
Short term traffic congestion prediction using publically available traffic data: a case study on Timisoara [30]	- ML	- Validates 15-min aggregation for urban short-term tasks and shows deepmodels miss spikes.	- Lacks explainability	urban traffic flow of 13 Romanian cities

Title	Methods & Metrics	Key Findings & Contributions	Limitations	Database
Short-Term Traffic Prediction Using DL LSTM: Taxonomy, Applications, Challenges, and Future Trends [5]	- LSTM - CNN - Graph Convolutional Network (GCN)	- Catalogs LSTM-based approaches from the last 3 years - Explains how combinations (e.g., CNN/GCN/attention) capture spatio-temporal dependencies - Provides comparative insights to help select algorithms for diverse traffic conditions - Highlights common applications and trends for short-term forecasting	N/A	N/A
Long-Term Traffic Prediction Based on LSTM Encoder-Decoder Architecture [6]	- LSTM	- Demonstrates feasibility of LSTM encoder-decoder for longer horizons - Shows sequence-to-sequence modeling can retain temporal context beyond short windows	- Long-horizon accuracy may degrade - Requires larger training data and careful regularization to reduce drift/overfitting	N/A
Prediction of Traffic Congestion Based on LSTM Through Correction of Missing Temporal and Spatial Data [4]	- LSTM	- Proposes a LSTM-based model for predicting road traffic congestion. - Introduces a missing value correction method that handles both temporal and spatial missing traffic data - Achieves a low (MAPE) of about 5% on average, outperforming previous models like Recurrent Neural Network (RNN)s, basic LSTMs, and STGCNs.	- The model lacks explainability tools, making it difficult to interpret or validate predictions, especially important for real-world traffic safety applications. - uses limited traffic factors to predict traffic, only uses heuristic traffic speed data to predict traffic, wheres there are many factors that can also influece the traffic flow.	N/A
Traffexplainer: A Framework Toward GNN-Based Interpretable Traffic Prediction [11]	- perturbation-based hierarchical masking	- framework that adds interpretability to GNN-based traffic prediction models	- Although proposed solution provides post-hoc explanations by training hierarchical masks after the predictor is frozen, it does not constitute an end-to-end interpretable learning framework. Prediction and explanation are separated, which could introduce gaps between the model true decision process and the generated interpretations.	N/A
DL on Traffic Prediction: Methods, Analysis, and Future Directions [2]	- DL	- Provides a systematic review of DL-based traffic prediction methods - Summarizes publicly available traffic datasets and external factors affecting traffic.	N/A	N/A
Optimizing Traffic Speed Prediction Using a Multi-Objective Genetic Algorithm-Enhanced RNN for Intelligent Transportation Systems [3]	- RNN - Bi-GRU - CNN-GRU - CNN-LSTM - LSTM-AM	- Proposes a Multi-Objective Genetic Algorithm (MOGA)-optimized (RNN) model to predict short-term traffic speed efficiently and accurately.	- The model lacks explainability tools, making it difficult to interpret or validate predictions, especially important for real-world traffic safety applications. - uses limited traffic factors to predict traffic, only speed was considered to predict traffic, wheres there are many factors that can also influece the traffic flow.	N/A
Advanced Learning Technologies for Intelligent Transportation Systems: Prospects and Challenges [1]	- feedforward neural networks (FNN) - CNN - RNN - Capsule Networks (CapsNets) - Autoencoders - Graph Convolutional Neural Networks (GCNNs)	- Provides a detailed explanation of ITS, their components, and operational challenges - Application of DL such as traffic flow prediction, vehicle detection/classification, traffic sign recognition, pedestrian detection, road condition monitoring, autonomous driving - challenges faced by DL techniques within ITS - highlighted importance of XAI for ITS	- The paper briefly mentions the issue of DL models being black-boxes, but it does not explore XAI methods in depth. Given the critical importance of safety and transparency in transportation systems, a more thorough discussion on explainable models would greatly strengthen the paper. - it lacks coverage of newer DL architectures such as Transformers, which are increasingly state-of-the-art for many ITS applications. - The paper mainly focuses on theoretical applications and overlooks real-world deployment issues, such as computational resource constraints, latency, and energy efficiency, which are crucial considerations when implementing ITS solutions in practice.	N/A
Autonomous vehicles congestion model: A transparent LSTM-based prediction model corporate with Explainable Artificial Intelligence (EAI) [10]	- SHAP - LSTM	- LSTM networks for time-series forecasting, - XAI techniques to provide transparency in predictions.	- 99.8% training accuracy and 99.1% validation accuracy seem very high, concerns of overfitting - the paper does not have visualizations or detailed explanation into which features (e.g., weather vs. time of day) drove decisions.	N/A

Title	Methods & Metrics	Key Findings & Contributions	Limitations	Database
Enhancing Transparency in Public Transportation Delay Predictions with SHAP and LIME [12]	- RF - kNN - SHAP, LIME	- Developed a public transportation delay prediction system using RF and k-Nearest Neighbors, integrated with XAI techniques: SHAP and LIME. - Applied SHAP for global feature importance analysis	- Although the paper focuses on predicting public transportation delays rather than road traffic congestion or flow, it remains highly relevant due to ITS extensive application of XAI methods.	N/A
Spatial-Temporal LLM for Traffic Prediction [13]	- Transformer backbone with spatial temporal embedding - Partially Frozen Attention, and regression head - Compared across full/few/zero shot settings.	- LLMs have shown promise in generating human like text and reasoning about complex problems. - Proposed ST LLM, a spatial temporal LLM model with Partially Frozen Attention for traffic prediction - Outperformed baselines in full shot and zero shot settings on two datasets.	- Pretrained LLMs misalign with time series data - No real world deployment - Manual tuning of frozen layers - No interpretability methods - High resource demand.	NYCTaxi (35M+ records) CHBike (2.6M records)
The Crossroads of LLM and Traffic Control: A Study on LLMs in Adaptive Traffic Signal Control [19]	- ZS-CoT prompting - GCA dual agent system with memory - GPT 5 - SUMO	- Proposed and compare LLM based frameworks (ZS-CoT and GCA) for adaptive traffic signal control using GPT 5 - GCA performed better than ZS-CoT - Demonstrated that LLMs can generate effective signal phase decisions via natural language reasoning, improving average speed and reducing vehicle halts in simulation.	- Used synthetic data , lacking validation in real world traffic scenarios with uncertainties - The controller's performance depends heavily on prompt engineering and pretrained LLM knowledge, which may not generalize across diverse intersections or unforeseen traffic conditions - Safety is enforced post decision, which may not catch all unsafe configurations, especially in multi intersection networks.	Synthetic traffic data
LLM-Assisted Arterial Traffic Signal Control [14]	- GPT 4 - SUMO	- Introduced a LLM assisted traffic signal control based on the traffic green wave method - LLM model is performed better	- Used synthetic data - SUMO only evaluation - Relies based on traffic green wave method, less efficient under highly variable or unpredictable traffic patterns.	Synthetic arterial traffic data no public datasets used.
UrbanGPT: Spatio-Temporal LLMs [20]	Spatio-Temporal - GPT	- introduced UrbanGPT, a LLM for traffic prediction - compensates for the absence of explicit spatial encoders by integrating extensive geographic and points of interest data within the textual input - This allows the model to reason about functional similarities between regions (e.g., two commercial districts in different cities) without needing a predefined graph - perform well in zero shot setting (pretrain on other urban dataset, using instruction tuning to encode both ST structure and semantic knowledge, leveraging general world knowledge + cross city pattern)	- High Computational Cost & Resource Requirements, not practical in my case - no explainability of model - other GNN model can perform better with less computation if using consists of plenty clean, long-term labeled data.	N/A
Comparative analysis of traffic flow prediction based on XGBoost before and after optimization [8]	- XGBoost - SSA - R Squared	- XGBoost vs SSA - XGBoost, slightly improved accuracy ( $\tilde{0.17\%}$ better $R^2$ )	- took way more time because SSA has to train many XGBoost models during the search	Xinjiang Expressway Data
A Novel Multi-Objective LSTM Framework for Enhancing Operational Efficiency of Yellow Taxis in New York City Using Big Data and Regional Clustering Analysis [9]	- LSTM - Kmean - Random Search	- Developed an LSTM based prediction model that simultaneously forecasts three taxi operational metrics (trip distance, trip duration and trip fare) - combined meteorological data and proved weather factors improved model robustness and prediction accuracy - integrated kmean clustering to group pickup and drop off locations into region with similar demand pattern, which result improved accuracy in trip distance and fare prediction too - Applied Random Search to tune LSTM parameters, improving generalization and reducing overfitting.	- Lacks explainability	NYC Yellow Taxi Data National Weather Service (NWS) historical weather database.
Traffic Flow Prediction Using LSTM, ARIMA & SARIMA Model for Intelligent Transportation System [7]	- LSTM - ARIMA - SARIMA	- Investigated LSTM, ARIMA, and SARIMA models for short term traffic flow forecasting	- Lacks explainability	Seattle traffic flow data

Title	Methods & Metrics	Key Findings & Contributions	Limitations	Database
Enhancing Traffic Analysis and Prediction through A Hybrid LSTM-ARIMA Model [16]	- ARIMA - LSTM	<ul style="list-style-type: none"> <li>- introduce an altogether more inclusive approach called stacked LSTM-ARIMA for addressing traffic forecasting problems</li> <li>- LSTM-ARIMA is more effective than both standalone LSTM and ARIMA approaches since it has a stacking MAE of 0.2662 in case of outgoing traffic as compared to 0.5385 for incoming traffic.</li> </ul>	<ul style="list-style-type: none"> <li>- no explainability of model</li> <li>- ARIMA is linear, cannot capture sudden spikes from accidents or weather</li> <li>- ARIMA work best on a single site, but hard to use for entire network without separate models</li> </ul>	N/A
Research on ARIMA Model for Short-Term Traffic Flow Prediction based on Time Series [17]	- LSTM - ARIMA	<ul style="list-style-type: none"> <li>- Demonstrated that ARIMA can accurately model short term traffic patterns using only historical time series data</li> </ul>	<ul style="list-style-type: none"> <li>- no explainability of model</li> <li>- ARIMA is linear, cannot capture sudden spikes from accidents or weather</li> </ul>	5-minute interval traffic volume data from electronic police cameras over five days as the training set.
Short-Time Traffic Flow Prediction Based on K-means++ and LSTM [18]	- LSTM - Kmean ++	<ul style="list-style-type: none"> <li>- Filtering training data by similar day patterns improves real time short term prediction</li> <li>- Helps avoid training with irrelevant or abnormal data, leading to more stable forecasts.</li> </ul>	- Lacks explainability	PeMS04

## E Discussion

Table III summarizes the key studies reviewed in this paper, outlining their methodologies, contribution, limitation and dataset within the context of traffic prediction technique, XAI and simulation environment. The reviewed papers and the discussion above reveal a research gap, which is integrating interpretable predictive models with real world simulation environment.

**RQ 1:** What explainable prediction models are best suited for traffic prediction?

Among the explainable prediction models reviewed, XGBoost is the most suitable choice for this research. Its ability to capture nonlinear relationships in traffic data, combined with strong predictive accuracy, has been demonstrated in comparative studies such as Zheng, Ye, Yin, *et al.*'s work on XGBoost and SSA-XGBoost [8]. More importantly, XGBoost integrates effectively with SHAP, a widely adopted explainability framework that provides feature-level insights into model outputs. For example, Lee proposed the HI-XGB model, which combines XGBoost with SHAP for urban traffic speed prediction, achieving superior performance compared with other baseline models [21]. This integration not only enhances predictive performance but also offers interpretable outputs by identifying the most influential road links and temporal patterns in traffic dynamics. For this reason, XGBoost will be adopted as the primary prediction model in this study, with SHAP employed to explain and validate its outputs.

**RQ 2:** LLM facilitates interpretable traffic policy suggestions based on explainable prediction models outputs?

The ST-LLM proposed by Chenxi, Sun, Qianxiong, *et al.* highlights the potential of combining spatial and temporal embeddings for traffic prediction, while UrbanGPT developed by Zhonghang, Lianghao, Jiabin, *et al.* demonstrates how LLMs can leverage geographic and semantic data for cross-city generalization [13] [20]. Furthermore, LLM-assisted traffic signal control approaches, such as the GPT-4 based method presented by Yiqing, Xingyuan, and Yisheng and the ZS-CoT and GCA controllers evaluated by Mohammad and Juyeong, show promising results in improving traffic flow and reducing congestion [14] [19]. Although many of these studies remain proof-of-concept or simulation-based, the literature suggests that LLMs can serve as a bridge between technical model predictions and interpretable, policy-level decision-making.

**RQ 3:** How effective are LLM-generated policy recommendations on real-world traffic performance when tested in a simulation environment like SUMO?

The effectiveness of LLM-generated traffic policy recommendations must be validated in simulation environments that capture real-world dynamics. Simulation platforms such as SUMO are widely used in academic research for modeling vehicle interactions and testing traffic control strategies under varying conditions [15], [22], [23]. SUMO provides a means of evaluating whether LLM-generated decisions can translate into measurable improvements in key metrics such as average travel time, congestion reduction when compared with traditional methods. Thus, this research will address RQ3

by implementing LLM-generated policy suggestions within SUMO to assess their practical impact on traffic performance.

## III. RESEARCH METHODOLOGY

The choice of research method is important because it directly influences how well the study can address its questions and the credibility of its conclusions. This paper considers three methodological approaches, experiment, survey, and case study. This paper discusses and evaluates them against predefined criteria: alignment with the research objectives, suitability to the problem context, depth and type of insight produced.

## A Experiment

An experiment is a method designed to test hypotheses by manipulating one or more independent variables, then observing the effect on outcome measures. In traffic studies mean simulation studies with before and after comparison and predefined performance metrics. Xia conducted comparative modeling experiments, train and test several DL technique against traditional baselines on loop detector time series, and report error such as RMSE, MAE and MAPE [25]. The author then propose an ensemble fusion to improve prediction accuracy.

Ahmed, Muvdi, Liu, *et al.* used experiment as their research method as well. The authors conduct the experiment focusing on decision making for vehicles [24]. They train an LSTM model on NGSIM and create an evaluation scheme in SUMO using simulation variables to investigate traffic behaviour. While both paper demonstrate how experiments can examine model performance or effects under controlled conditions, this approach is not the best fit for this paper. This paper objective emphasize on investigate real world traffic patterns and policy using XAI, and validates LLM policy suggestion using SUMO.

## B Systematic Literature Review

A systematic literature review is a synthesis of prior studies that follows an explicit protocol such as search strings, inclusion/exclusion, quality appraisal to answer focused research questions and map evidence without introducing selection bias. In traffic analytics, SLR commonly scope model classes, data sources, and evaluation metrics. For example, Johny and Dahiya surveys several AI techniques including DL, ML used for congestion forecasting in ITS, organizing methods and reporting challenges such as data availability, privacy, and multi-intersection coordination [27].

Maulida and Mutijarsa also done similarly approach where they scopes techniques and trends for density prediction across recent IEEE papers, functioning as a structured map of approaches rather than an implementation study [31]. Finally, Saghrje, Karan, Kurnaz, *et al.* provides a targeted synthesis on SVM, outlining theoretical foundations, kernels, application and limitations in traffic prediction which again emphasizing compilation over new innovation design [26].

Although prior work shows that SLRs can thoroughly synthesize evidence and identify gaps, a full SLR does not suit

this paper's aims, which focus on designing, implementing, and causally testing a pipeline in a single real-world context. Instead, SLR principles are applied as a secondary method to reduce selection bias in the literature review. Specifically, a systematic evidence scan is conducted, guided by SLR practices such as predefined search strings, screening, and data extraction. Searches were run on IEEE Xplore and the ACM Digital Library for 2020–2025 using terms related to traffic prediction, ITS, SUMO, and XAI. Titles and abstracts were screened against inclusion criteria. For included studies, the extracted items were the contribution, dataset, method, and limitations. The resulting synthesis structured the literature review taxonomy and highlighted gaps that motivate this paper's objective.

### C Case Study

A case study is an empirical inquiry that investigates a situation within its real-world context, especially when the boundaries between the situation and context are not clearly evident. It is suited to questions of “how” and “why,” supports mixed data sources, and emphasizes depth over breadth. For example, Alahmadi conducted a case study using k-means clustering to demonstrate how big data can improve ITS performance [28]. The simulation demonstrate how vehicles offload driving decisions (e.g., lane changes, turns, routing) to the cloud via roadside unit (RSU) and retrieve results from any RSU. The main challenge is selecting the optimal nearby RSU to efficiently deliver the computed results back to the vehicle. This approach shows the advantages of using data for decision making in improving ITS performance.

Avramoni, Iovanovici, Iliescu, *et al.* aimed to identify patterns and predict traffic congestion caused by factors such as accidents. To support this objective, they conducted a case study in Timisoara, Romania, focusing on the street Calea Stan Vidrighin [30]. The goal was to forecast traffic one hour ahead and compare the predictions with ground-truth data. They performed multiple runs of Bayesian optimization over a few hyperparameters and, after generating and training 40 architectures, concluded that sigmoid activation functions outperform ReLU. The best loss achieved was 0.019 for an architecture with a learning rate of 2.852e-05, five layers of LSTM cells, 32 dense layers, and a sigmoid activation function. Their approach shows how a case study can provide in-depth insight into traffic patterns, and it aligns with this paper's objective as well.

Rahman, Zhang, and Abdel-Raheem examines the integration of Unmanned Aerial Vehicles (UAVs), the Internet of Things, and ML in ITS, highlighting their potential to transform transportation management [29]. Through case studies in urban traffic management, emergency response, intelligent bridge monitoring, smart city transportation monitoring, and flood management, the paper demonstrates significant improvements in traffic flow, emergency response times, and infrastructure monitoring. It also discusses challenges such as UAV battery limitations, bandwidth constraints, and privacy concerns, and outlines future directions including quantum

computing integration and swarm UAV technology. Overall, the paper provides a comprehensive overview of the current state of UAV–IoT–ML integration in ITS and identifies key areas for innovation to address modern transportation challenges and achieve sustainable mobility. The case studies illustrate the practical applications and challenges of emerging ITS technologies.

By comparing all three method, case study is the most appropriate for this paper. The case study method aligns well with the research objectives, which focus on understanding how explainable AI can inform traffic policy in a real-world context. It allows for an in-depth exploration of the complex interactions between predictive modeling, explainability, and policy implementation within a specific urban environment. Additionally, the case study approach supports the integration of diverse data sources, including traffic data, model outputs, and simulation results from SUMO, providing a comprehensive view of the research problem. Therefore, a case study methodology is adopted to investigate the application of XAI in traffic prediction and policy generation.

## IV. IMPLEMENTATION

### A Case Study Context & Scope

Auckland is New Zealand's largest and most populous city, with a population of over 1.71 million people as of 2025. Auckland is also the economic hub of New Zealand, contributing around 38% of the country's GDP. The city is known for its beautiful natural surroundings, including beaches, parks, and volcanic cones. Aucklander often refer Auckland into north shore, west auckland, central auckland and south auckland, where central auckland is the main business district and the most densely populated area, connected by several main motorways to other Auckland region. This is also why central auckland often experience heavy traffic congestion during peak hours, as commuters from surrounding suburbs travel into the city for work. To address these issues, Auckland has implemented various initiatives, such as expanding public transportation options and promoting active transportation modes like cycling and buses. Despite these efforts, traffic congestion remains a significant concern for Aucklander.

This makes Auckland CBD an ideal case study location for investigate how XAI can inform traffic policy. The study will focus on key arterial roads and intersections within the CBD area, where congestion is most pronounced during peak hours. The scope will include data collection from traffic sensors to build predictive models. By focusing on Auckland CBD, the research aims to develop actionable insights and policy recommendations that can be directly applied to reduced congestion and improve overall traffic flow in this critical urban area.

### B Data Sources & Pre-processing

There are two main organization that manages traffic in Auckland, New Zealand Transport Agency (NZTA) and Auckland Transport (AT). NZTA is responsible for the national state highway network, while AT manages local roads, public



Fig. 3. Auckland CBD Road Network

transport, and active transportation within the Auckland region. Both organizations collect extensive traffic data through various means, including traffic sensors, cameras, GPS data from vehicles, and public transport usage statistics. For this case study, traffic data will be collected from AT. A formal request is made to AT for access to historical traffic data, including lane level vehicle count for key arterial roads and intersections within Auckland CBD. The raw traffic data were provided by AT in CSV format through their data request system. The data will cover a period of recent two year to capture seasonal variations and trends.

As shown in Figure 3, the road network within Auckland CBD is complex, with numerous intersections, one-way streets, and varying lane configurations. The dataset includes 40 detectors (also referred to as sites) placed at key locations to monitor traffic flow. Each detector records vehicle counts at 1-hour intervals, with each lane recorded separately. The dataset spans from 2023-01-01 to 2025-07-31. To prepare the data for modeling, several preprocessing steps will be applied. First, data cleaning will address missing values and outliers. An initial review shows that some detectors have substantial missing data, possibly due to road maintenance or infrastructure upgrades. Therefore, any day with continuous zero counts will be removed from the dataset. Although this approach is not ideal, as it may discard valuable information, time-series imputation can introduce bias. Thus, removing the affected periods is the most reliable way to ensure data quality.

After data cleaning, feature engineering will be performed to create additional variables that can improve model performance. Temporal features such as time of day and day of week will be included to capture recurring traffic patterns. Spatial features, such as traffic direction, will also be included to improve performance. Finally, the dataset will be split into training and test sets, preserving temporal order to prevent data leakage.

After processing, the final dataset figure 4 contains 7,863,406 rows and 10 columns, including timestamp, detector

<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 7863406 entries, 2023-01-01 00:00:00 to 2025-07-31 23:00:00
Data columns (total 10 columns):
# Column Dtype
0 Detector_ID int64
1 Lane int64
2 Volume int64
3 Direction int64
4 hour int32
5 day int32
6 dayofweek int32
7 month int32
8 year int32
9 is_weekend int64
dtypes: int32(5), int64(5)

Fig. 4. Dataframe info after preprocessing

id, lane id, vehicle count, hour, day, month, year, is\_weekend, and direction.

### C Predictive Models

The goal of the predictive modeling stage is to forecast hourly traffic volume at each lane detector within the Auckland CBD. This is formulated as a regression task in which the model predicts the number of vehicles per lane per hour. Several techniques and models were explored in the literature review. After careful consideration, XGBoost is selected as the primary training method because it performs well on tabular data and integrates effectively with XAI techniques. XGBoost is selected because its strong performance in tabular dataset, it can handle non linear interaction, and traffic data is non linear. Moreover, XGBoost integrates well in SHAP explainability. As SHAP explain model by showing how each feature affects the prediction model, SHAP can explain XGBoost easily and straight forward. As each tree split clearly shows a feature contribution.

TABLE IV  
FEATURE CATEGORIES FOR TRAFFIC VOLUME PREDICTION

Category	Features	Description
Temporal	hour, day, dayofweek, month, year, is_weekend	Time-based features to capture daily and weekly traffic patterns.
Spatial	lane, direction	Encodes directional flow.

Table IV summarizes the feature categories used in the model. Temporal features capture daily and weekly traffic patterns, while spatial features encode directional flow. The target variable is the vehicle count per lane per hour.

Given that the objective of this research is evaluate explainability and policy generation, the model was implemented using parameters that derived from recommended defaults and prior studies in traffic prediction task, instead of using exhaustive hyperparameter optimization method. The model was implemented using the XGBRegressor from the XGBoost library, configured with the root mean squared error objective function and the histogram-based tree construction method

to optimize computational efficiency. Key hyperparameters were tuned to balance model complexity and generalization, including a maximum tree depth of 8, learning rate of 0.05, and 3,000 boosting estimators with early stopping after 50 rounds to prevent overfitting. Additionally, a subsample ratio of 0.8 and column sampling ratio of 0.8 were used to enhance robustness. The model was trained using an 80:20 temporal split, with evaluation sets defined for both training and test partitions to monitor learning behavior during training. The training process reported evaluation metrics every 100 iterations, allowing for performance monitoring. Although no formal optimization method is used in hyperparameter tuning, the final model achieved consistent accuracy across multiple runs, validate that model complexity was sufficient.

For baseline comparison, the trained model's predictions were evaluated against the ground truth traffic volumes to benchmark its performance. The modeling process was conducted using Python 3.12, with key libraries including scikit-learn, pandas, and XGBoost. All experiments were executed on a workstation equipped with an NVIDIA RTX 3050 Ti GPU and an AMD Ryzen 7 5800H processor, ensuring efficient training and evaluation. The finalized model was saved for explainability analysis, where SHAP-based interpretation will be applied to identify the most influential temporal and spatial features affecting traffic volume predictions.

#### D Explainability (XAI)

While the predictive model achieved strong performance, interpretability is important for understanding the factors influencing traffic flow. XAI methods provide insights into how temporal and spatial variables contribute to predictions, enabling data-driven policy recommendations for traffic management. To interpret the predictions made by the XGBoost model, SHAP will be employed as the primary explainability technique. SHAP is a game-theoretic approach that assigns each feature an importance value for a particular prediction, allowing for both global and local interpretability of the model's outputs.

The SHAP analysis was implemented using the SHAP Python library with TreeExplainer, directly compatible with the trained XGBoost model. Explanations were computed for the test dataset to evaluate both global feature importance and instance-level (local) explanations.

To interpret the SHAP results, several visualization techniques were utilized. Summary plots were generated to provide an overview of feature importance across all predictions, highlighting which temporal and spatial features most significantly influenced traffic volume forecasts. Dependence plots were created to illustrate the relationship between individual features and their SHAP values, revealing how changes in specific variables impact model predictions. Additionally, force plots were used for local explanations, visualizing how each feature contributed to the prediction for specific instances, such as peak congestion hours.

##### a) Congestion Hotspot Identification

Due to time constraints and the complexity of the Auckland CBD road network, a focused congestion analysis was conducted to identify the most congested detector sites rather than performing a full network-wide simulation. The analysis used predicted hourly traffic volumes from the trained model to estimate congestion severity at each detector. Only peak-hour periods (07:00–09:00 and 15:00–18:00) were considered, as these represent the most critical times for urban traffic management. Besides, non peak hour have relatively stable flow and offer variance, offering limited insights, including them would lower the model sensitivity to congestion dynamic. Concentrating on peak hours also aligns with real world operation policies where signal timing, lane allocation and flow control interventions are primarily deployed during congestion period.

For each detector, the total hourly vehicle volume was aggregated using grouped data operations. The effective number of active lanes was estimated by counting lanes with non-zero flow and taking the 95th percentile of active lane counts to avoid bias from occasional lane closures. Assuming a capacity of 200 vehicles per lane per hour, the total site capacity was computed as the product of effective lanes and per-lane capacity. The utilization ratio was then calculated as the ratio of observed traffic volume to estimated capacity, serving as a congestion indicator.

Key performance indicators were derived for each site. A severity level ("Free," "Moderate," "High," or "Severe") was assigned based on the 95th-percentile utilization threshold. Sites were then ranked by congestion severity and frequency to identify the most critical bottlenecks.

Finally, these results were merged with geographic coordinates obtained from Auckland Transport's site list to enable spatial visualization. A heatmap was generated to display congestion hotspots across the Auckland CBD, highlighting locations where observed or predicted utilization consistently exceeded capacity.

#### E Policy Generation

To automate the translation of model explainability results into actionable traffic interventions, a two-phase LLM and simulation feedback pipeline was implemented. This framework integrates SHAP-based evidence, network structure, and simulation results to iteratively propose and evaluate policy adjustments within the Auckland CBD network.

##### a) Input Preparation and Network Parsing

The process begins by extracting relevant information from the SUMO network files (.net.xml) and detector outputs (.xml). Custom parsers were developed using Python's xml.etree.ElementTree and pandas libraries to summarize:

- **Network topology:** total number of edges, junctions, lane count, and average lane speeds.
- **Detector data:** lane-level flow, speed, and occupancy aggregated by site.

- **Simulation summary:** average mean speed, mean travel time, and total active vehicles.

These components were stored as structured JSON objects capturing both static network geometry and dynamic traffic performance indicators from baseline simulations.

#### *b) Prompt Construction and Policy Generation*

The summarized data, together with SHAP explainability results, were embedded into a structured prompt and submitted to the GPT-5 model through the OpenAI API. Each prompt contained:

- Network, detector, and simulation summaries.
- Explainability evidence highlighting key temporal and spatial contributors to congestion.
- Operational constraints including signal cycle range and pedestrian safety rules.
- Task list for specific policy recommendations targeting congestion reduction, formatted as JSON with reasoning, actions, and modified XML snippets.

The LLM was instructed to generate policy recommendations in a valid JSON structure containing reasoning, recommended actions, and modified XML snippets such as `<t1Logic>` elements. Recommended actions included adjustments to signal timing, creation or refinement of traffic-light programs, and corridor coordination. All generated responses were archived for reproducibility.

#### *c) Policy Application and Re-Simulation*

The LLM-generated JSON output was parsed and applied to the baseline SUMO network using a custom network-modification module. This module primarily updates the original network according to the LLM output and validates the changes to ensure robustness, as LLM-generated policies can contain bugs. The updated network, containing the new control logic, was then re-simulated using the simulation runner module. Performance metrics such as mean speed, travel time, and congestion levels were extracted and compared against the baseline, providing quantitative evidence of policy effectiveness.

#### *d) Iterative Fine-Tuning*

Following the initial policy evaluation, a second refinement stage was performed. In this iteration, the LLM received updated simulation comparison data and detector summaries, focusing exclusively on optimizing existing signal timings rather than creating new controls. This iterative approach allowed the pipeline to converge toward improved network performance while maintaining compliance with safety and operational constraints. The reason why this is necessary is that by performing once, the LLM generated policies might not be optimal. Through multiple iterations, the LLM can learn from previous results and refine its suggestions to achieve better traffic outcomes.

## F SUMO Integration

In SUMO, each simulation run requires a set of interdependent input files that collectively define the road environment, traffic demand, and monitoring elements. The three core components are summarised in Table V.

The `.net.xml` file describes the physical road network, including all edges, lanes, junctions, and signalised intersections. The `.trips.xml` file specifies the traffic demand by defining the individual vehicle trips, each with a unique departure time, origin edge, and destination edge. The `.add.xml` file provides supplementary information such as induction loops or area detectors that record real-time flow, speed, and occupancy during simulation.

TABLE V  
KEY SUMO FILES USED IN THE SIMULATION PIPELINE.

Component	Functionality
<code>.net.xml</code>	Represents the road network structure, including edges, junctions, lanes, and traffic lights.
<code>.trips.xml</code>	Defines vehicle trips and traffic demand for simulation scenarios.
<code>.add.xml</code>	Specifies additional simulation elements such as detectors.

These files are processed by SUMO's simulation engine, which computes vehicle movement using a microscopic traffic model. Each vehicle is represented as an autonomous agent that follows car-following and lane-changing rules while interacting with other vehicles, road geometry, and traffic-signal logic. The simulator iteratively updates vehicle positions at sub-second intervals, producing output data on speed, travel time, and detector measurements.

SUMO integration forms the practical bridge between the analytical phase, where congestion causes are interpreted through explainability and LLM reasoning, and the experimental phase, where their effects are tested in a simulated traffic environment. It is a critical component of the policy generation pipeline, encompassing the stages before, during, and after policy application.

#### *a) Before Policy Generations*

The process starts by cropping the network to the area of interest, corresponding to the most congested site identified by the explainability analysis. The network is then cleaned using SUMO's NetEdit tool to remove unused edges and junctions. Next, detectors are added to each lane of the selected site to capture lane-level traffic data. To generate traffic demand, a traffic generation function is created to produce a `trips.xml` file containing vehicle trip definitions, each representing a single vehicle journey from an entry edge to an exit edge within the cropped network.

A total of 5,692 trips were generated, reflecting the average historical traffic volume observed at the selected junctions according to Auckland Transport's detector data. This ensures the simulated demand realistically matches real-world flow conditions during typical peak periods.

Finally, a baseline SUMO simulation is executed and its results recorded for policy generation.

#### *b) During and after Policy Generations*

The baseline simulation results are one of the input in the LLM prompt, alongside the network and detector summaries

and SHAP explainability evidence. After the LLM generates policy recommendations, the network modification module applies the suggested changes to the original .net.xml file, creating an updated network file that incorporates the new traffic-signal logic.

Then a simulation is run again with the same baseline configuration but using the modified network file. This allows for a direct comparison of traffic performance before and after policy application. This also applies to the iterative fine-tuning stage, where the updated network from the previous iteration is used as the new baseline for further policy refinement.

## G Evaluation Design

The evaluation design aims to assess the effectiveness of the proposed LLM-generated traffic policies in improving network performance compared to the baseline network. It defines the experimental setup, performance indicators, and comparison criteria used to quantify changes in traffic flow, speed and delay. This ensures that all simulations are conducted under consistent conditions and that improvements are measured objectively using reproducible metrics.

### a) Objectives and Hypotheses

This study evaluates whether LLM-generated signal policies improve peak-hour operations relative to the baseline control plan. The primary hypotheses are:

- 1) average network speed increases
- 2) average delay and travel time decrease

### b) Experimental Conditions

Three scenarios are tested: Baseline, Policy-v1 (initial LLM policies), and Policy-v2 (fine-tuned). Each scenario is simulated over the same traffic demand. All configurations such as network geometry, detectors, routing, car-following/lane-changing parameters are held constant across scenarios.

### c) Key Performance Indicators

#### Network-level KPIs:

- **Total arrived vehicles** ( $\uparrow$ ): number of vehicles that successfully arrive at their destinations within the simulation horizon.
- **Total teleports** ( $\downarrow$ ): count of forced reinsertions due to deadlocks or routing failures (lower indicates better network stability).
- **Vehicles still running at end** ( $\downarrow$ ): vehicles remaining on the network at  $t = \text{end}$  (proxy for clearance).
- **Vehicles waiting at end** ( $\downarrow$ ): vehicles in a waiting state at termination (proxy for residual queues).
- **Average mean speed over steps (m/s)** ( $\uparrow$ ): time-averaged network mean speed.
- **Mean travel time at end (s)** ( $\downarrow$ ): average door-to-door travel time of completed trips.
- **Average mean waiting time over steps (s)** ( $\downarrow$ ): time-averaged network mean waiting time.

#### Detector-level KPIs:

- **Average Flow Baseline** ( $\uparrow$ ): Baseline network average flow per detector (veh/h).
- **Average Flow Policy-v1** ( $\uparrow$ ): Policy-v1 network average flow per detector (veh/h).

- **Average Flow Policy-v2** ( $\uparrow$ ): Policy-v2 network average flow per detector (veh/h).

### d) Success Criteria

A policy is considered beneficial if it achieves higher *Total arrived*, *Average mean speed*, *Average Flow Policy-v2*, and lower *Total teleports*, *Vehicles still running at end*, *Vehicles waiting at end*, *Mean travel time*, *Average mean waiting time*. The highest-performing policy proceeds to discussion.

## V. RESULT

### A Data & Split Summary

This subsection summarizes the key characteristics of the raw and processed traffic dataset used for model training and evaluation. It highlights the data volume, structure, cleaning process, and temporal split adopted to ensure reliable prediction performance.

	Detector_ID	Lane	Volume	Direction
DateTime				
2023-01-01	2001	1	31	L
2023-01-01	2001	2	13	R
2023-01-01	2001	3	0	S
2023-01-01	2001	4	154	S
2023-01-01	2001	5	267	S
2023-01-01	2001	7	0	L
2023-01-01	2001	8	0	S
2023-01-01	2001	9	206	S
2023-01-01	2001	10	208	S

Fig. 5. Dataframe Head before processing

Figure 5 shows the first 10 rows of the dataframe before processing, including timestamp, detector id, lane id, vehicle count and direction. In total there is 8,918,470 entries. Direction is encoded as L for left, R for right, S for straight. It can be also encoded combination of direction to represent more complex maneuvers, such as 'LS' for left and straight or 'RS' for right and straight.

Figure 6 shows the daily volumes for sites 2057 and 2909. As shown in the figure, there is a lot of missing data, which may be due to site construction or maintenance. These are only two examples among the 40 sites. Similar patterns can be observed at other sites as well. Therefore, any day with continuous zero counts is removed from the dataset. After performing data cleaning, the final dataset contains 7,863,406 rows. Around 12 percent of data is removed after cleaning.

Figure 7 shows the direction distribution for the whole dataset. Straight movements have the highest volume, followed by right and left turns. The network also includes combined movements (SL, SR, SLR, and LR). This suggests that most roads in Auckland CBD are designed for through traffic, with fewer lanes for turning.

Figure 8 shows the one-week traffic pattern for site 2906, lane 3. The figure shows clear daily patterns, with peak traffic in the morning and a smaller evening peak on weekdays,

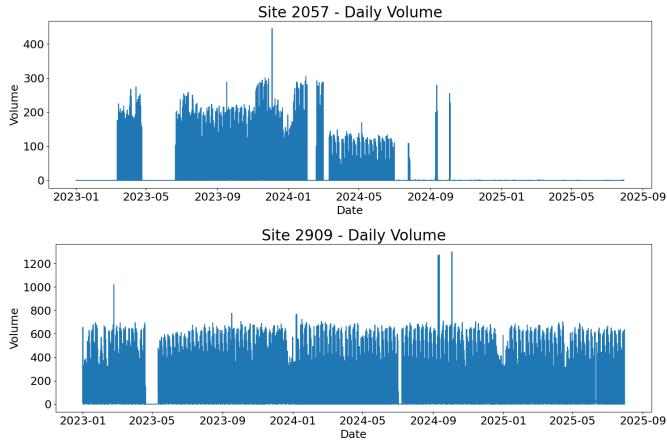


Fig. 6. Site 2057 & 2909 Daily Volume

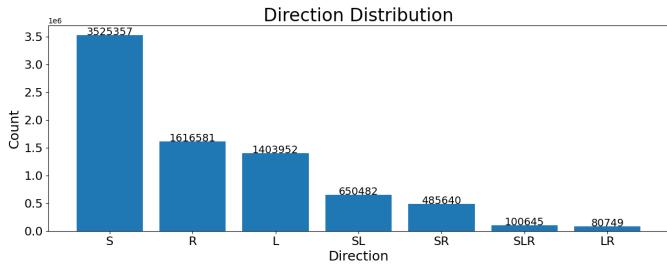


Fig. 7. Direction Distribution for the whole dataset

and overall lower volumes on the weekend (2025-07-05 and 2025-07-06). These patterns show the temporal pattern that the model can learn to improve prediction accuracy.

Figure 9 shows the train test split. The training set contains data from 2023-01-01 to 2024-10-01, while the test set contains data from 2024-10-01 to 2025-07-31. This temporal split ensures that the model is evaluated on future data, simulating real-world prediction scenarios.

In summary, the final dataset captures spatial and temporal variations across Auckland CBD, providing sufficient diversity for model generalisation and performance evaluation in subsequent sections.

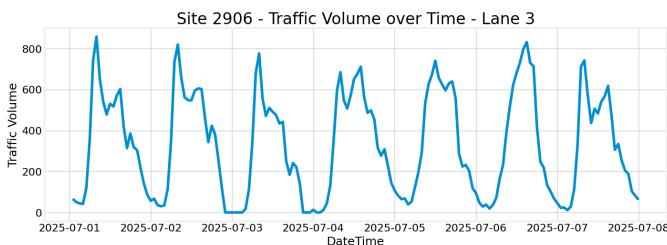


Fig. 8. Site 2906 Lane 3 One week traffic pattern

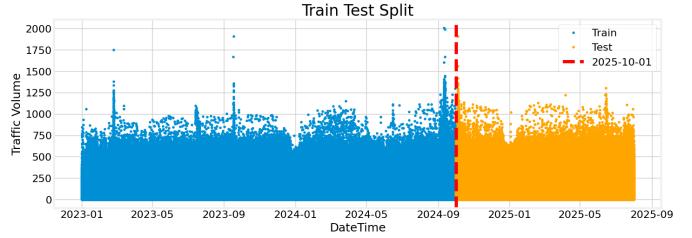


Fig. 9. Train Test Split

## B Prediction Performance

This subsection presents the performance of the XGBoost model in predicting hourly traffic volumes at lane-level detectors within Auckland CBD. Key evaluation metrics are reported for both training and test sets, along with visualizations comparing predicted and actual traffic volumes.

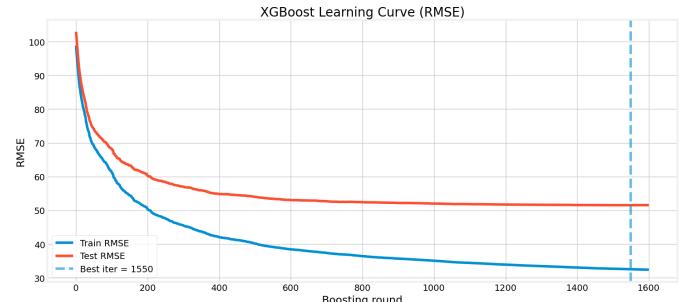


Fig. 10. XGBoost Learning Curve

TABLE VI  
PREDICTION PERFORMANCE

Model (Test)	MAE (veh/h)	RMSE (veh/h)	$R^2$
XGBoost	26.38	51.53	0.755

Figure 10 shows the learning curve of the XGBoost model during training. The training and test RMSE decrease steadily over iterations, indicating effective learning without significant overfitting. Early stopping was triggered after 50 rounds of no improvement on the test set, resulting in a final model with 1550 estimators.

Table VI shows the other prediction metrics of the trained model. The XGBoost model achieved  $R^2$  of 0.755 on the test set, indicating that it explains approximately 75.5% of the variance in traffic volumes. The MAE of 26.38 veh/h suggests that, on average, predictions deviate from actual counts by about 26 vehicles per hour. These results are sufficient to support downstream explainability and policy generation.

Table VII lists the top 10 best-performing detectors based on RMSE. Detector 2038 achieved the lowest RMSE of 24.98 veh/h and highest  $R^2$  of 0.839, indicating strong predictive accuracy at this site. Other top detectors also show low

TABLE VII  
BEST-PERFORMING DETECTORS RANKED BY LOWEST RMSE.

Detector ID	MAE (veh/h)	RMSE (veh/h)	$R^2$
2038	15.89	24.98	0.839
2037	14.60	25.15	0.775
2243	16.27	26.99	0.829
2077	16.38	27.07	0.815
2393	19.30	29.88	0.807
2102	16.95	31.22	0.793
2040	17.91	31.29	0.763
2008	20.11	32.12	0.665
2902	20.42	32.81	0.889
2094	22.73	35.10	0.823

MAE and high  $R^2$  values, demonstrating consistent model performance across multiple locations.

TABLE VIII  
WORST-PERFORMING DETECTORS RANKED BY HIGHEST RMSE.

Detector ID	MAE (veh/h)	RMSE (veh/h)	$R^2$
2130	156.85	252.97	-0.009
2134	100.07	139.40	-0.455
2185	54.07	107.57	0.292
2025	54.07	89.93	0.505
2020	47.74	77.39	0.376
2005	59.92	73.20	-0.727
2001	33.47	63.22	0.558
2052	27.18	63.03	0.596
2068	38.52	60.74	0.848
2019	41.52	54.78	-0.840

Table VIII lists the 10 worst-performing detectors based on RMSE. Detector 2130 exhibited the highest RMSE of 252.97 veh/h and a negative  $R^2$  of -0.009, indicating poor predictive accuracy at this site. Other low-performing detectors also show high MAE and negative or low  $R^2$  values, suggesting challenges in modeling traffic patterns at these locations.

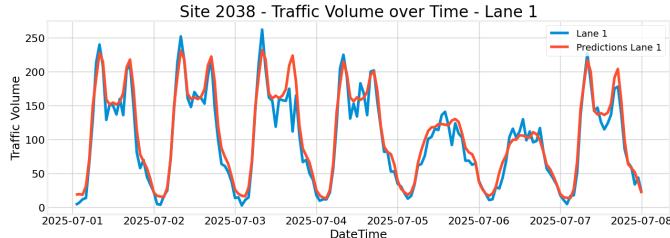


Fig. 11. Best Prediction Site

Figure 11 shows the best prediction site. As shown in the figure, the model predictions closely follow the ground truth traffic volumes, capturing both peak and off-peak patterns accurately.

Figure 12 shows the worst prediction site. For further investigation, figure 13 can provide more insight. The sudden

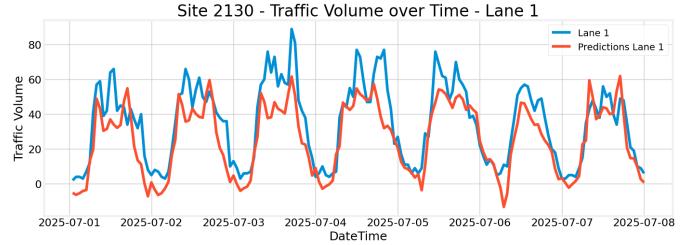


Fig. 12. Worst Prediction Site

level shifts in traffic volume, moving from mostly below 400 vehicles/day to 600–800 vehicles/day thereafter, suggesting detector recalibration, lane reconfiguration, or road works that introduce non-stationarity, and high noise with sporadic spikes exceeding 1000 vehicles/day, likely due to sensor errors or abnormal events. Because the underlying data from Site 2130 are highly inconsistent, with sudden level shift and high sporadic spikes, the predictive model cannot reliably capture its temporal dependencies. Consequently, the residual variance exceeds the total variance, leading to negative  $R^2$  and identifying this detector as one of the worst-performing sites.

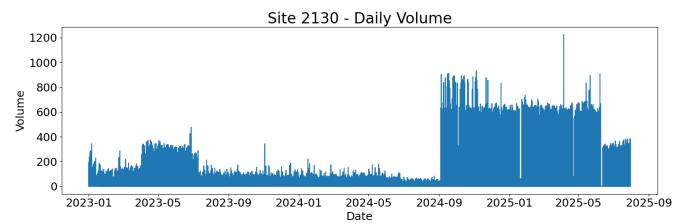


Fig. 13. Daily volume for worst prediction site (2130)

Figure 14 shows the distribution of  $R^2$  scores across all detectors. Most detectors achieved  $R^2$  values between 0.6 and 0.9, indicating that the model explained most portion of the traffic. However, a small number of detectors exhibit negative  $R^2$  values, suggesting that the model performed worse than the baseline. These outliers typically correspond to detectors with unstable or inconsistent traffic readings, as discussed in figure 13.

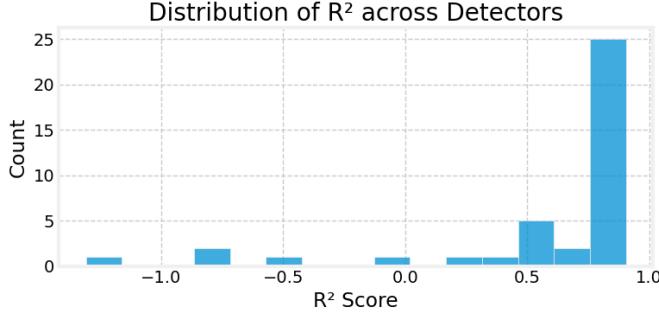


Fig. 14. Distribution of  $R^2$  across detectors.

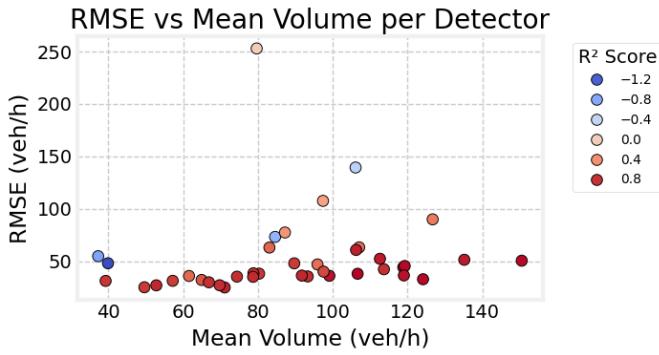


Fig. 15. Relationship between RMSE and mean volume across detectors.

Figure 15 further explores the relationship between model error and average traffic intensity. Usually detectors with higher mean volumes tend to have higher RMSE values, reflecting the natural scaling of error magnitude with traffic flow levels. However, most detector tend to have low RMSE values across different mean volume. Additionally, their  $R^2$  values remain generally high, indicating that predictive accuracy remains proportionally consistent across both high and low volume detectors. This analysis suggests that model errors are primarily influenced by traffic scale and data quality rather than systematic bias.

Overall, the XGBoost model demonstrated strong predictive performance across most detectors. Despite a few low-performing sites, the model's overall accuracy and consistency are sufficient to support subsequent explainability analysis and simulation-based policy evaluation.

### C Explainability Findings

This section present the key insights derived from the SHAP explainability analysis of the XGBoost traffic volume prediction model. Both global and local interpretations are discussed, highlighting the most influential temporal and spatial features affecting predictions with the help of the SHAP library. Additionally, congestion hotspot identification based on predicted volumes is presented.

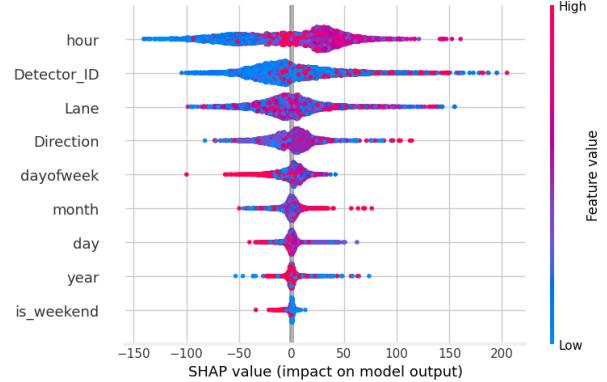


Fig. 16. Global SHAP summary: most influential features across all detectors.

Figure 16 presents the global SHAP summary for the XGBoost model, ranking the overall importance of features across all detectors. The horizontal spread of SHAP values indicates feature significance, the wider spreads reflect greater variability in feature contribution and hence stronger predictive influence. Temporal variables such as hour, Detector ID, and lane show the highest SHAP magnitudes, confirming that traffic volume is strongly influenced by time-of-day patterns and detector location. This reflects expected behaviour, as traffic increases during peak hours and different sites contribute varying volumes. Spatial attributes, including lane and direction, further adjust predicted intensity according to lane capacity and traffic flow direction.

TABLE IX  
TOP 5 MOST CONGESTED DETECTORS BY AVERAGE UTILISATION (UTIL\_AVG).

Rank	Detector ID	Util. Avg	Severity
1	2906	1.220838	Severe
2	2905	1.009348	Severe
3	2025	0.885252	Severe
4	2118	0.912585	Severe
5	2097	0.808090	Severe

To further explain why detector location is so influential, congestion hotspot analysis was performed using predicted traffic volumes. Table IX lists the top 5 most congested detectors based on average utilisation during peak hours. Detector 2906 exhibited the highest average utilisation of 1.22, indicating severe congestion as demand exceeds capacity. Other detectors also show high utilisation ratios, confirming persistent bottlenecks in the network.

Figure 17 visualises congestion hotspots across the Auckland CBD using a heatmap representation. Areas with high utilisation ratios are highlighted in red, indicating locations where traffic demand consistently exceeds roadway capacity. The map shows that detectors located near the perimeter of the CBD experience higher congestion levels than those in the city centre. Sites such as 2118, 2905, and 2902 display particularly intense utilisation, likely because they serve as primary entry and exit points for vehicles travelling to and from the city.

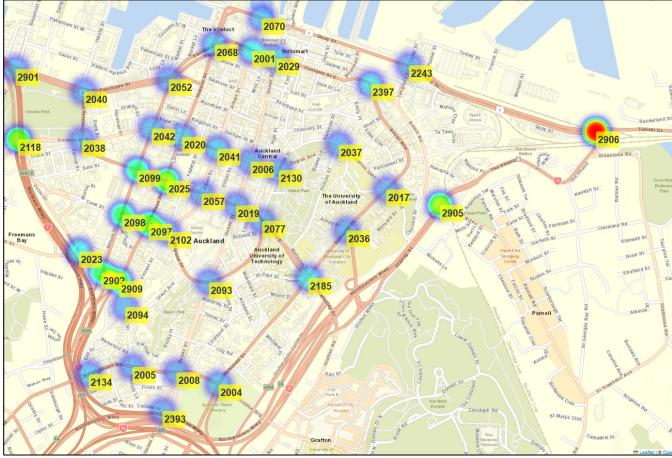


Fig. 17. Congestion heatmap of Auckland CBD based on capacity utilisation

For further analysis, local SHAP explanations were generated for Detector 2906, identified as the most congested site in the network.

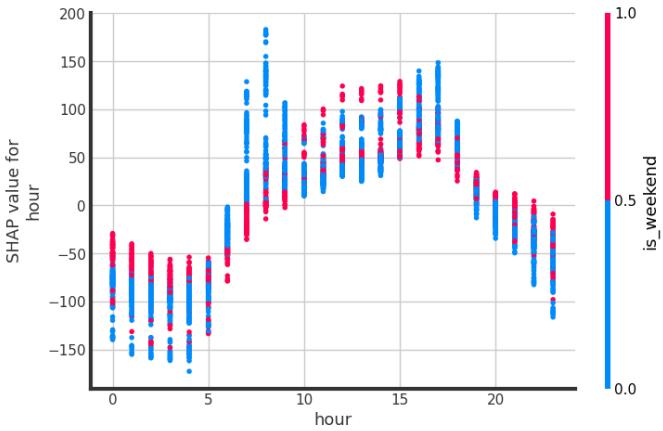


Fig. 18. Dependence plot for hour feature showing its impact on traffic volume predictions for site 2906

Figure 18 shows the dependence of SHAP values on the hour feature for Site 2906, with colour representing `is_weekend`. Positive SHAP values between 07:00 and 10:00 and again around 17:00 indicate that these hours substantially increase predicted traffic volumes, consistent with the figure 8 shown in earlier section. Weekend points cluster nearer to zero, reflecting reduced commuting demand. This demonstrates that the model correctly captures rush-hour dynamics at the congested site.

Figure 19 presents the local SHAP dependence plot for the lane feature at Site 2906, complemented by the SCATS intersection layout shown in Figure 20. Lanes 2 and 3 exhibit the highest positive SHAP values, indicating that they contribute most strongly to the predicted traffic volume. As seen in the SCATS layout, these lanes correspond to the main movements

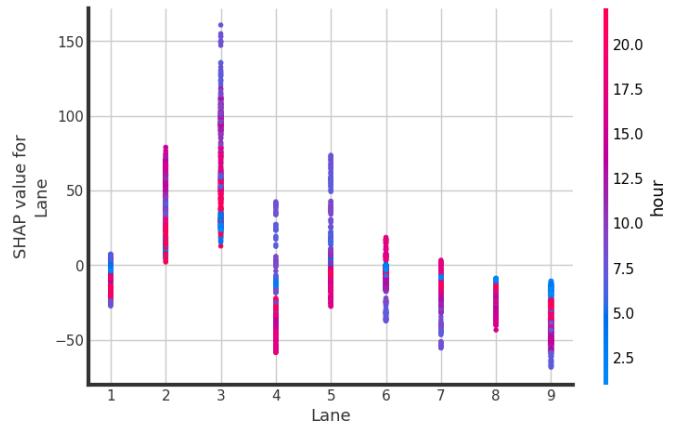


Fig. 19. Dependence plot for lane feature showing its impact on traffic volume predictions for site 2906



Fig. 20. SCATS intersection layout for site 2906

along Tamaki Drive and The Strand Street. Conversely, lanes 6–9 show near-zero or negative SHAP values, reflecting their roles with lower flow rates. The colour gradient, representing the hour feature, shows that the positive influence of lanes 2 and 3 intensifies, aligning with the observed morning peak. This are important information that can guide LLM policy generation to prioritise signal timing adjustments for these critical lanes.

Figure 21 presents the SHAP waterfall plot for a high congestion prediction at Site 2906 during the morning peak hour (08:00). The baseline prediction ( $E[f(X)] = 87.6 \text{ veh/h}$ ) is sequentially increased by strong positive contributions from key features such as `Detector_ID` (+263.2  $\text{veh/h}$ ), `hour` (+183.0  $\text{veh/h}$ ), `lane` (+160.7  $\text{veh/h}$ ), and `direction` (+128.4  $\text{veh/h}$ ). The direction feature was numerically encoded from categorical movement labels ('L', 'LR', 'R', 'S', 'SL', 'SLR', 'SR' → 0–

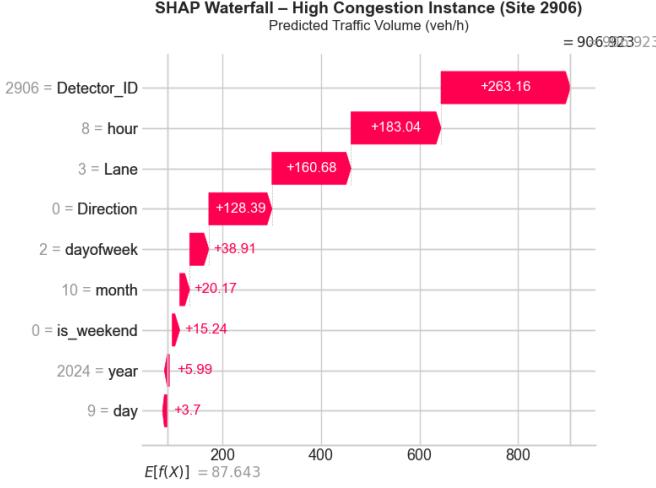


Fig. 21. Waterfall plot showing local SHAP explanations for a high congestion prediction at site 2906

6). These features collectively represent inbound traffic along the turning left lanes of Tamaki Drive, corresponding to the most congested movement pattern at this intersection. Smaller temporal factors dayofweek, month, and is\_weekend further reinforce the prediction. The final prediction of approximately 906 veh/h.

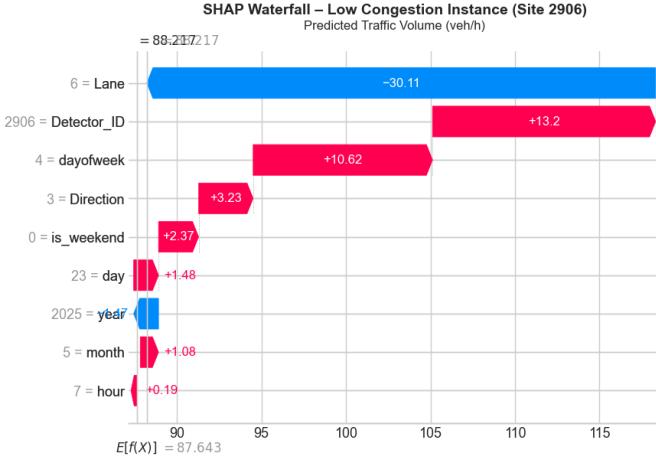


Fig. 22. Waterfall plot showing local SHAP explanations for a low congestion prediction at site 2906

Figure 22 illustrates a low congestion prediction for the same site under off-peak conditions. Starting from the same baseline ( $E[f(X)] = 87.6$  veh/h), the model's output remains near 88 veh/h due to a dominant negative contribution from lane 6 (-30.1 veh/h), indicating a low utilisation lane. Minor positive influences from Detector\_ID (+13.2 veh/h), dayofweek (+10.6 veh/h), and direction (+3.2 veh/h) slightly offset this reduction. The small feature contributions and balanced positive, negative effects reveal that the model appropriately

suppresses volume predictions.

Overall, the SHAP explainability analysis confirms that the XGBoost model captures realistic spatio-temporal dynamics of urban traffic flow. By focusing on the most congested site (Detector 2906), the model explanations not only validate predictive behaviour but also provide interpretable evidence to support policy generation and simulation adjustments in the following section. In the next section, focusing on generated LLM traffic policies based on the detector site 2906.

## D Policy Proposals

This section outlines the policy generation process, which translates interpretable model findings into actionable traffic management strategies. Using the SHAP insights derived from the most congested site (Detector 2906), the study employs a LLM to assist generating traffic policies. The generated policies are then validated using SUMO traffic simulation to quantify their performance impacts. As discussed in the Implementation section, the policy generation pipeline consists of several key steps: input preparation, prompt construction, LLM policy generation, policy application, re-simulation, and iterative fine-tuning.

The LLM was guided using structured prompts containing the network layout containing existing traffic configurations, XAI evidence, detector-level statistics. The prompt explicitly instructed the model to propose feasible SUMO-compatible <t1Logic> updates, following predefined constraints such as cycle length, yellow interval, and coordination offsets for adjacent intersections. This ensured that all generated policies adhered to operational and safety standards while remaining explainable and reproducible.

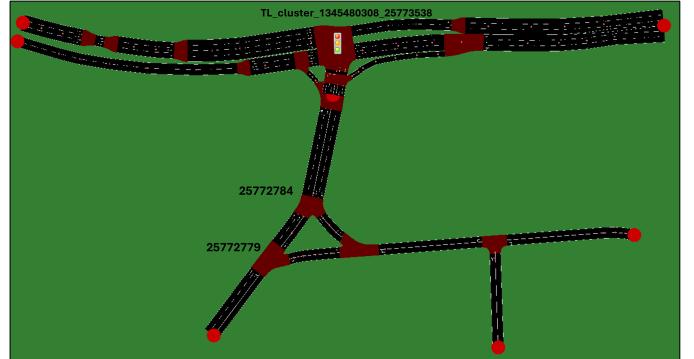


Fig. 23. Original network layout for site 2906 before policy application

The first set of LLM-generated policies (V1) turns congestion insights and SHAP-derived feature importance into signal-control strategies. As detailed in the structured JSON output (see appendix A), the LLM identifies problematic junctions, explains the observed congestion, and proposes targeted adjustments to their control logic. For example, at the main Quay Street intersection (cluster\_1345480308\_25773538), it detects a phase imbalance causing eastbound spillback and recommends reallocating green time from the Tamaki

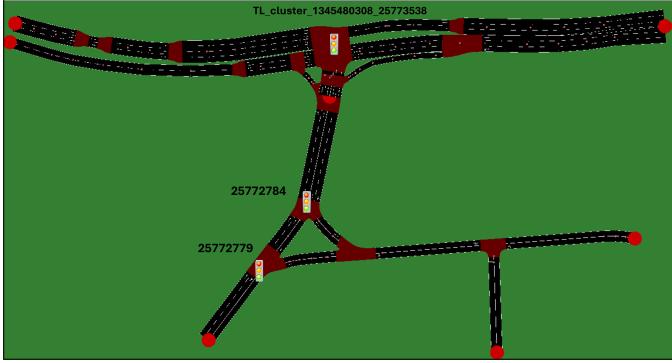


Fig. 24. Modified network layout for site 2906 after policy application

Drive approach to eastbound movements while keeping a 90 s cycle. At The Strand × Gladstone (25772784) and the upstream Port-end junction (25772779), where congestion was previously unmanaged, it proposes installing new actuated two-phase signals with explicit yellow and all-red intervals to meter merging traffic and reduce queue spillback. The detector level evidence such as low speeds, high occupancies aligns with XAI indicators especially the dominance of hour and lane features which justify peak hour retiming. The output includes SUMO compatible `<tlLogic>` XML elements, enabling direct integration into the simulation for validation. The modified network layout after applying V1 policies is shown in Figure 24, compared with the original layout in Figure 23.

TABLE X  
STRUCTURAL COMPARISON BETWEEN VERSION 1 (V1) AND VERSION 2 (V2) LLM-GENERATED TRAFFIC-SIGNAL CONFIGURATIONS.

Junction / ID	Version 1 (V1)	Version 2 (V2)
TL_cluster_1345480308_25773538	Fine-tuned existing actuated signal; reallocated green from Tamaki Dr to Quay St EB; maintained 90 s cycle.	Further refined timing; shortened total cycle ( $\approx 69$ s); added corridor offset (9 s) for progression along The Strand–Quay EB.
25772784	New actuated signal added at The Strand × Gladstone; two-phase logic controlling main vs side movements.	Retimed and integrated into coordinated corridor (offset = 2 s); added six-phase detailed logic and increased NB through green.
25772779	New actuated two-phase signal at upstream The Strand junction; favoured 624523286 → 606859931 throughs.	Re-timed mainline green, kept minor phases; set as corridor base (offset = 0 s) for downstream coordination.

The second iteration of the LLM focuses on refining the timing of the traffic signal to improve traffic flow. (see Appendix B) Version 2 (V2) of the LLM-generated policies proposed refining local timings and coordinating a 40 km/h green wave across three junctions—25772779, 25772784, and cluster\_1345480308\_25773538. For

TABLE XI  
TIMING DIFFERENCES BETWEEN VERSION 1 (V1) AND VERSION 2 (V2)  
LLM-GENERATED SIGNAL PLANS.

Junction / ID	Offset (s)	Cycle (s)	Phase Durations (s)	Key Change
TL_cluster_1345480308_25773538	V1 10 → V2 9	90 → 69	(28, 3, 22, 3, 31, 3) → (27, 3, 36, 3)	Simplified from 6 to 4 phases; merged minor movements; reallocated 12 s green to Quay St EB.
25772784	V1 0 → V2 2	64 → 74	(35, 4, 1, 20, 4, 1) → (30, 4, 1, 35, 4, 1)	Extended NB through +15 s; introduced coordination offset (2 s).
25772779	V1 0 → V2 0	66 → 76	(40, 4, 1, 18, 4, 1) → (40, 3, 30, 3)	Simplified to 4 phases; extended mainline green for smoother progression.

Quay Street, the LLM identifies severe eastbound oversaturation (occupancy = 92.6%) and shifts green splits toward this approach while trimming the opposing phase. At the intermediate and upstream intersections, it extends mainline greens by 15 s and applies offsets (0 s, 2 s, 9 s) to sustain platoon progression. Compared with V1, V2 shortens cycle times, simplifies phase structures, and encodes corridor timing via the shared `TheStrand_Quay_EB_Progression` reference. Each change is justified with detector evidence (speed, occupancy, queues) and emitted as SUMO compatible `<tlLogic>` definitions for direct simulation-based validation. Tables X and XI summarise structural and timing differences between policies V1 and V2, highlighting refinements from iterative fine-tuning.

## E Simulation Outcomes

This subsection evaluates the effects of the LLM-generated signal control policies on overall and per-detector traffic performance. Each scenario Baseline, Policy-V1, and Policy-V2 was simulated under identical demand conditions. Key network-level indicators are summarised in Table XII, while Figures 25 and 26 visualise detector-level flow variations across the road network.

In table XII, both Policy-V1 and Policy-V2 demonstrate improvements over the Baseline across multiple KPIs. Total arrived vehicles increase from 1647 to 1920 (+16.6%) under V1 and further to 2276 (+38.2%) with V2, indicating enhanced throughput. Total teleports drop dramatically from 58 to 6 with V1 and to zero with V2, reflecting improved network stability and reduced congestion rerouting. Average mean speed rises from 1.32m/s at Baseline to 1.64m/s (+24.2%) with V1 and 2.22m/s (+68.2%) with V2, signifying smoother traffic flow.

TABLE XII  
SIMULATION RESULTS FOR BASELINE, POLICY-V1, AND POLICY-V2

Metric	baseline	v1	v2
Simulation end time (s)	3599.0	3599.0	3599.0
Total arrived vehicles	1647.0	1920.0	2276.0
Total teleports	58.0	6.0	0.0
Vehicles still running at end	129.0	120.0	97.0
Vehicles waiting at end	3916.0	3652.0	3319.0
Avg. mean speed over steps (m/s)	1.32	1.64	2.22
Mean travel time at end (s)	222.94	196.26	166.44
Avg. mean waiting time over steps (s)	359.35	459.40	408.80
# summary steps	3600.0	3600.0	3600.0
Detector rows	996.0	996.0	996.0
Detector count	19.0	19.0	19.0
Detector window seconds	3600.0	3600.0	3600.0

Mean travel time decreases from 222.94s to 196.26s (-12.0%) with V1 and 166.44s (-25.3%) with V2, indicating faster journeys. Although average mean waiting time increases slightly with V1 (+27.8%), it reduces with V2 (-13.9%) compared to V1, suggesting that refined traffic signal timing effectively mitigates delays.

TABLE XIII  
SUMMARY OF KEY NETWORK-LEVEL IMPROVEMENTS

Metric	Baseline	V1 vs Base	V2 vs Base
Total arrived vehicles	1647	+16.6%	+38.2%
Total teleports	58	-89.7%	-100%
Vehicles still running at end	129	-7.0%	-24.8%
Vehicles waiting at end	3916	-6.7%	-15.2%
Average mean speed (m/s)	1.32	+24.2%	+68.2%
Mean travel time (s)	222.94	-12.0%	-25.3%
Average mean waiting time (s)	359.35	+27.8%	+13.7%*

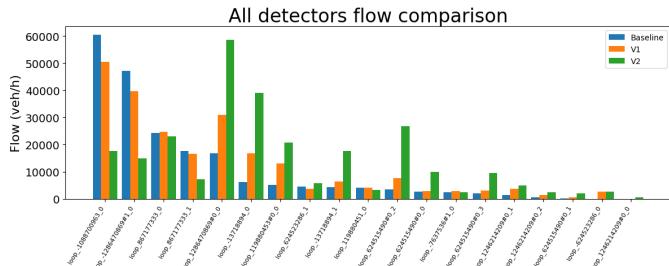


Fig. 25. Detector flow comparison across baseline, Policy-v1, and Policy-v2 for site 2906

Figure 25 presents the flow comparison across all detectors under Baseline, Policy-V1, and Policy-V2. While V1 achieves moderate improvements over the baseline through local signal retiming, the V2 produces a substantial increase in overall throughput. Most detectors record higher flow under V2, confirming that the LLM-generated coordination successfully improved platoon progression and released congestion bottlenecks. However there are few detectors showing decreased

flow, which may be due to introduced new traffic light phases in the road network or reallocated green time.

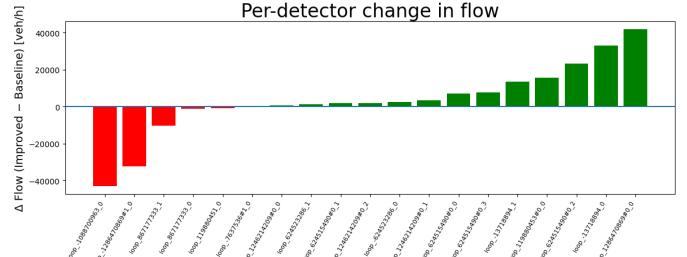


Fig. 26. Change in detector flow from baseline to Policy-v2 for site 2906

Figure 26 illustrates the change in detector flow from Baseline to Policy-V2. The majority of detectors exhibit positive flow changes. This indicates that the changing signal timing effectively improved congestion and enhanced throughput across the network. A few detectors show slight decreases in flow, likely due to new green time or new phase structures introduced by the LLM policies. Overall, the positive flow changes at most detectors confirm the feasibility of the LLM-generated traffic management strategies.

## F Discussion

The simulation outcomes demonstrate that integrating XAI with data-driven policy generation can deliver measurable improvements in traffic performance across the Auckland CBD network. The XGBoost model together with SHAP feature interpretation and LLM policy generation, achieved better performance compared to the baseline SUMO configuration. Comparison showed a mean speed increase of +68.2% and total arrival growth of +38.2%, confirming that explainable policy refinements can translate into tangible congestion reduction.

From a planning perspective, these results highlight the potential of XAI-informed decision systems to complement traditional transport modelling. Policy makers such as AT can use explainable models to continuously learn and adapt signal policies based on traffic flow behaviour. By identifying interpretable factors such as lane utilisation, congestion patterns, the proposed framework allows policy makers to understand why a policy performs well.

Beyond the Auckland CBD, the proposed framework can be generalised to other urban cities. The combination of SHAP interpretability and LLM-generated policies establishes a replicable pipeline to build data to decision infrastructures. Such automation ensures that optimisation processes remain transparent, auditable, and justifiable within public-sector governance, which is an important requirement for critical system such as ITS.

## VI. CONCLUSION AND FUTURE WORK

This paper demonstrates an end-to-end pipeline that links traffic prediction, explainability, policy generation, and simulation for urban traffic management, using Auckland CBD as a

case study. The XGBoost model achieved strong predictive accuracy, while SHAP provided interpretable feature attributions. Leveraging these insights, an LLM-assisted policy framework produced SUMO-compatible signal plans that were iteratively refined and validated in simulation. Key performance indicators confirmed that the LLM-generated policies improved network throughput, reduced congestion, and increased travel speeds relative to the baseline, showing that XAI can effectively inform actionable traffic management strategies.

#### A Linking Research Questions to Results

**RQ3:** How effective are LLM-generated policy recommendations on real-world traffic performance when tested in SUMO?

Simulation results indicate that LLM-generated recommendations can substantially enhance performance. Relative to the Baseline, Policy-V2 increased total arrivals by **+38.2%**, eliminated teleports, raised average mean speed by **+68.2%**, and reduced mean travel time by **-25.3%**. These gains indicate that the generated policies successfully addressed bottlenecks and improved flow.

The case study yields several additional findings:

- **Explainability matters.** SHAP insights offered clear, actionable evidence that guided the LLM to propose targeted signal-timing adjustments. Combined with detector statistics and network layout, this produced feasible, high-impact policies.
- **Iterative refinement is crucial.** Policy-V1 produced mixed outcomes, but simulation guided tuning led to Policy-V2 with consistent and substantial improvements. This highlights the importance of validating and refining AI-generated strategies.
- **Green-wave coordination helps.** Policy-V2 introduced coordinated offsets (a green-wave) across multiple intersections, materially improving traffic flow, in line with prior literature [14].

#### B Limitations

Despite promising results, several limitations remain:

- **Data quality.** The dataset contained missing values and anomalies that may affect model performance. Time-series forecasting models are sensitive to data quality, and imputation can introduce bias.
- **Network scope.** The Auckland CBD network is complex. Due to time and compute constraints, policies targeted a limited set of junctions and may not capture wider network interactions.
- **Calibration and transferability.** SUMO calibration may not fully capture real-world behaviour. Further calibration and field validation are required to strengthen external validity.

#### C Future Work

- **Richer features.** Integrate weather, events, transit schedules, and connected-vehicle feeds, incorporate multimodal demand (pedestrians, cycling) to improve predic-

tion, extend explainability insights and broaden policy options.

- **Larger networks.** As this study focused on a limited junction, future work could expand to larger networks with more intersections to capture broader interactions, and test the LLM capabilities in handling complex scenarios.

By connecting prediction, explainability, LLM-based policy generation, and simulation, this work shows how AI can move beyond forecasting to actionable and auditable traffic management. The Version-2 policies demonstrate that explainable, LLM-assisted design can deliver meaningful network improvements, while the identified limitations outline a practical path toward robust, real-world deployment.

#### VII. ACKNOWLEDGEMENT

The author would like to express sincere gratitude to the supervisor, Matthew Kuo., for his continuous guidance and constructive feedback throughout this research. Appreciation is also extended to Auckland Transport for providing access to the SCATS dataset used in this study, and to the Auckland University of Technology for the academic and technical support that made this work possible.

The author has made all source code and related documentation publicly available at GitHub Repository.

## LIST OF ACRONYM

<b>AI</b>	Artificial Intelligence. 1, 5, 11, 24	<b>ML</b>	Machine Learning. 1–3, 7, 11, 12
<b>AIMSUN</b>	Advanced Interactive Microscopic Simulator for Urban and Non-urban Networks. 2, 7	<b>NZTA</b>	New Zealand Transport Agency. 12
<b>ARIMA</b>	Autoregressive Integrated Moving Average. 1, 3, 5, 9, 10	<b>OSM</b>	OpenStreetMap. 6
<b>AT</b>	Auckland Transport. 12, 13, 23	<b>RF</b>	Random Forest. 5, 9
<b>CBD</b>	Central Business District. 2, 12–14, 17, 19, 23, 24	<b>RMSE</b>	Root Mean Squared Error. 5, 7, 11, 17–19
<b>CNN</b>	Convolutional Neural Network. 7, 8	<b>RNN</b>	Recurrent Neural Network. 8
<b>CORSIM</b>	Corridor microscopic Simulation. 7	<b>SARIMA</b>	Seasonal Autoregressive Integrated Moving Average. 3
<b>DL</b>	Deep Learning. 1–3, 7, 8, 11	<b>SHAP</b>	Shapley Additive Explanations. 1–3, 5, 7–9, 11, 13–16, 19–21, 23, 24
<b>GAI</b>	Generative Artificial Intelligence. 2	<b>SLR</b>	Systematic Literature Review. 2
<b>GCA</b>	Generally Capable Agent. 4, 9, 11	<b>SSA</b>	Sparrow Search Algorithm. 3, 9, 11
<b>GCN</b>	Graph Convolutional Network. 8	<b>ST-LLM</b>	Spatio-Temporal Large Language Model. 4, 11
<b>GNN</b>	Graph Neural Network. 5, 8, 9	<b>SUMO</b>	Simulation of Urban Mobility. 1, 2, 4, 6, 7, 9, 11, 12, 14, 15, 21–24
<b>GRU</b>	Gated Recurrent Unit. 5, 8	<b>SVM</b>	Support Vector Machine. 5, 7, 11
<b>HI-XGB</b>	High-Importance Links with XGBoost. 5, 7, 11	<b>TraCI</b>	Traffic Control Interface. 6
<b>ITS</b>	Intelligent Traffic Management System. 1, 7–9, 11, 12, 23	<b>TRANSIMS</b>	Transportation Analysis and Simulation System. 7
<b>kNN</b>	k-Nearest Neighbours. 5, 9	<b>VISSIM</b>	Verkehr In Städten – SIMulation (PTV microscopic traffic simulator). 2, 7
<b>LIME</b>	Local Interpretable Model-Agnostic Explanations. 1, 5, 9	<b>XAI</b>	Explainable Artificial Intelligence. 1, 2, 5, 8, 9, 11–14, 21–24
<b>LLM</b>	Large Language Model. 1–4, 9, 11, 14–16, 20–24	<b>XGBoost</b>	Extreme Gradient Boosting. 1–3, 5, 7, 9, 11, 13, 14, 17, 19, 21, 23, 24
<b>LSTM</b>	Long Short-Term Memory. 1, 3–5, 8–12	<b>ZS-CoT</b>	Zero-Shot Chain of Thought. 4, 9, 11
<b>MAE</b>	Mean Absolute Error. 5, 7, 10, 11, 17, 18		
<b>MAPE</b>	Mean Absolute Percentage Error. 7, 8, 11		
<b>MATSim</b>	Multi-Agent Transport Simulation. 2, 6, 7		

## REFERENCES

- [1] R. A. Khalil, Z. Safelnasr, N. Yemane, M. Kedir, A. Shafiqurrahman, and S. N, “Advanced learning technologies for intelligent transportation systems: Prospects and challenges,” *IEEE Open Journal of Vehicular Technology*, vol. 5, pp. 397–427, 2024, ISSN: 2644-1330. DOI: 10.1109/OJVT.2024.3369691.
- [2] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, and B. Yin, “Deep learning on traffic prediction: Methods, analysis, and future directions,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4927–4943, 2022, ISSN: 1558-0016. DOI: 10.1109/TITS.2021.3054840.
- [3] C. S. Priya and F. S. Francis, “Optimizing traffic speed prediction using a multi-objective genetic algorithm-enhanced rnn for intelligent transportation systems,” *IEEE Access*, vol. 13, pp. 35 688–35 706, 2025, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2025.3544864.
- [4] D. H. Shin, K. Chung, and R. C. Park, “Prediction of traffic congestion based on lstm through correction of missing temporal and spatial data,” *IEEE Access*, vol. 8, pp. 150 784–150 796, 2020, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.3016469.
- [5] A. Khan, M. M. Fouda, D. T. Do, A. Almaleh, and A. U. Rahman, “Short-term traffic prediction using deep learning long short-term memory: Taxonomy, applications, challenges, and future trends,” *IEEE Access*, vol. 11, pp. 94 371–94 391, 2023, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3309601.
- [6] Z. Wang, X. Su, and Z. Ding, “Long-term traffic prediction based on lstm encoder-decoder architecture,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 10, pp. 6561–6571, 2021, ISSN: 1558-0016. DOI: 10.1109/TITS.2020.2995546.
- [7] R. Verma, S. Pandey, and S. Rup, “Traffic flow prediction using lstm, arima & sarima model for intelligent transportation system,” in *2024 International Conference on Intelligent Computing and Sustainable Innovations in Technology (IC-SIT)*, 2024, pp. 1–6. DOI: 10.1109/IC-SIT63503.2024.10862828.
- [8] Y. Zheng, W. Ye, R. Yin, Y. Yin, and X. Dai, “Comparative analysis of traffic flow prediction based on xgboost before and after optimization,” in *2024 4th International Conference on Communication Technology and Information Technology (ICCTIT)*, 2024, pp. 545–551. DOI: 10.1109/ICCTIT64404.2024.10928357.
- [9] Y. Zhang, “A novel multi-objective lstm framework for enhancing operational efficiency of yellow taxis in new york city using big data and regional clustering analysis,” in *2024 6th International Conference on Frontier Technologies of Information and Computer (ICFTIC)*, 2024, pp. 1065–1070. DOI: 10.1109/ICFTIC64248.2024.10913058.
- [10] M. Waqas, S. Abbas, U. Farooq, M. A. Khan, M. Ahmad, and N. Mahmood, “Autonomous vehicles conges-
- tion model: A transparent lstm-based prediction model corporate with explainable artificial intelligence (eai),” *Egyptian Informatics Journal*, vol. 28, p. 100 582, 2024, ISSN: 1110-8665. DOI: 10.1016/j.eij.2024.100582.
- [11] L. Kong, H. Yang, W. Li, Y. Zhang, J. Guan, and S. Zhou, “Traffexplainer: A framework toward gnn-based interpretable traffic prediction,” *IEEE Transactions on Artificial Intelligence*, vol. 6, no. 3, pp. 559–573, 2025, ISSN: 2691-4581. DOI: 10.1109/TAI.2024.3459857.
- [12] A. Vijaya, S. Bhattacharai, L. S. Angreani, and H. Wicaksana, “Enhancing transparency in public transportation delay predictions with shap and lime,” in *IEEE International Conference on Industrial Engineering and Engineering Management*, Export Date: 05 April 2025; Cited By: 0, 2024, pp. 1285–1289. DOI: 10.1109/IEEEM62345.2024.10857000.
- [13] L. Chenxi, Y. Sun, Qianxiong, et al., “Spatial-temporal large language model for traffic prediction,” *2024 25th IEEE International Conference on Mobile Data Management (MDM)*, 2024. DOI: 10.1109/MDM61037.2024.000025.
- [14] T. Yiqing, D. Xingyuan, and L. Yisheng, “Large language model-assisted arterial traffic signal control,” *IEEE Journal of Radio Frequency Identification*, vol. 8, 2024, ISSN: 2469-7281. DOI: 10.1109/JRFID.2024.3384289.
- [15] T. Alghamdi, S. Mostafi, G. Abdelkader, et al., “A comparative study on traffic modeling techniques for predicting and simulating traffic behavior,” *Future Internet 2022, Vol. 14, Page 294*, vol. 14, no. 10, 2022, ISSN: 1999-5903. DOI: 10.3390/fi14100294.
- [16] L. X. Lim, T. Connie, and M. K. O. Goh, “Enhancing traffic analysis and prediction through a hybrid lstm-arima model,” in *2024 International Conference on Information Management and Technology (ICIMTech)*, 2024, pp. 672–677. DOI: 10.1109/ICIMTech63123.2024.10780834.
- [17] H. Wang and B. Zhang, “Research on arima model for short-term traffic flow prediction based on time series,” in *2023 8th International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, vol. 8, 2023, pp. 92–95. DOI: 10.1109/ICIIBMS60103.2023.10347816.
- [18] J. Li, G. Xu, Y. Cheng, and B. Huang, “Short-time traffic flow prediction based on k-means++ and lstm,” in *2022 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom-/SustainCom)*, 2022, pp. 286–293. DOI: 10.1109/ISPA-BDCloud-SocialCom-SustainCom57177.2022.00043.
- [19] M. Mohammad and C. Juyeong, “The crossroads of llm and traffic control: A study on large language models in adaptive traffic signal control,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 26, no. 2, 2025, ISSN: 1524-9050. DOI: 10.1109/TITS.2024.3498735.

- [20] L. Zhonghang, X. Lianghao, T. Jiabin, *et al.*, “Urbangpt: Spatio-temporal large language models — proceedings of the 30th acm sigkdd conference on knowledge discovery and data mining,” *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024. DOI: 10.1145/3637528.3671578.
- [21] E. H. Lee, “Traffic speed prediction of urban road network based on high importance links using xgb and shap,” *IEEE Access*, vol. 11, 2023, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3324035.
- [22] R. Ullah, K. Khattak, Z. H. Khan- P. Eng, M. Ahmad Khan, N. Minallah, and A. Khan, “Vehicular traffic simulation software: A systematic comparative analysis,” *Pakistan Journal of Engineering and Technology*, 2021.
- [23] A. O. Diallo, G. Lozenguez, A. Doniec, and R. Mandiau, “Comparative evaluation of road traffic simulators based on modeler’s specifications: An application to intermodal mobility behaviors,” *Proceedings of the 13th International Conference on Agents and Artificial Intelligence*, 2021. DOI: 10.5220/0010238302650272.
- [24] K. M. Ahmed, S. Muvdi, J. Liu, and M. H. Amini, “A federated learning framework for automated decision making with microscopic traffic simulation,” in *2022 International Conference on Computer Communications and Networks (ICCCN)*, pp. 1–9. DOI: 10.1109/ICCCN54977.2022.9868930.
- [25] L. Xia, “Design of urban road traffic induction algorithm based on dl algorithm,” in *2022 IEEE 2nd International Conference on Electronic Technology, Communication and Information (ICETCI)*, pp. 1213–1216. DOI: 10.1109/ICETCI55101.2022.9832120.
- [26] O. N. H. Saghrje, O. Karan, S. Kurnaz, and A. K. Turkben, “Support vector machines (svm) in traffic prediction for intelligent transportation systems- a comprehension review,” in *2025 7th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (ICHORA)*, pp. 1–7. DOI: 10.1109/ICHORA65333.2025.11016984.
- [27] C. Johny and V. Dahiya, “Machine learning applications in vehicular traffic prediction and congestion control: A systematic review,” in *2022 6th International Conference on Electronics, Communication and Aerospace Technology*, pp. 948–954. DOI: 10.1109/ICECA55336.2022.10009384.
- [28] A. Alahmadi, “Towards data empowered intelligent transportation systems: Use cases and future opportunities,” in *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, pp. 1–6. DOI: 10.1109/I2CT61223.2024.10543425.
- [29] M. F. F. Rahman, N. Zhang, and E. Abdel-Raheem, “Intelligent transportation systems utilizing uavs: Integration with iot and machine learning,” in *2025 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 580–586. DOI: 10.1109/IWCMC65282.2025.11059669.
- [30] D. Avramoni, A. Iovanovici, A. M. Ilienescu, and L. Prodan, “Short term traffic congestion prediction using publically available traffic data: A case study on timisoara,” in *2022 IEEE 20th Jubilee World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pp. 000 121–000 126. DOI: 10.1109/SAMI54271.2022.9780813.
- [31] N. R. Maulida and K. Mutijarsa, “Technology trend of traffic density prediction – a systematic literature review,” in *2021 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, pp. 107–110. DOI: 10.1109/ISITIA52817.2021.9502266.

## A Raw LLM Output

The following listing shows the full raw output generated by the LLM before corridor coordination was applied. This version proposed new signal installations and rebalanced phases based on congestion reasoning.

```

1  {
2    "reasoning": [
3      {
4        "edge_or_junction_id": "cluster_1345480308_25773538",
5        "issue_detected": "Phase imbalance at the main signal starving Quay St eastbound approach  

6          (624515490#0) causing spillback.",
7        "proposed_policy": "Retime the existing actuated tlLogic: reallocate green from Tamaki Dr approach to  

8          Quay St EB movements; keep 90 s cycle, increase greens for linkIndex 3-7 (Quay EB through/right)  

9          .",
10       "justification": "Detectors on Quay St EB approach show extreme congestion: loop_624515490#0_2  

11         avg_speed -0.262 m/s with 95.69% occupancy; loop_624515490#0_0 speed 2.54 m/s, high occupancy.  

12         Simulation-wide avg speed is 1.32 m/s indicating gridlock. XAI shows 'hour' and 'Lane' dominate  

13         flow variance, supporting peak direction retiming."
14     },
15     {
16       "edge_or_junction_id": "25772784",
17       "issue_detected": "Unsignalised multi-leg The Strand x Gladstone node with conflicting merges and  

18         high upstream queues.",
19       "proposed_policy": "Install an actuated 2-phase traffic signal controlling The Strand both directions  

20         vs Gladstone movements; add yellow and all-red intervals.",
21       "justification": "Upstream detectors indicate severe blocking: loop_-13718894_0 speed 0.35 m/s,  

22         92.23% occupancy; loop_-13718894_1 speed 4.05 m/s (High congestion); loop_-7637536#1_0 occupancy  

23         97.18%. Signalization will meter conflicts and clear platoons."
24   },
25   {
26     "edge_or_junction_id": "25772779",
27     "issue_detected": "Priority junction at The Strand (near Port end) with heavy demand from 624523286  

28         and conflicting flows; queues propagate to dead-end.",
29     "proposed_policy": "Install an actuated 2-phase signal favouring 624523286 -> 606859931 straight-  

30         throughs with a shorter minor phase for -7637457/-606859931 -> -624523286; include yellow and all  

31         -red.",
32     "justification": "Downstream occupancy on -624523286_1 is 80.50% with low speeds, indicating  

33         spillback from this node; loop_-7637457_0 and corridor demands suggest metering needed to prevent  

34         blocking. XAI hour dominance implies recurring AM peak platoons that benefit from control."
35   },
36   {
37     "edge_or_junction_id": "-1286470869#1",
38     "issue_detected": "High demand on Gladstone Rd approach to 25772782 with low speed (6.94 m/s) and  

39         elevated occupancy (23.19%).",
40     "proposed_policy": "Rely on the new signals at 25772784 and 25772779 to break downstream blockages;  

41         monitor and consider minor geometric/channelization tweaks or peak hour turn restrictions if  

42         congestion persists.",
43     "justification": "loop_-1286470869#1_0 shows sustained congestion; relieving downstream nodes should  

44         restore progression without costly widening (aligned to budget constraints)."
45   },
46   {
47     "edge_or_junction_id": "-1088700963",
48     "issue_detected": "Strong inflow towards 25773189 (1008 veh/h) with reduced speeds (9.74 m/s)  

49         indicating approach delay at an unsignalised T-junction.",
50     "proposed_policy": "Short-term: keep priority control but add peak-hour right-turn restriction from  

51         Balfour Rd or dynamic give-way signage; reassess after corridor retiming. Medium-term: consider  

52         minor signal if queues persist.",
53     "justification": "loop_-1088700963_0 High congestion; however, main bottlenecks are downstream at  

54         25772784 and 25772779. Addressing those first is low-cost and consistent with Auckland ITS policy  

55         on signal coordination."
56   }
57 ],
58 "actions": [
59   {
60     "type": "update_attribute",
61     "target": "tlLogic",
62     "id": "GS_cluster_1345480308_25773538",
63     "attribute": "offset|phase.duration",
64     "new_value": "offset=10; phase[0]=28; phase[1]=3; phase[2]=22; phase[3]=3; phase[4]=31; phase[5]=3",
65     "xml_snippet": "<tlLogic id=\"GS_cluster_1345480308_25773538\" type=\"actuated\" programID=\"0\"  

66       offset=\"10\">\n        <phase duration=\"28\" state=\"GGGGGgggrr\" minDur=\"5\" maxDur=\"50\"/>\n        <phase duration=\"3\" state=\"yyYGgggrr\"/>\n        <phase duration=\"22\" state=\"rrrGGGGGr\" minDur=\"5\" maxDur=\"50\"/>\n        <phase duration=\"3\" state=\"rrryyyyGrr\"/>\n      <phase duration"

```

```

      ="\"31\" state=\"rrrrrrrGGG\" minDur=\"5\" maxDur=\"50\"/>\n    <phase duration=\"3\" state=\"rrrrrrgyy\"/>\n</tlLogic>"
```

42 },  
43 {  
44 "type": "create\_element",  
45 "target": "tlLogic",  
46 "id": "25772784",  
47 "xml\_snippet": "<tlLogic id=\"25772784\" type=\"actuated\" programID=\"0\" offset=\"0\">\n <!-- The  
 Strand both directions (intLanes 2,3,4,5) -->\n <phase duration=\"35\" state=\"rrGGGG\"/>\n <phase duration=\"4\" state=\"rryyyy\"/>\n <phase duration=\"1\" state=\"rrrrrr\"/>\n <!-- Gladstone movements (intLanes 0,1) -->\n <phase duration=\"20\" state=\"GGrrrr\"/>\n <phase duration=\"4\" state=\"yyrrrr\"/>\n <phase duration=\"1\" state=\"rrrrrr\"/>\n</tlLogic>"

48 },  
49 {  
50 "type": "create\_element",  
51 "target": "tlLogic",  
52 "id": "25772779",  
53 "xml\_snippet": "<tlLogic id=\"25772779\" type=\"actuated\" programID=\"0\" offset=\"0\">\n <!--  
 Favour 624523286 -> 606859931 straight-throughs (intLanes 2,3,4,5) -->\n <phase duration=\"40\" state=\"rrGGGGr\"/>\n <phase duration=\"4\" state=\"rryyyyr\"/>\n <phase duration=\"1\" state=\"rrrrrrr\"/>\n <!-- Minor phase: -7637457/-606859931 to -624523286 and right from 624523286 (intLanes 0,1,6) -->\n <phase duration=\"18\" state=\"GGrrrrG\"/>\n <phase duration=\"4\" state=\"yyrrrry\"/>\n <phase duration=\"1\" state=\"rrrrrrr\"/>\n</tlLogic>"

54 }  
55 ],  
56 "modified\_snippets": [  
57 "<tlLogic id=\"GS\_cluster\_1345480308\_25773538\" type=\"actuated\" programID=\"0\" offset=\"10\">\n <phase duration=\"28\" state=\"GGGGGgggrr\" minDur=\"5\" maxDur=\"50\"/>\n <phase duration=\"3\" state=\"yyyGGgggr\"/>\n <phase duration=\"22\" state=\"rrrGGGGGr\" minDur=\"5\" maxDur=\"50\"/>\n <phase duration=\"3\" state=\"rrrryyyyr\"/>\n <phase duration=\"31\" state=\"rrrrrrrGGG\" minDur=\"5\" maxDur=\"50\"/>\n <phase duration=\"3\" state=\"rrrrrrrgyy\"/>\n</tlLogic>",  
58 "<tlLogic id=\"25772784\" type=\"actuated\" programID=\"0\" offset=\"0\">\n <phase duration=\"35\" state=\"rrGGGG\"/>\n <phase duration=\"4\" state=\"rryyyy\"/>\n <phase duration=\"1\" state=\"rrrrrrr\"/>\n <phase duration=\"20\" state=\"GGrrrr\"/>\n <phase duration=\"4\" state=\"yyrrrr\"/>\n <phase duration=\"1\" state=\"rrrrrrr\"/>\n</tlLogic>",  
59 "<tlLogic id=\"25772779\" type=\"actuated\" programID=\"0\" offset=\"0\">\n <phase duration=\"40\" state=\"rrGGGGr\"/>\n <phase duration=\"4\" state=\"rryyyyr\"/>\n <phase duration=\"1\" state=\"rrrrrrr\"/>\n <phase duration=\"18\" state=\"GGrrrrG\"/>\n <phase duration=\"4\" state=\"yyrrrry\"/>\n <phase duration=\"1\" state=\"rrrrrrr\"/>\n</tlLogic>",  
60 "<junction id=\"25772784\" type=\"traffic\_light\" x=\"172.38\" y=\"355.22\" incLanes=\"7637453\_0  
-13718894\_0 -13718894\_1 606859931\_0 606859931\_1\" intLanes=\"25772784\_0\_0 :25772784\_1\_0 :25772784\_2\_0 :25772784\_2\_1 :25772784\_4\_0 :25772784\_4\_1\" shape=\"174.79,344.37 180.49,347.35 179.22,349.88  
178.86,351.18 178.70,352.49 178.72,353.82 178.93,355.17 166.44,357.96 165.89,355.65 165.39,353.96  
164.83,352.61 164.09,351.34 163.08,349.87 161.68,347.94 170.73,344.25 172.02,345.78 172.69,345.99  
173.37,345.82 174.07,345.28\">\n <request index=\"0\" response=\"111100\" foes=\"111100\" cont=\"0\"/>\n <request index=\"1\" response=\"000000\" foes=\"000000\" cont=\"0\"/>\n <request index=\"2\" response=\"001000\" foes=\"001001\" cont=\"0\"/>\n <request index=\"3\" response=\"000000\" foes=\"000001\" cont=\"0\"/>\n <request index=\"4\" response=\"000000\" foes=\"000001\" cont=\"0\"/>\n <request index=\"5\" response=\"000000\" foes=\"000001\" cont=\"0\"/>\n</junction>",  
61 "<junction id=\"25772779\" type=\"traffic\_light\" x=\"145.43\" y=\"316.15\" incLanes=\"-7637457\_0  
-606859931\_0 624523286\_0 624523286\_1\" intLanes=\"25772779\_0\_0 :25772779\_1\_0 :25772779\_2\_0 :25772779\_2\_1 :25772779\_2\_2 :25772779\_2\_3 :25772779\_7\_0\" shape=\"160.29,317.64 158.28,323.72  
156.45,323.43 156.06,323.63 156.00,324.07 156.29,324.75 156.93,325.66 149.37,331.58 137.31,316.00  
144.98,310.23 147.33,312.97 149.41,314.62 151.50,315.55 153.84,316.13 156.69,316.70\">\n <request index=\"0\" response=\"00000010\" foes=\"0000010\" cont=\"0\"/>\n <request index=\"1\" response=\"00000000\" foes=\"1000001\" cont=\"0\"/>\n <request index=\"2\" response=\"0010000\" foes=\"0000001\" cont=\"0\"/>\n <request index=\"3\" response=\"0110000\" foes=\"0110000\" cont=\"0\"/>\n <request index=\"4\" response=\"00000000\" foes=\"0001100\" cont=\"0\"/>\n <request index=\"5\" response=\"00000000\" foes=\"0001000\" cont=\"0\"/>\n <request index=\"6\" response=\"00000010\" foes=\"0000010\" cont=\"1\"/>\n</junction>"  
62 },  
63 ]

## B Improved LLM Output

The improved iteration refines the timing plan by adding coordinated offsets, shorter cycles, and explicit corridor progression along The Strand–Quay Street route.

```

1  {
2      "reasoning": [
3          {
4              "edge_or_junction_id": "TL_cluster_1345480308_25773538",
5                  "issue_detected": "Quay Street eastbound approach (edge 624515490#0) to the waterfront cluster is
6                      over-saturated: very low speeds (0.99-2.11 m/s) and extremely high occupancy (lane 2 =92.6%).
7                          Meanwhile, Tamaki Drive inbound (1246214209#0) shows low flows and lower occupancy.",
8                  "proposed_policy": "Shift green split toward Quay St EB by lengthening its through phase and trimming
9                      the opposite phase. Coordinate offsets to form a short green wave along The Strand/Quay corridor
10                     .",
11                 "justification": "Detectors loop_624515490#0_2 occupancy =92.6% with crawling speeds indicates strong
12                     unmet demand into this junction. By reallocating green from the underused Tamaki inbound to Quay
13                     EB, we relieve the bottleneck that propagates upstream. A small positive offset aligns platoons
14                     emerging from upstream signals."
15             },
16             {
17                 "edge_or_junction_id": "25772784",
18                     "issue_detected": "The Strand northbound approach (-13718894) into this signal exhibits severe
19                         congestion (avg speeds =3.6-4.2 m/s and occupancy up to =84%).",
20                     "proposed_policy": "Increase the green time for the phase serving The Strand NB through movement.
21                         Keep clearances unchanged. Add a small positive offset for progression from 25772779.",
22                     "justification": "High occupancy and low speeds on -13718894 indicate the approach is under-served.
23                         Rebalancing the split (+15 s to this movement group) reduces queue spillback that otherwise
24                         blocks upstream links and degrades corridor performance."
25             },
26             {
27                 "edge_or_junction_id": "25772779",
28                     "issue_detected": "Upstream of 25772784, the approach from The Strand (-624523286) is heavily loaded
29                         (loop_624523286_1 occupancy =83.8%).",
30                     "proposed_policy": "Increase the mainline green (The Strand) while keeping side-street service
31                         sufficient. Maintain yellows and intergreens. Use this junction as corridor time base (offset 0)
32                         and coordinate downstream.",
33                     "justification": "Elevated occupancy on -624523286_1 shows sustained demand. Extending the primary
34                         phase reduces recurrent queueing and better feeds downstream signals that are now retimed to
35                         accept the platoon."
36             }
37         ],
38         "actions": [
39             {
40                 "type": "update_attribute",
41                     "target": "tlLogic",
42                     "id": "TL_cluster_1345480308_25773538",
43                     "attribute": "offset|phase.duration",
44                     "new_value": "offset=9; phase[0]=27; phase[1]=3; phase[2]=36; phase[3]=3",
45                     "corridor": [
46                         {
47                             "id": "TheStrand_Quay_EB_Progression",
48                             "order": [
49                                 "25772779",
50                                 "25772784",
51                                 "TL_cluster_1345480308_25773538"
52                             ],
53                             "progression_speed_kmh": 40,
54                             "segment_lengths_m": [
55                                 21,
56                                 74
57                             ],
58                             "reference_junction": "25772779"
59                         },
60                         {
61                             "xml_snippet": "<tlLogic id=\"TL_cluster_1345480308_25773538\" type=\"actuated\" programID=\"0\""
62                                 "offset=\"9\">\n                                <phase duration=\"27\" state=\"GGGGGGGrrrr\" minDur=\"10\" maxDur=\"70\"/>\n                                <phase duration=\"3\" state=\"yyyyyyyyrrrr\" minDur=\"3\" maxDur=\"3\"/>\n                                <phase duration=\"36\" state=\"rrrrrrrrGGGG\" minDur=\"7\" maxDur=\"40\"/>\n                                <phase duration=\"3\" state=\"rrrrrrrryyyy\" minDur=\"3\" maxDur=\"3\"/>\n                            </tlLogic>"
63                         }
64                     ],
65                     "type": "update_attribute",
66                     "target": "tlLogic",
67                     "id": "25772784",
68                     "attribute": "offset|phase.duration",
69                 }
70             ]
71         ]
72     }
73 }
```

```

50
51     "new_value": "offset=2; phase[0]=30; phase[1]=4; phase[2]=1; phase[3]=35; phase[4]=4; phase[5]=1",
52     "corridor": {
53         "id": "TheStrand_Quay_EB_Progression",
54         "order": [
55             "25772779",
56             "25772784",
57             "TL_cluster_1345480308_25773538"
58         ],
59         "progression_speed_kmh": 40,
60         "segment_lengths_m": [
61             21,
62             74
63         ],
64         "reference_junction": "25772779"
65     },
66     "xml_snippet": "<tlLogic id=\"25772784\" type=\"actuated\" programID=\"0\" offset=\"2\">\n    <phase\n        duration=\"30\" state=\"rrGGGG\"/>\n        <phase duration=\"4\" state=\"rryyyy\"/>\n        <phase\n            duration=\"1\" state=\"rrrrrr\"/>\n            <phase duration=\"35\" state=\"GGrrrr\"/>\n            <phase\n                duration=\"4\" state=\"yyrrrr\"/>\n    </phase>\n</tlLogic>\n",
67 },
68     {
69         "type": "update_attribute",
70         "target": "tlLogic",
71         "id": "25772779",
72         "attribute": "offset|phase.duration",
73         "new_value": "offset=0; phase[0]=40; phase[1]=3; phase[2]=30; phase[3]=3",
74         "corridor": {
75             "id": "TheStrand_Quay_EB_Progression",
76             "order": [
77                 "25772779",
78                 "25772784",
79                 "TL_cluster_1345480308_25773538"
80             ],
81             "progression_speed_kmh": 40,
82             "segment_lengths_m": [
83                 21,
84                 74
85             ],
86             "reference_junction": "25772779"
87         },
88         "xml_snippet": "<tlLogic id=\"25772779\" type=\"actuated\" programID=\"0\" offset=\"0\">\n    <phase\n        duration=\"40\" state=\"GGGGGrrr\" minDur=\"10\" maxDur=\"70\"/>\n        <phase duration=\"3\"\n            state=\"yyyyyrrr\" minDur=\"3\" maxDur=\"3\"/>\n            <phase duration=\"30\" state=\"rrrrrGGG\" minDur=\"7\" maxDur=\"40\"/>\n            <phase duration=\"3\" state=\"rrrrryyy\" minDur=\"3\" maxDur=\"3\"/>\n    </phase>\n</tlLogic>\n",
89     },
90     "modified_snippets": [
91         "<tlLogic id=\"TL_cluster_1345480308_25773538\" type=\"actuated\" programID=\"0\" offset=\"9\">\n    <phase\n        duration=\"27\" state=\"GGGGGGGrrrr\" minDur=\"10\" maxDur=\"70\"/>\n        <phase duration=\"3\" state=\"yyyyyyrrrr\" minDur=\"3\" maxDur=\"3\"/>\n        <phase duration=\"36\" state=\"rrrrrrrGGGG\" minDur=\"7\" maxDur=\"40\"/>\n        <phase duration=\"3\" state=\"rrrrrrryyyy\" minDur=\"3\" maxDur=\"3\"/>\n    </phase>\n</tlLogic>\n",
92         "<tlLogic id=\"25772784\" type=\"actuated\" programID=\"0\" offset=\"2\">\n    <phase duration=\"30\"\n        state=\"rrGGGG\"/>\n        <phase duration=\"4\" state=\"rryyyy\"/>\n        <phase duration=\"1\" state=\"rrrrrr\"/>\n        <phase duration=\"35\" state=\"GGrrrr\"/>\n        <phase duration=\"4\" state=\"yyrrrr\"/>\n    </phase>\n</tlLogic>\n",
93         "<tlLogic id=\"25772779\" type=\"actuated\" programID=\"0\" offset=\"0\">\n    <phase duration=\"40\"\n        state=\"GGGGGrrr\" minDur=\"10\" maxDur=\"70\"/>\n        <phase duration=\"3\" state=\"yyyyyrrr\" minDur=\"3\" maxDur=\"3\"/>\n        <phase duration=\"30\" state=\"rrrrrGGG\" minDur=\"7\" maxDur=\"40\"/>\n        <phase duration=\"3\" state=\"rrrrryyy\" minDur=\"3\" maxDur=\"3\"/>\n    </phase>\n</tlLogic>\n"
94     ]
95 }

```