TRACE: Traffic Response Anomaly Capture Engine for Localization of Traffic Incidents

Ammar Bin Zulqarnain[1], Jacob Buckelew[1], Jose Paolo Talusan[1], Ayan Mukhopadhyay[1], Abhishek Dubey[1]

[1]Vanderbilt University

***Abstract*—Effective traffic incident management is critical for road safety and operational efficiency. Yet, many transportation agencies rely on reactionary methods, where incidents are reported by human agents and managed through rule-based frameworks like traditional Traffic Incident Management (TIM) systems. However, these are vulnerable to human error, oversight, and delays during high-stress conditions. Although recent initiatives incorporating real-time sensor data for corridor monitoring and enhanced roadway information systems represent strides toward modernization, these systems often still require substantial human intervention. Recent advancements in graph-based deep learning models offer promising potential for addressing the limitations of traditional methods. While state-of-the-art models exist, the complexities of incident localization within dynamic and interconnected road networks, along with limited availability of high-quality labeled data and variability in real-time traffic measurements, are still open challenges. To address these, we propose the Traffic Response Anomaly Capture Engine (TRACE), a novel approach that combines graph neural networks, transformers, and probabilistic normalizing flows to accurately detect and localize traffic anomalies in real time. TRACE captures spatial-temporal dependencies, manages data uncertainty, and enhances automation, supporting more precise and timely incident localization. Our approach is validated on real-world traffic data and improved incident localization by 0.6 miles (17%) than SOTA methods while maintaining similar incident detection accuracy and mean detection delay.**

I. Introduction

Traffic anomaly detection is essential for ensuring road safety and maintaining operational efficiency. Rapid identification and response to roadway incidents help mitigate congestion, reduce the risk of secondary accidents, and facilitate timely emergency interventions [1]. Research has shown that delayed response times significantly impact roadway safety, prolonging congestion and increasing the likelihood of secondary crashes which account for nearly 20% of all freeway accidents [2, 3].

Traffic incidents—ranging from minor accidents to major collisions—can cause widespread disruptions across road networks, leading to economic losses and compromised safety. Despite technological advancements, many transportation agencies continue to struggle with accurate real-time anomaly detection and localization due to the complexity and scale of modern road systems. Traditional manual monitoring remains labor-intensive, prone to human error, and often results in delayed responses, particularly under high-stress conditions [4]. Transportation agencies and private companies have added sensors and utilized crowd-sourced information to get ahead of potential incidents.

Although real-time sensor-based monitoring systems have emerged, they often demand significant human intervention and struggle with dynamic, large-scale road networks [5]. Research indicates that such inefficiencies can delay emergency responses by 20 to 30% [6], while precise real-time localization strategies have been shown to reduce secondary crashes by up to 50% [7]. Consequently, robust methods capable of accurately localizing incidents and managing uncertainties from missing or delayed incident reports are urgently needed to ensure efficient and scalable traffic management. This underscores the urgent need for automated, accurate, and scalable incident detection solutions.

Recent advances in traffic incident detection have predominantly employed data-driven approaches[8], especially leveraging graph-based autoencoder models such as Graph Convolutional Networks (GCNs) [9], Graph Attention Networks (GATs) [10], and Transformer-based frameworks. For instance, Relational Graph Convolutional Network Autoencoders [11] explicitly encode spatial-temporal dependencies, while hybrid models like GCN-LSTM[12] combine graph convolutions with RNN-based temporal modeling to enhance predictive accuracy. These architectures have significantly improved anomaly detection performance by capturing complex traffic dynamics and patterns. By focusing primarily on global traffic patterns, these methods often under perform in accurately localizing incidents. This limitation arises because the aggregation of global features can mask localized disruptions, making it difficult for models to isolate and identify the precise regions where anomalies occur. Furthermore, these approaches typically assume that the data is complete and pristine, thereby neglecting the inherent uncertainty in real-world traffic data. In practice, sensor measurements are often noisy, incomplete, or subject to environmental disturbances, leading to significant data uncertainty. The absence of robust mechanisms to account for and mitigate these uncertainties not only hampers the detection of subtle local anomalies but also impairs the precise localization of incidents. Consequently, while global traffic patterns provide a useful overview, they fail to capture the nuanced variations necessary for pinpointing localized disruptions, leaving a critical gap in current methodologies.

In this paper, we present the Traffic Response Anomaly Capture Engine (TRACE), a novel framework designed for robust traffic anomaly localization. TRACE's primary innovation lies in its integration of Graph Neural Networks (GNNs), Transformers, and Normalizing Flows into a unified probabilistic modeling approach. Unlike existing methods, our framework explicitly captures complex spatial dependencies through GNNs, models long-range temporal interactions via Transformers, and quantifies predictive uncertainty using Normalizing Flows. This innovative combination enables TRACE to adapt dynamically to varying traffic conditions, significantly improving the reliability, precision, and efficiency of anomaly detection, ultimately enhancing overall road safety.

We demonstrate accurate, real-time anomaly detection across large-scale real-world road networks. We show that TRACE significantly improves incident localization precision, reducing response times and enhancing traffic management compared to existing state-of-the-art models. To evaluate the proposed approach, we consider six baselines ranging from complex relational graph convolutional network encoders to simple multi-layer perceptron autoencoders. We validated our approach on real-world traffic data for a mid-sized metropolitan city in the USA. Our experimental evaluations show that TRACE outperforms the baselines in terms of incident localization and detection.

II. Problem Statement

We formulate the problem of traffic incident localization and detection as a general anomaly detection problem with the goal of finding diversions from the expected normal traffic operations.

Spatio-Temporal Network Representation

Consider a spatial area of interest where V denotes the set of all road sensors under consideration with $M = |V|$. Consider an arbitrary sensor $v \in V$ on which (near) real-time speed is monitored continuously. We assume that the estimated harmonic mean speed on sensor v is computed and stored at discrete times $t \in \{1, 2, \ldots, T\}$. We denote this observation at time t by s_t.

The problem can be converted into a graph problem. Consider now a graph $G = (V, E)$, where V represents the set of nodes corresponding to road sensors, and $E \subseteq V \times V$ represents the undirected edges indicating connectivity between road sensors. Each $v \in V$ corresponds to a sensor in the area of interest, with features x_t representing the speed, volume and occupancy at time t. The adjacency matrix $A \in \mathbb{R}^{|V|\times|V|}$ of the graph G is defined to encode spatial connectivity. Thus, a graph $G(t)$ is a snapshot of existing congestion across an entire road network at an arbitrary time t. For each t, we define $X_t = \big[x_t(v_1), x_t(v_2), \ldots, x_t(v_m)\big]$, where X_t is the concatenation of the feature vectors for all sensors in V.

Objective: Anomaly Detection and Localization

We aim to detect and localize anomalies by comparing each sensor's observed measurements to those expected under normal traffic conditions. In our formulation, the probability $p(x_t(v))$ quantifies the likelihood of observing the feature vector $x_t(v)$ (which includes speed, volume, and occupancy) at sensor v at time t under normal conditions. To enhance numerical stability and ease the detection of significant deviations, we employ the logarithm of this probability, i.e., the log density $\log p(x_t(v))$.

Formally, we define an anomaly indicator $A(v, t)$ for sensor v at time t as:

$$A(v,t) = \begin{cases} 1, & \text{if } \log p\big(x_t(v)\big) \text{ deviates significantly from} \\ & \quad \text{normal conditions,} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Thus, anomalies capture points in space and time where the log density of the measurements deviates markedly from typical patterns observed across the network.

Optimization Objective

The objective of the problem is to find the weights of the model that maximize the likelihood of the observed values across all edges and time steps:

$$\max \sum_{t=1}^{T} \sum_{v \in V} \log p(\mathbf{Z}(t)_v) \tag{2}$$

In this equation, $\mathbf{Z}(t)_{v,\cdot}$ denotes the full latent representation (i.e., all latent dimensions) of sensor v at time t. Thus, the optimization objective maximizes the log-probability of observing these latent representations across all sensors and time steps. This objective ensures that the model accurately captures normal relationships within the network, enabling effective anomaly detection and localization.

III. Related Work

This section reviews the progression of traffic anomaly detection methods, highlighting the strengths and limitations of each approach and identifying gaps that inform the need for more robust solutions.

Data Sources and Sensor Technologies: Traffic anomaly detection systems leverage both intrusive and non-intrusive sensor technologies [13, 14]. They offer high accuracy for vehicle detection and classification but often require traffic disruptions, maintenance challenges, and can be affected by environmental factors [13, 15, 16]. Non-intrusive sensors, installed overhead or roadside, provide real-time monitoring with minimal impact on traffic flow. While effective in diverse environments, these sensors also face issues such as reduced performance in adverse weather, occlusion, and high installation costs [13, 14, 16].

Rule-based and Statistical Approaches: Transportation agencies like the Tennessee Department of Transportation (TDOT) initially relied on manual monitoring and basic Intelligent Transportation Systems (ITS) [17]. Operators watched camera feeds, analyzed sensor data, and coordinated with emergency responders. While these methods provided foundational monitoring capabilities, they were prone to human error, fatigue, and delays in incident detection, especially during peak traffic hours. Fixed-threshold methods improved this by flagging anomalies based on predefined limits, such as sudden drops in speed or traffic flow. These struggled with the complexity of real-world traffic patterns, often failing to distinguish between anomalies and routine congestion.

Machine Learning and Graph-Based Methods: The availability of large-scale traffic data and increased computational power enabled advanced machine learning (ML) and deep learning (DL) techniques in traffic anomaly detection [18]. Long Short-Term Memory (LSTM) models, improved the modeling of temporal dependencies in traffic data [19]. By representing intersections as nodes and road segments as edges, GNNs capture how traffic conditions at one location influence neighboring areas [18, 20]. These models significantly improved the ability to detect network-wide anomalies and understand complex spatial-temporal patterns. However, these approaches encounter uncertainty [21], real-time adaptation, interpretability, and scaling problems.

Transformer Architectures and Attention Mechanisms: Transformer architectures introduced attention mechanisms that enhanced the modeling of long-range temporal dependencies in traffic data [22]. By selectively focusing on critical segments of input sequences, Transformers effectively detected both gradual and abrupt anomalies, outperforming traditional RNN-based models in capturing complex temporal patterns. Integrating Transformers with GNNs allowed researchers to combine attention-based temporal modeling with graph-based spatial learning. This hybrid approach showed promise in capturing intricate spatiotemporal dynamics, though it introduced higher computational overhead and did not fully resolve challenges related to uncertainty quantification.

Probabilistic Modeling and Normalizing Flows: Normalizing Flows (NFs) are used to model the flexible data distributions and providing likelihood estimates for traffic patterns [23]. Early density-based approaches, such as kernel density estimation for anomaly scoring [24] and RealNVP for trajectory anomalies [25], demonstrated the potential of flow-based models. However, these methods often overlooked sequential dependencies critical in traffic data. The integration of NFs with GNNs and Transformers offers a promising avenue for creating models that can capture complex spatiotemporal relationships while explicitly modeling uncertainty.

IV. Proposed Method

We propose a probabilistic approach that learns the *distribution of normal traffic patterns*, enabling anomalies to be identified as deviations from this distribution. Our method integrates *Graph Convolutional Networks (GCNs)* to capture spatial dependencies between road segments. *Transformers* to model temporal dependencies, capturing long-range interactions across time, and *Normalizing Flows* to estimate the probability distribution of traffic conditions, enabling unsupervised anomaly detection without requiring labeled anomalies. This architecture, shown in Figure 1, ensures a fully data-driven approach, learning expressive spatio-temporal representations while providing probabilistic anomaly scores that highlight deviations from learned traffic patterns.

A. Graph Convolutional Networks for Spatial Modeling

To capture spatial dependencies among road segments, we employ Graph Convolutional Networks (GCNs), which aggregate information from neighboring nodes based on graph adjacency relationships. Each layer l in the GCN updates node features using the propagation rule:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$$

where $\tilde{A} = A + I$ is the adjacency matrix with self-loops, and $\tilde{D}$ is the diagonal degree matrix, where $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$. $H^{(l)}$ denotes the node feature matrix at layer l, with $H^{(0)} = X_t$, while $W^{(l)}$ is a learnable weight matrix. Finally, $\sigma(\cdot)$ represents a non-linear activation function, such as ReLU.

The final spatial representation is given by:

$$Z_{\text{spatial}} = \text{GCN}(X_t, A) \in \mathbb{R}^{M\times H}$$

where H is the embedding dimension encoding spatial relationships.

B. Transformer Networks for Temporal Modeling

To effectively capture temporal dependencies, we leverage Transformer Networks, which employ self-attention mechanisms to model interactions across different time steps. Given a sequence of feature matrices over a temporal window T:

$$\mathbf{X} = \{X_t\}_{t=1}^{T} \in \mathbb{R}^{M\times T\times d}$$

The Transformer applies self-attention to compute contextualized embeddings:

$$\mathbf{Z}_{\text{temporal}}(t) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V$$

where Q, K, V are the query, key, and value matrices derived from $\mathbf{X}$, and d represents the feature dimension. The self-attention mechanism assigns weights to different time steps, prioritizing relevant past observations.The final temporal representation is computed as:

$$Z_{\text{temporal}} = \text{Transformer}(\mathbf{X}) \in \mathbb{R}^{M\times d}$$

C. Unified Latent Spatio-Temporal Latent Representation of Road Networks

Following the construction of $\mathbf{Z}_{\text{spatial}}$ and $\mathbf{Z}_{\text{temporal}}$ (detailed in the previous subsections), we obtain a comprehensive spatio-temporal embedding at each timestep t by concatenating and reducing these components:

$$\mathbf{Z}(t) \;=\; \text{ConcatAndReduce}\big(\mathbf{Z}_{\text{spatial}}(t),\, \mathbf{Z}_{\text{temporal}}(t)\big) \;\in\; \mathbb{R}^{M\times H},$$

where H denotes the dimensionality of the latent space. Each row $\mathbf{Z}(t)_v$ encodes the spatio-temporal characteristics of node v at time t. Over a time window of length T, these time-specific latent matrices form the sequence

$$\mathbf{Z} \;=\; \{\mathbf{Z}(t)\}_{t=1}^{T} \;\in\; \mathbb{R}^{M\times T\times H},$$

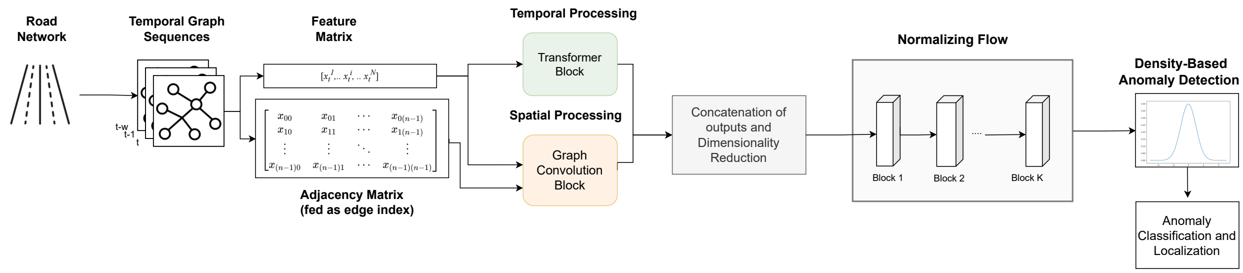

Fig. 1. Overview of the proposed model framework. The diagram illustrates the integration of GCNs for spatial representation, Transformers for temporal modeling, and normalizing flows for anomaly detection.

which serves as a compact representation of spatial and temporal relationships across all timesteps. This design leverages Graph Neural Networks (GNNs) to capture spatial connectivity, while Transformers handle the temporal dynamics, ensuring comprehensive coverage of traffic behavior in both space and time.

D. Normalizing Flow for Anomaly Detection

Most existing deep learning models for traffic anomaly detection rely on deterministic representations or heuristic-based anomaly scores. In contrast, our approach explicitly characterizes the probability distribution of traffic representations via normalizing flows, enabling a principled, likelihood-based anomaly detection strategy.

Rather than directly classifying anomalies, we model the distribution of normal traffic behaviors through the latent representation of the road network at time t, denoted as $\mathbf{Z}(t)$. We adopt the RealNVP architecture [26] as the foundation for our normalizing flow model. To better capture the nuances of traffic data, we introduce slight modifications to the standard coupling layers by conditioning them on the spatial-temporal features extracted from our GCN and Transformer layers, thereby enhancing the flow's expressive power for anomaly detection. These flows transform a simple base distribution (e.g., a standard Gaussian) into a complex target distribution via a sequence of invertible and differentiable mappings. By applying the inverse of these mappings, the density of any sample can be computed explicitly with respect to the base distribution. This bijective property allows us to obtain precise likelihood scores for each sensor's observation over time, so that observations with low likelihoods—indicating significant deviations from normal patterns—are flagged as potential anomalies.

To estimate likelihoods, we introduce a normalizing flow model $f : \mathbb{R}^{M\times T\times H} \to \mathbb{R}^{M\times T}$, which transforms the learned latent representations into a structured probability space. The log-likelihood of the traffic representations at time t is computed as:

$$\log p(\mathbf{Z}(t)) = [f(\mathbf{Z}(t)_1), f(\mathbf{Z}(t)_2), \ldots, f(\mathbf{Z}(t)_M)] \tag{3}$$

where $\log p(\mathbf{Z}(t))$ represents the estimated log-likelihoods of all nodes in the traffic network at time t.

TABLE I
SUMMARY OF FEATURES USED IN THE STUDY.

Attribute	Source	Description
Traffic Speed	Radar Detection Sensors(RDS)	Measures the average speed of vehicles within a specific road segment.
Vehicle Volume	Radar Detection Sensors(RDS)	Counts the number of vehicles passing a sensor per time segment.
Occupancy	Radar Detection Sensors(RDS)	Represents the percentage of time a sensor detects a vehicle over a given period.
Sensor Location	Radar Detection Sensors(RDS)	Geographic coordinates of traffic sensors placed along the roadway.
Network Structure	Derived	Defines the connectivity between different sensors in the road network.
Incident Labels	TDOT Reports	Ground truth labels indicating reported crashes or anomalies in the traffic system.

For anomaly detection, we compute an anomaly score for each node at each timestep as the negative log-likelihood:

$$\mathcal{L}(v,t) = -\log p(\mathbf{Z}(t)_v) \tag{4}$$

Nodes with high anomaly scores are flagged as potential outliers, indicating unusual traffic patterns. This likelihood-based formulation enables an unsupervised, robust, and interpretable anomaly detection framework, effectively capturing deviations from expected traffic behavior.

We extend this formulation to structured traffic networks, optimizing the joint likelihood of all nodes and time steps:

$$\max \sum_{t=1}^{T} \sum_{v\in V} \log p(\mathbf{Z}(t)_v) \tag{5}$$

This objective enables the model to capture both local road behaviors and global network-wide dependencies, enhancing its ability to detect deviations from normal traffic patterns.

V. EXPERIMENTS AND EVALUATION

In this section, we discuss the datasets used in the evaluation, introduce the baseline models for comparison, review the experimental setup, and finally discuss the results. All of our code, models and dataset is available here: https://github.com/ammarzulu/TRACE

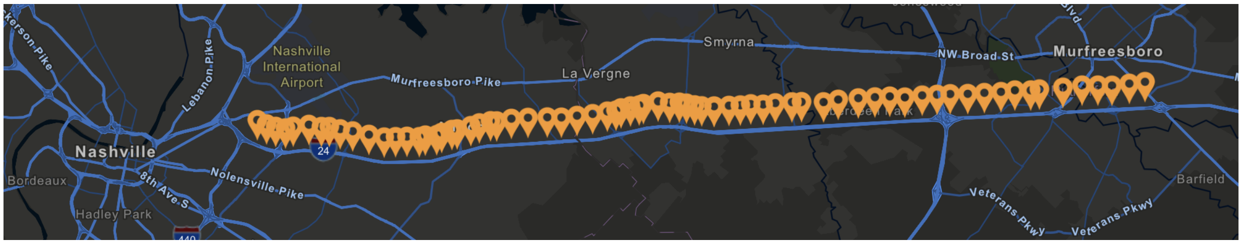

Fig. 2. Layout of radar detection sensors along Interstate 24 West near Nashville, TN.

A. Dataset Description

We used the FT-AED dataset that has been curated to provide high-resolution, lane-level data collected via 49 Radar Detection Sensors (RDS) installed along Interstate 24 (I-24) [8]. This stretch, spanning 18 miles near Nashville, TN, shown in fig. 2, captures critical freeway dynamics during peak traffic hours. The data set is limited to a temporal scope of morning peak hours (4:00 AM to 12:00 PM) over October 2023. The data set includes the speed, volume and occupancy of traffic recorded every 30 seconds and includes 12 confirmed TDOT crash reports and 8 expert-labeled anomalies. The granularity of the data set facilitates the detection of subtle anomalies that traditional aggregated data sources often miss. A complete list of features is included in Table I.

B. Baseline Models

Several baseline models, from state-of-the-art to simple perceptrons, were used for comparison against TRACE to evaluate anomaly detection and localization performance:

- **STG-RGCN AE**: A Relational Graph Convolutional Network Autoencoder designed to capture relational dependencies in traffic networks using Relational Graph Convolution blocks [8, 11]. It is important to note that this model is specifically tailored to this dataset, as it relies on predefined relational graphs constructed from the spatial and temporal relationships between lanes and adjacent sensors across timesteps. This dependence on structured relationships limits its generalizability to other datasets with different network configurations.
- **STG-GAT AE**: A Spatiotemporal Graph Attention Network Autoencoder that leverages Graph Attention blocks to learn spatial dynamics and improve anomaly detection [10].
- **GCN-LSTM AE**: A hybrid model combining Graph Convolutional Networks (GCNs) for spatial learning with Long Short-Term Memory (LSTM) networks for temporal learning [12].
- **GCN AE**: A standard Graph Convolutional Network Autoencoder that models spatial dependencies without additional temporal features.
- **Transformer AE**: A Transformer-based autoencoder that captures temporal dependencies but does not explicitly model spatial relationships.
- **MLP AE**: A simple Multi-Layer Perceptron Autoencoder, treating each node independently without leveraging spatial-temporal correlations.

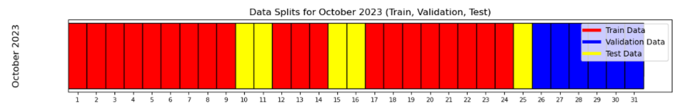

Fig. 3. Dataset Setup: The days in yellow are the detection target as it contains crashes of interest. The days in red are used to train the models for detection and days in blue are used as validation dataset for hyperparameter tuning

C. Experimental Setup

The goal of this experiment is to evaluate the effectiveness of machine learning methods in detecting anomalous events on freeways using the FT-AED dataset. Specifically, the study aims to benchmark models for their ability to detect anomalies in traffic data with high accuracy, reduce reporting delays for crashes and other incidents, and accurately localize anomalies at the lane and mile-marker level.

The experiment involves three distinct datasets—training, validation, and test sets—designed as seen in Figure 3 to capture various traffic conditions and enable comprehensive evaluation. The training dataset comprises 80% of the month's morning traffic data , including anomalies, to train models for generating negative log-likelihoods (used by TRACE) or reconstructing traffic conditions (used by baselines) and identifying deviations. The validation dataset includes 10% of the morning data, covering diverse scenarios such as multi-lane congestion and isolated incidents, with the objective of fine-tuning model hyperparameters and dynamically adjusting anomaly detection thresholds. Finally, the test dataset consists of five specific days containing incidents of interests in October 2023 (10, 11, 15, 16, and 25) and is used to evaluate model generalization on unseen, challenging traffic scenarios.

D. Data Preprocessing and Hyperparameter Tuning

The preprocessing and hyperparameter pipeline follows multiple steps to ensure data quality and compatibility with machine learning algorithms.

For preprocessing the data, duplicate record entries need to be pruned and missing localization information was imputed using merging different datasets for localization of information in FT-AED Dataset. Then, incident records are verified using Enhanced Tennessee Roadway Information Management System. Normalization is done by applying Min-Max scaling to numerical attributes to ensure uniform input distribution.

TABLE II
PERFORMANCE COMPARISON OF TRACE AND BASELINE MODELS (BEST METRICS IN BOLD, SECOND BEST UNDERLINED).

Methods	AUC	Mean Detection Delay (minutes)	Mean Localization Error (miles)	Incidents Detected (out of 10)
TRACE (Our Approach)	**0.7124**	-9.140 ± 8.955	**2.5549 ± 2.100**	7
STG-RGCN	0.6843	-9.350 ± 8.600	<u>3.0800 ± 3.210</u>	**10**
STG-GAT	0.6600	**-10.500 ± 6.819**	3.3600 ± 3.064	7
GCN	<u>0.7040</u>	<u>-9.750 ± 5.680</u>	3.4460 ± 3.475	8
GCN-LSTM	0.6900	-8.000 ± 6.142	3.4461 ± 3.343	<u>9</u>
MLP	0.6204	-3.300 ± 8.7950	3.9670 ± 2.876	5
Transformer	0.4987	-4.375 ± 7.2402	3.3249 ± 2.100	4

Hyperparameter tuning was systematically conducted using Optuna [27], an efficient hyperparameter optimization framework, to identify the configuration yielding the lowest validation loss. We systematically optimized both data and model parameters for TRACE and baselines. For data parameters, we varied the length of the input time windows. For model parameters, we tuned the hidden and latent dimensions, the number of GNN layers, temporal layers (e.g., Transformer and LSTM layers), attention heads (for GAT and Transformer), dropout rates. We also adjusted training parameters such as the batch size, learning rate to further enhance performance. The summary of the steps taken to tune models are as follows:

- **Search Framework**: Hyperparameters were explored through Optuna's Bayesian optimization method. This approach adaptively selects promising hyperparameter sets based on previous trials, significantly reducing computational overhead.
- **Holdout Validation Set**: Hyperparameters were evaluated using a dedicated holdout validation set, providing an unbiased estimate of model performance and minimizing the risk of overfitting.
- **Final Selection**: The optimal hyperparameter set was determined by the configuration with the lowest validation loss, promoting model reliability and performance consistency.

This targeted hyperparameter tuning approach ensured that the final model achieved optimal performance suitable for practical anomaly detection and localization applications.

E. Evaluation Metrics

The models were assessed using multiple performance metrics to ensure a thorough evaluation. All of these models were evaluated on the fixed 5% FPR as done in Coursey et al.:

- **AUC (Area Under the Curve)**: Measures the model's effectiveness in distinguishing between normal and anomalous traffic conditions by analyzing the trade-off between true positive and false positive rates across varying detection thresholds, serving as a key performance indicator.
- **Mean Localization Error:** We introduce this metric to quantify the average deviation between predicted anomaly locations and actual incident locations, measured in miles. This metric evaluates the spatial precision of the model by computing the mean absolute distance between detected anomalies and ground truth locations. Lower values indicate better localization accuracy, ensuring that detected incidents align closely with real-world traffic disruptions.
- **Mean Detection Delay**: This metric quantifies the time lag between the model's detection of an anomaly and its official reporting. A lower detection delay indicates the model's effectiveness in providing early warnings, which is critical for proactive traffic incident management.
- **Incidents Detected**: This measures the number of incidents successfully identified by the model out of 10 known incidents that include localization information. A higher detection count reflects the model's capability to capture significant traffic disruptions effectively

F. Results and Discussion

TRACE outperformed multiple baselines in anomaly detection, as evaluated using AUC, detection delay, localization accuracy, and anomaly detection rate. The results, summarized in Table II, highlight TRACE's ability to effectively distinguish anomalies from normal traffic patterns.

TRACE achieved the highest AUC score of 0.7124, outperforming all other models. This highlights its strong capability in distinguishing anomalies from normal traffic patterns. For comparison, STG-RGCN obtained an AUC of 0.6843, while GCN and GCN-LSTM followed with scores of 0.704 and 0.69, respectively. TRACE's integration of Normalizing Flows, which estimate the likelihood of traffic states, significantly enhanced its anomaly detection performance.

In terms of mean localization error, TRACE exhibited the lowest mean localization error of 2.55 miles, outperforming the other models by at least 17%. STG-RGCN had a slightly higher localization error of 3.08 miles, while STG-GAT and GCN-LSTM recorded errors of 3.36 miles and 3.44 miles, respectively. TRACE's ability to leverage Graph Neural Networks (GNNs) for spatial modeling played a crucial role in achieving high precision, making it particularly effective for real-world applications where precise incident localization is essential.

TRACE also demonstrated competitive detection speed, with a mean delay of -9.14 minutes. Meanwhile STG-GAT achieved the shortest detection delay of -10.5 minutes, followed by GCN (-9.75 minutes) and STG-RGCN (-9.35 minutes). Although its detection speed is marginally slower, this trade-off is generally acceptable because TRACE maintains a stronger overall detection capability and high localization precision. In many real-world scenarios, a difference of less than two minutes is offset by the benefits of consistently accurate anomaly identification and precise location estimates—key factors in ensuring effective incident response. By contrast, Transformer-based methods exhibited notably lower detection speed (-4.375 minutes), and GCN-LSTM recorded -8.0 minutes, indicating that TRACE still achieves a favorable balance between speed and computational efficiency.

In terms of overall anomaly detection, TRACE identified 7 out of 10 incidents, demonstrating reliable performance. STG-RGCN detected all 10 anomalies, while GCN-LSTM identified 9 cases. Although TRACE performed slightly below these models in terms of detection rate, it remained a robust approach for identifying traffic anomalies.

It is important to note that while STG-RGCN achieved the highest detection rate, it operates on a relational graph structure, which is tailored to this specific traffic dataset as mentioned in Coursey et al. This makes STG-RGCN domain-dependent, requiring predefined relationships between nodes, limiting its adaptability to different road networks or sensor configurations. In contrast, TRACE and GCN-LSTM are more domain-agnostic, as they do not rely on a fixed relational graph and can adapt better across varying traffic environments.

These results indicate room for improvement in handling complex or edge-case scenarios where TRACE slightly underperformed compared to some baselines. Nevertheless, TRACE's superior anomaly detection accuracy, competitive response time, and precise localization confirm its effectiveness as a robust and adaptive solution for real-time traffic anomaly detection.

G. Ablation Studies

To evaluate the contributions of key components in the TRACE model, we conducted ablation studies by systematically removing Graph Neural Networks (GNNs) and the temporal modeling module. We then analyze the effects of different temporal architectures on the performance of TRACE. The results,shown in Table III, demonstrate that both components are essential for achieving high anomaly detection accuracy and precise incident localization.

Impact of Removing Key Components: Removing GNNs significantly reduced anomaly detection and localization, as the model struggled to capture spatial dependencies in traffic data, with the AUC decreasing from 0.7124 to 0.3628. This suggests that spatial relationships among road segments are critical for effectively identifying and localizing anomalies. Similarly, eliminating the temporal modeling module resulted in a decline in anomaly detection , as the model failed to capture temporal dependencies in traffic flow. The AUC decreased from 0.7124 to 0.5000, indicating that sequential traffic patterns play a crucial role in distinguishing anomalous events from normal fluctuations.

Comparison of Different Temporal Architectures: We evaluated the model's performance using GRU and LSTM instead of the Transformer-based temporal module. Both alternatives showed a decline in AUC, indicating that the Transformer better captures long-range dependencies in traffic data.

TABLE III
ABLATION OF MODEL COMPONENTS AND TEMPORAL ARCHITECTURES.

Model	Ablation Type	AUC
TRACE/Transformer (Full)	-	**0.7124**
TRACE (No GNN)	Model	0.3628
TRACE (No Temporal)	Model	0.5000
TRACE/LSTM	Temporal Architecture	0.624
TRACE/GRU	Temporal Architecture	0.603

These results emphasize the importance of TRACE's spatial-temporal architecture, where both GNNs and Transformer-based temporal modeling significantly contribute to accurate anomaly detection and localization.

VI. Conclusion and Future Works

The TRACE model advances traffic anomaly detection by integrating *Graph Neural Networks (GNNs), Transformers, and Normalizing Flows* to capture complex spatial-temporal dependencies and manage data sparsity. Its hybrid architecture enables real-time processing of large-scale traffic streams, ensuring timely detection and precise localization of anomalies. Moreover, its probabilistic framework maintains robustness even under noisy or incomplete data.

Nonetheless, TRACE has limitations. In highly dynamic traffic environments, its detection rate is lower than that of models like STG-RGCN, highlighting challenges with rapid fluctuations. Additionally, although its detection delay is competitive, it is slightly higher than that of GCN, suggesting a need for improved computational efficiency. Finally, TRACE struggles to generalize in regions with high traffic variability, resulting in performance inconsistencies.

Moving forward, several key areas could enhance TRACE's effectiveness:

- Adaptive Detection Thresholds: Implementing dynamic thresholding mechanisms that adjust in real time based on evolving traffic patterns could improve sensitivity to anomalies.
- Multimodal Data Integration: Incorporating additional contextual data, such as weather conditions, roadwork schedules, and large-scale event information, could enhance robustness and adaptability in unpredictable scenarios.
- Uncertainty Quantification: Further integrating probabilistic techniques to provide confidence scores for detections could improve reliability in decision-making processes for traffic management systems.

By addressing these areas, TRACE has the potential to become a **transformative tool for transportation agencies**, improving road safety, alleviating congestion, and accelerating incident response. Future iterations of TRACE should focus on optimizing scalability, ensuring that the model can be effectively deployed in diverse real-world traffic environments.

Acknowledgments

This research was partially supported by the NSF SCC-CIVIC-PG Track B Grant (#2431246) and by the Tennessee

Department of Transportation (TDOT) Grant (RES2023-17). We would also like to thank the Transportation Cyber Physical Systems Lab at the Institute for Software Integrated Systems, Vanderbilt University, for their invaluable assistance with data collection and processing. Results presented in this paper were obtained using the Chameleon Testbed supported by the NSF.

References

[1] R. Sánchez-Mangas, A. García-Ferrrer, A. De Juan, and A. M. Arroyo, "The probability of death in road traffic accidents. how important is a quick medical response?" *Accident Analysis & Prevention*, vol. 42, no. 4, pp. 1048–1056, 2010.

[2] H. J. Haule, T. Sando, R. Lentz, C.-H. Chuan, and P. Alluri, "Evaluating the impact and clearance duration of freeway incidents," *ScienceDirect*, 2018.

[3] C. Ding, X. Ma, Y. Wang, and Y. Wang, "Exploring the influential factors in incident clearance time: Disentangling causation from self-selection bias," *ScienceDirect*, 2015.

[4] C. Liu, "Exploration of the police response time to motor-vehicle crashes in pennsylvania, usa," *ScienceDirect*, 2021.

[5] K. G. Zografos, K. N. Androutsopoulos, and G. M. Vasilakis, "A real-time decision support system for roadway network incident response logistics," *ScienceDirect*, 2000.

[6] U.S. Department of Transportation (DOT), "Challenges in Traffic Incident Localization and Response Efficiency," 2021.

[7] Federal Highway Administration (FHWA), "Secondary Crashes: Understanding the Scope and Mitigation Strategies," 2019.

[8] A. Coursey, J. Ji, M. Quinones-Grueiro, W. Barbour, Y. Zhang, T. Derr, G. Biswas, and D. B. Work, "FT-AED: Benchmark Dataset for Early Freeway Traffic Anomalous Event Detection," Jun. 2024, arXiv:2406.15283 [cs].

[9] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *CoRR*, vol. abs/1609.02907, 2016.

[10] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2018.

[11] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *The Semantic Web*. Springer, 2018, pp. 593–607.

[12] H. Laman, S. Yasmin, and N. Eluru, "Joint modeling of traffic incident duration components (reporting, response, and clearance time): A copula-based approach," *Transportation Research Record*, vol. 2672, no. 30, pp. 76–89, 2018.

[13] P. P. Tasgaonkar, R. D. Garg, and P. K. Garg, "Vehicle detection and traffic estimation with sensors technologies for intelligent transportation systems," *Sens. Imaging*, vol. 21, no. 1, p. 29, 2020.

[14] D. A. Badoe, S. M. Click, J. Oswalt, M. Edwards, E. Humphreys, and T. T. University, "The traffic monitoring program research project: Comparison of tdot practices with selected states," Tennessee Technological University, Tech. Rep. RES2016-01, 2017.

[15] S. Thakur, "A review of traffic congestion problem and various automated traffic measurement sensors and techniques," *Indian J. Sci. Technol.*, vol. 9, no. 1, pp. 1–16, 2016.

[16] U. S. F. H. Administration, "Traffic monitoring guide - updated october 2016," Tech. Rep. FHWA-PL-17-003, 2016.

[17] "What is Transportation Systems Management and Operations (TSMO)? | Transportation Systems Management and Operations (TSMO) Plans | Organizing and Planning for Operations - FHWA Office of Operations."

[18] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A Comprehensive Survey on Graph Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, Jan. 2021, conference Name: IEEE Transactions on Neural Networks and Learning Systems.

[19] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, "DNN-based prediction model for spatio-temporal data," in *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ser. SIGSPACIAL '16, Oct. 2016, pp. 1–4.

[20] B. Yu, H. Yin, and Z. Zhu, "Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting," pp. 3634–3640, 2018.

[21] W. Jiang and J. Luo, "Graph Neural Network for Traffic Forecasting: A Survey," *Expert Systems with Applications*, vol. 207, p. 117921, Nov. 2022.

[22] Z. Xing, M. Huang, W. Li, and D. Peng, "Spatial linear transformer and temporal convolution network for traffic flow prediction," *Scientific Reports*, vol. 14, no. 1, p. 4040, Feb. 2024, publisher: Nature Publishing Group.

[23] D. P. Kingma and P. Dhariwal, "Glow: Generative Flow with Invertible 1x1 Convolutions," Jul. 2018, arXiv:1807.03039 [stat].

[24] M.-F. Chiang, E.-P. Lim, W.-C. Lee, and A. T. Kwee, "BTCI: A new framework for identifying congestion cascades using bus trajectory data," in *2017 IEEE International Conference on Big Data (Big Data)*. Boston, MA: IEEE, Dec. 2017, pp. 1133–1142.

[25] M. L. D. Dias, C. L. C. Mattos, T. L. C. da Silva, J. A. F. de Macedo, and W. C. P. Silva, "Anomaly detection in trajectory data with normalizing flows," *arXiv*, April 13 2020.

[26] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real NVP," *CoRR*, vol. abs/1605.08803, 2016.

[27] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," 2019.