IEEE Open Journal of
Intelligent Transportation Systems

Received 28 June 2025; revised 21 October 2025; accepted 24 November 2025. Date of publication 26 November 2025; date of current version 2 December 2025.

Digital Object Identifier 10.1109/OJITS.2025.3637305

Advanced Prediction of Traffic at Different Temporal Scales Using Heterogeneous Data Sources

IVÁN GÓMEZ [1] AND SERGIO ILARRI [1,2]

[1] Department of Computer Science and Systems Engineering, Universidad de Zaragoza, 50018 Zaragoza, Spain
[2] Aragón Institute of Engineering Research, Universidad de Zaragoza, 50018 Zaragoza, Spain

CORRESPONDING AUTHOR: I. GÓMEZ (e-mail: ivan.gomez@unizar.es)

This work was supported in part by the Project PID2020-113037RB-I00 (NEAT-AMBIENCE—Next-Generation Data Management to Foster suitable Behaviors and the Resilience of Citizens against Modern Challenges), funded by MICIU/AEI/10.13039/501100011033 and led by Sergio Ilarri, and in part by the Departamento de Ciencia, Universidad y Sociedad del Conocimiento del Gobierno de Aragón (Government of Aragon: Group Reference T64_23, COSMOS Research Group).

ABSTRACT Efficient urban traffic management is a crucial challenge in modern smart cities, especially in densely populated areas with complex and dynamic traffic conditions. In this paper, we tackle the traffic prediction problem and present a lightweight architecture that combines sensor embeddings with dense layers, sustaining strong performance across both short- and long-term forecasting horizons while substantially reducing training time and enabling fast inference times. In comparative evaluations, our approach matches or surpasses the accuracy of more complex methods and consistently improves efficiency. To foster reproducibility, we release the code along with an enriched dataset that integrates traffic flows with contextual features such as weather conditions, temporal variables, and urban attributes. The richness and coverage of this dataset exceed those of existing public resources, enabling deeper and more comprehensive analyses of traffic dynamics. Overall, we demonstrate that a lightweight, well-designed architecture can achieve high performance and practical scalability for urban mobility management.

INDEX TERMS Traffic prediction, traffic datasets, urban traffic management, heterogeneous data sources, traffic data analysis.

I. INTRODUCTION

URBAN traffic management, a key area within Intelligent Transportation Systems (ITS), has become critical for planning in dense metropolitan regions, where ongoing urban expansion and motorization steadily increase travel demand. In this context, traffic prediction is a central enabler for tactical planning and operational decision-making: it allows agencies to anticipate congestion, evaluate traffic restrictions in critical areas, and design priority routes for emergency vehicles, with direct effects on productivity, emissions, air quality, and ultimately public health [1], [2]. The growing availability of data –from pervasive sensor networks, mobile sources, and open platforms– creates both an opportunity and a challenge. It demands robust, scalable predictive models capable of turning large data volumes into effective operational decisions [3], [4].

Efficiently integrating and processing large volumes of heterogeneous data (e.g., traffic flow, meteorology, and urban characteristics) remains a central challenge for producing reliable, actionable predictions across multiple time horizons. At the city scale, traffic dynamics are nonlinear and context-dependent; signals are heterogeneous (temporal, spatial, meteorological, and infrastructural), and sensor networks are extensive and evolving. Many approaches (i) fail to fuse diverse sources, (ii) lose accuracy when the horizon extends beyond a few hours, and (iii) are not computationally efficient. This work addresses these gaps with an architecture that combines expressive representations of sensor data and urban context with scalable learning components. We evaluate it on data from the city of Madrid, Spain, for multi-horizon intensity forecasting using heterogeneous inputs. The main contributions of this work to the ITS field are:

1) *Development of a traffic model that balances accuracy and computational efficiency*. By using a neural network architecture that combines embedding layers (to capture sensor-specific characteristics) with dense layers (to model complex nonlinear relationships), the model achieves improved performance at lower

The review of this article was arranged by Associate Editor Jun Liu.

TABLE 1. **Surveyed traffic-forecasting studies by forecast horizon, data sources, scenario, model family, and code availability.**

Work	Forecast horizon	Data Sources	Scenario	Model	Availability
[14]	5–60 min (1–12 steps)	Traffic data	Highway sensor networks (METR-LA and PEMS-BAY; California, USA)	Diffusion graph convolution for directed graphs combined with GRU-based temporal modeling (GRU: Gated Recurrent Unit)	✓
[15]	5–60 min (1–12 steps)	Traffic data	Highway sensor networks (PeMS; California, USA)	Graph neural ODE (Ordinary Differential Equation) with spatial and semantic adjacency integrated with dilated temporal convolution networks	✓
[16]	12/24/48/72 h (1 step)	Traffic data, weather, and labor calendar	Urban sensor network (4 target + 2 aux/target; Madrid, Spain)	Convolutional network combined with bidirectional LSTM (Long Short-Term Memory)	✓
[17]	5–60 min (1–12 steps; traffic)	Traffic data	Highway sensor networks (PeMS; California, USA)	Adaptive graph sparsification of the learned adjacency with node-dependent hard-concrete masking and a temporal module using either AGCRN (Adaptive Graph Convolutional Recurrent Network) or AGFormer (Adaptive Graph Transformer).	×
[18]	5–120 min (24 steps; PeMS); 15–180 min (12 steps; LargeSt)	Traffic data	Highway sensor networks (PeMS and LargeST; California, USA)	Dynamic graph generation via cross-attention with time-evolving adjacency and localization personalization	✓
[19]	15–180 min (1–12 steps)	Traffic data	Highway sensor networks (LargeSt; California, USA)	Transformer model using patch embeddings and hierarchical attention	✓
[20]	Next interval (fixed grid map)	Traffic (GPS trajectories), weather (TaxiBJ only), labor calendar	Urban grid, citywide inflow/outflow (TaxiBJ: Beijing, China; NYCBike: New York, USA)	Hybrid model with GCN (Graph Convolutional Network) and LSTM integrating ConvLSTM (Convolutional Long Short-Term Memory) branches	×
[21]	Next interval (fine-grained map)	Traffic data (GPS trajectories), weather (TaxiBJ only), labor calendar	Urban grid, citywide inflow/outflow (BJTaxi: Beijing, China; NYCBike: New York, USA)	CNN–ConvLSTM with spatio-temporal attention (CNN: Convolutional Neural Network)	×
[22]	Next interval (fixed grid map)	Traffic (GPS trajectories), weather, labor calendar	Urban grid, citywide inflow/outflow (TaxiBJ: Beijing, China; BikeNYC: New York, USA)	Residual CNN with ConvLSTM-based CNN–LSTM branches	×
[23]	Next interval (fixed grid map)	Traffic (GPS trajectories), weather (TaxiBJ only), labor calendar	Urban grid, citywide inflow/outflow (TaxiBJ: Beijing, China; BikeNYC: New York, USA)	ConvLSTM-based multi-branch model with attention mechanism	×
TrafficDatorNet (ours)	1–384 h (1–384 steps; 1 h step)	Traffic, weather, urban attributes, and labor calendar	Urban sensor network (Madrid Traffic Dataset; Madrid, Spain)	Lightweight dense neural model with embedding layer and dense layers	✓

computational cost, ensuring scalability to large sensor networks.

2) *Predictions at different time horizons*. Unlike traditional models, which limit their forecast to a few hours due to high computational costs and reduced accuracy, our model maintains high precision and efficiency at both short- and long-term horizons.
3) *Evaluation of various models*. Evaluation of both classical and state-of-the-art traffic prediction models identifies key challenges and opportunities to enhance accuracy, efficiency, and scalability. Testing all models on the same dataset ensures a clear and consistent performance comparison.
4) *Introduction and comparison of a rich multi-source dataset*. We introduce the *Madrid Traffic Dataset* and compare it with other public datasets, emphasizing its superior coverage and information richness. It integrates temporal and spatial traffic data, weather conditions, and urban features (e.g., flow direction, lanes, road type, speed limits), offering a comprehensive basis for advanced analysis. Its broad temporal span further enables the study of long-term trends and seasonality.
5) *Commitment to the Principles of Reproducibility [5], FATE [6], and FAIR [7]*. This work relies on publicly available data, provides a project website with detailed documentation [8], and makes the source code and experimental configurations openly accessible [9], [10]. Furthermore, the datasets are deposited in Mendeley Data [11], ensuring transparency, accessibility, and reproducibility, while enabling validation and extension by the scientific community.

The rest of this paper is organized as follows. Section II surveys related work on traffic forecasting. Section III describes the adopted methodology, including the development and training of our proposed model, which we call *TrafficDatorNet*. Section IV reviews datasets used for traffic forecasting, including the dataset proposed in this study. Section V outlines the applied models (from traditional methods to advanced architectures) and presents an in-depth experimental evaluation of their accuracy and efficiency. Finally, Section VI summarizes the key contributions of our work and outlines potential directions for future research.

II. RELATED WORK

Traffic prediction is a key issue for the development of intelligent transportation systems and suitable decision making related to mobility. For this reason, it has attracted considerable research attention. Interesting surveys on traffic prediction can be found in [12], [13].

In Table 1, we show a comparative summary of representative related work, highlighting each approach's main features alongside its strengths and weaknesses. Studies are organized column-wise by critical aspects to enable a clear comparison. The columns include: "Temporal Focus", which distinguishes studies by their forecast horizon; "Data Sources", listing the types of inputs (traffic and external variables) used to enhance predictive accuracy; "Scenario", which summarizes geographical coverage and sensing setup; "Model", indicating the forecasting architecture employed; and "Availability", which flags public availability of source code, an important driver of openness and reproducibility in this field. This structured view streamlines cross-paper comparison, surfaces current trends and gaps, and sets the stage for the discussion that follows.

As summarized in Table 1, most graph-centric studies prioritize short-term horizons [14], [15], [17], [18], typically ranging from 5 to 60 minutes and evaluated on compact benchmarks. These works usually rely exclusively on traffic sensor data, omitting exogenous signals that could provide richer contextual information.

In parallel, grid-based crowd-flow research [20], [21], [22], [23] does integrate external signals (e.g., weather, holidays, points of interest) but targets a different task (citywide inflow/outflow on spatial grids). It also relies on spatial aggregation that alters the road network structure, uses metrics not directly comparable to sensor-level traffic, focuses mainly on short-term horizons, and depends on regional trajectory data (e.g., GPS records projected onto city-specific spatial grids).

By contrast, long-term studies [16] are usually restricted to smaller sensor deployments or simplified setups, incorporating exogenous variables such as weather conditions and calendar effects through convolutional or recurrent neural architectures. Similarly, attention-based transformer models [19], although not graph-based, aim to capture long-range temporal dependencies and global correlations; however, they are generally evaluated on short- to medium-term horizons and lack explicit integration of heterogeneous external inputs at the sensor level. It is also worth noting that only about half of these works provide open-source implementations, which still limits transparency and reproducibility within the field. This gap motivates the development of approaches that combine multi-horizon forecasting at the sensor level with the integration of exogenous variables, while emphasizing openness and reproducibility. Accurate traffic prediction can support a variety of applications, not only for urban or mobility planning, but also for other scenarios, like service-migration strategies in vehicular Multi-access Edge Computing (MEC) environments [24] by estimating when vehicles will attach to new edge clouds.

III. METHODOLOGY

This section presents the proposed methodology for traffic prediction. In Section III-A, we formalize the problem of traffic prediction. In Section III-B, we describe our proposed architecture, that we call *TrafficDatorNet*, highlighting its key components and implementation details. Finally, in Section III-C, we explain the approach used for model training, hyperparameter optimization, and evaluation. Figure 1 provides an overview of our methodology, which encompasses the initial data input from diverse sources, subsequent preprocessing, the application of various predictive models including *TrafficDatorNet*, followed by a rigorous training and evaluation phase, and concluding with the generation of traffic predictions.

A. PROBLEM STATEMENT FOR TRAFFIC PREDICTION

Traffic prediction in urban transportation systems is a highly complex due to the dynamic, nonlinear, and multidimensional spatio-temporal interactions that govern traffic behavior. Traffic conditions are influenced by a wide range of heterogeneous data sources, including traffic sensor, meteorological variables, and specific urban characteristics.

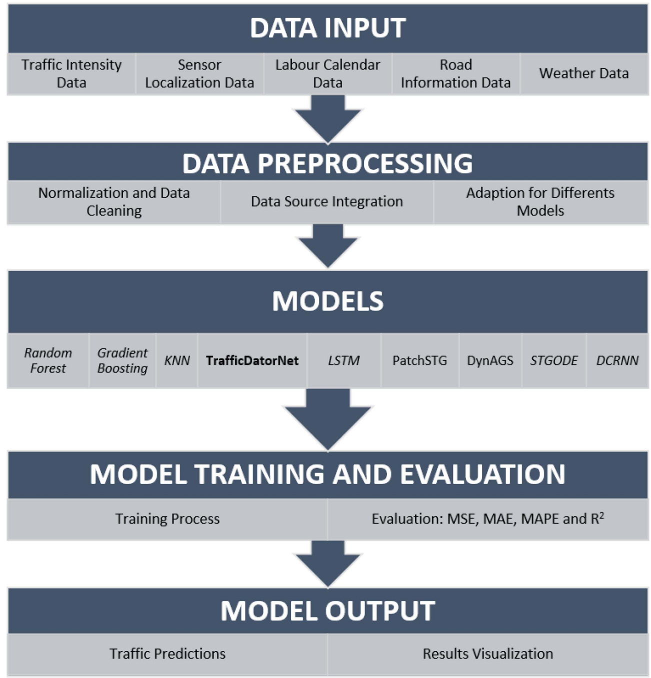

FIGURE 1. Overview of the methodology: data ingestion, preprocessing and integration, model application, evaluation, and forecast generation.

Different inputs must be integrated into a continuous representation that captures temporal and spatial patterns for accurate modeling. In this context, an efficient and optimized solution is required to process large volumes of urban data while maintaining high predictive accuracy, even in dense and complex road networks. Additionally, it should ensure scalability and efficient data management, enabling predictions across multiple temporal horizons to support both short-term control and long-term planning.

We formalize the task as follows. Let $\mathcal{D}$ be a set of multivariate time series (one per location). For series i, we consider $\mathbf{X}_i = \{x_{i1}, \ldots, x_{it}\}$, where each $x_{ij} \in \mathbb{R}^d$ is a preprocessed feature vector (e.g., time, weather, urban attributes); braces $(\{\cdot\})$ denote a time-ordered sequence. Let t be the last observed time index and $k \geq 1$ the forecast horizon. The target $y_{i,t+k} \in \mathbb{R}$ is the traffic intensity for series i at time $t+k$. Our goal is to learn a model $f(\cdot; \Theta)$, with Θ the trainable parameters, such that Equation (1) is satisfied.

$$\hat{y}_{i,t+k} = f(\mathbf{X}_i, k; \Theta). \tag{1}$$

This formulation supports predictions across locations i and horizons k, capturing the granularity and dynamics of urban traffic.

B. OUR PROPOSED ARCHITECTURE: TRAFFICDATORNET

TrafficDatorNet is our proposed architecture. It is based on neural networks designed with a sequence of specialized layers to transform the input space into a continuous output prediction, as depicted in Fig. 2. This architecture integrates

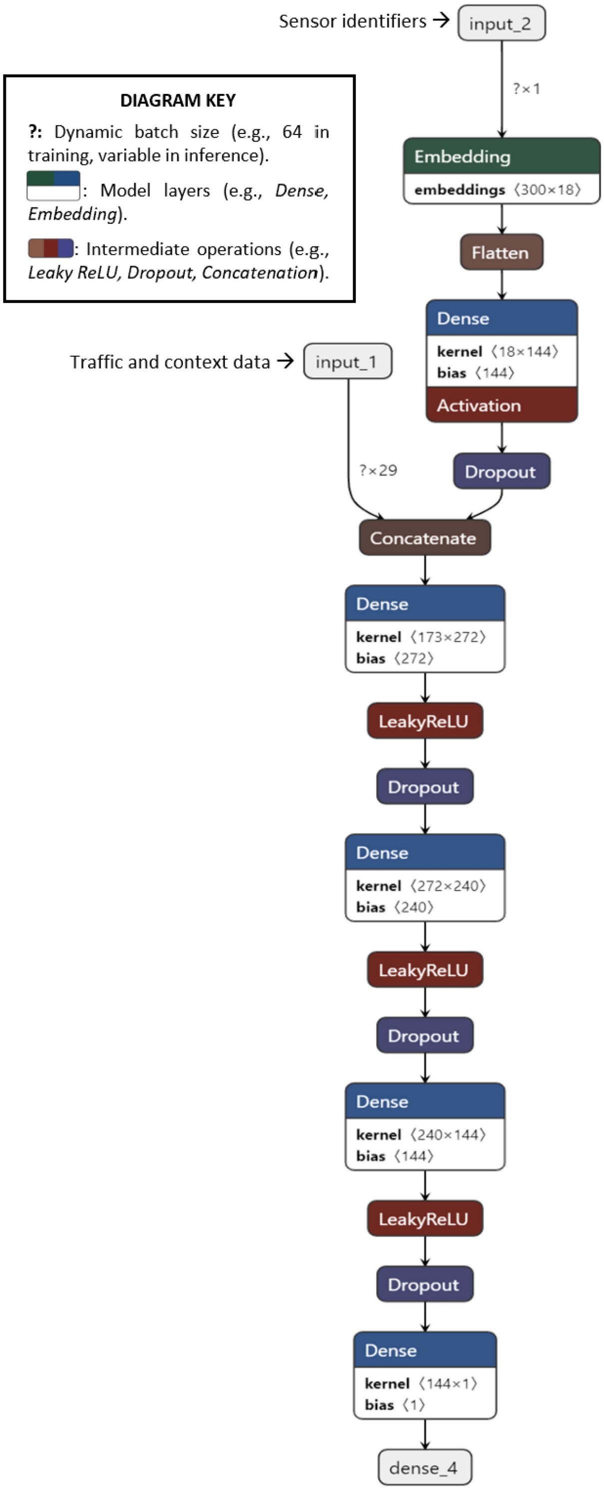

FIGURE 2. Overview of the *TrafficDatorNet* architecture: sensor IDs encoded by embeddings, concatenation with traffic/context variables, three Dense–LeakyReLU–Dropout blocks, and linear output; dimensions indicated in each block.

embedding layers to encode the distinctive traits of each sensor location, while dense layers are employed to model the intricate, nonlinear interactions among all input variables, resulting in a robust traffic flow prediction. The hyperparameter tuning for *TrafficDatorNet* was based on a hybrid strategy in which theoretical principles established plausible ranges for each parameter. Specifically, the embedding size was defined to balance representation capacity and parameter efficiency; the number of units in the dense layers was chosen to ensure sufficient expressiveness without overfitting; the dropout rate was adjusted to regulate model complexity through proper regularization; and the learning rate was set to maintain a balance between convergence speed and training stability. Following this, an automated search strategy refined this selection of hyperparameters to minimize validation errors effectively. This approach allowed for a detailed model adaptation, leveraging both foundational theories of neural network design and systematic exploration to optimize the performance comprehensively across key hyperparameters such as the embedding dimensionality, the number of dense layers and their units, dropout rates, and learning rates.

The first layer is an embedding layer that learns 18-dimensional vector representations for each traffic monitoring location, identified by its unique sensor identifier $s \in S$. The embedding function is defined as

$$E\colon S \to \mathbb{R}^{18}, \tag{2}$$

where each identifier s is mapped to its corresponding learnable embedding vector $E(s)$. Numerical features are processed by three dense layers. For $i \in \{1, 2, 3\}$, each layer computes the following:

$$\mathbf{h}_i = \phi(W_i \mathbf{h}_{i-1} + \mathbf{b}_i), \tag{3}$$

where $\mathbf{h}_0 = \mathbf{x}$ is the input vector (numerical features), W_i is the weight matrix of layer i, $\mathbf{b}_i$ the bias vector, and ϕ is the LeakyReLU activation, applied element-wise and defined as follows:

$$\phi(z) = \max(\alpha z,\ z), \tag{4}$$

with $\alpha = 0.05$ being a small coefficient that allows nonzero gradients for negative inputs. These three layers provide sufficient depth to capture complex patterns without excessively increasing overfitting [25]. Here, i indexes layers in the equations above, while in the dropout and loss formulas below i indexes training samples.

To further mitigate overfitting, we insert a dropout layer between hidden layers. During training, for the activation vector x_i (sample i), the dropout transform is given by:

$$D_{\text{drop}}(x_i) \;=\; x_i \odot m_i, \qquad m_i \sim \text{Bernoulli}(1-p), \tag{5}$$

where $p \in [0, 1)$ is the drop probability, m_i is a random mask with independent entries, and $\odot$ denotes element-wise multiplication.

The architecture ends with a single linear output unit (regression of traffic intensity). Its prediction is given by:

$$\hat{y} \;=\; W_o D_n + b_o, \tag{6}$$

where $D_n \equiv \mathbf{h}_3$ is the output of the last dense layer, W_o and b_o are the output-layer weights and bias, respectively, and $\hat{y}$ is the estimated traffic intensity.

To build the model, a training method based on previous data must be carried out. Training minimizes the mean squared error (MSE):

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{y}_i - y_i\right)^2, \tag{7}$$

where N is the number of training samples, i indexes the samples, y_i is the true traffic intensity (target), and $\hat{y}_i$ is the model prediction. All trainable parameters are collected in Θ see Equation 1.

C. APPROACH FOR MODEL TRAINING AND EVALUATION

To address the complexity of urban traffic patterns, it is essential to train the model on a dataset that captures a representative mix of conditions across different types of days (e.g., weekdays, weekends, holidays). This ensures that the model can generalize effectively and predict traffic patterns accurately in diverse scenarios. In this work, we specifically used the dataset presented in Section IV-B. The training of *TrafficDatorNet* involves dividing the data into 60% for training, 20% for validation, and 20% for testing. Before feeding the data into the model, numerical features are scaled and categorical variables are encoded. This split ensures that each subset contains a representative mix of data across different types of days, enabling a thorough evaluation and optimization process to accurately predict traffic patterns from multiple sensors.

Parameter optimization used Adam [26]. Let t index the training step, and let W_t be the vector collecting all trainable weights at step t. Adam updates the vector W as follows:

$$W_{t+1} = W_t - \eta\,\frac{m_t}{\sqrt{v_t} + \epsilon}, \tag{8}$$

where m_t and v_t are the bias-corrected first-moment and second-moment estimates of the instantaneous gradient $g_t = \nabla_W \mathcal{L}_t$, $\eta > 0$ is the learning rate, $\epsilon > 0$ is a small stabilizer, and both the square root and the division are applied element-wise. Here, $\mathcal{L}_t$ denotes the training loss (MSE) on the minibatch at step t, and m_t and v_t are Adam's bias-corrected exponential moving averages of the gradient and its element-wise square.

After training, in our implementation, as we used Keras [27], the model of *TrafficDatorNet* was stored in the *.keras* format together with the pipeline artifacts (preprocessing and encoders). This packaging enables simple deployment across different environments (e.g., behind a lightweight API). If new months of data need to be incorporated, the model can be updated through short retraining sessions on the saved version, reusing *EarlyStopping* and checkpoints while maintaining the same pipeline and artifact versioning.

IV. DATASETS USED IN TRAFFIC FORECASTING

Accurate traffic forecasting models rely heavily on high-quality, comprehensive datasets that capture the complexities of real-world traffic dynamics. Over the years, several benchmark datasets have been established to facilitate research in this domain. However, many of these datasets exhibit limitations in terms of scale, temporal coverage, and the richness of integrated data types, which can impede the development of robust and scalable traffic forecasting models. In the following, in Section IV-A, we provide an overview of prominent existing traffic datasets and discuss their limitations. Then, in Section IV-B, we introduce the *Madrid Traffic Dataset* used in this work, including its features, preprocessing steps, and advantages. Finally, in Section IV-C, we present a comparative analysis highlighting its contributions and versatility compared to other datasets.

A. OVERVIEW OF PREVIOUS TRAFFIC-RELATED DATASETS

The PeMS (Performance Measurement System) datasets [28], such as PeMS03, PeMS04, PeMS07, and PeMS08, are widely utilized benchmarks in traffic forecasting research. These datasets typically encompass hundreds of sensors with limited temporal coverage, often spanning less than six months. Each dataset varies in geographic coverage and number of sensors, making them suitable for studying regional, short-term traffic trends rather than comprehensive, long-term analysis. For example, PeMS07 includes 883 sensors with traffic flow data collected over approximately four months. With a geographic coverage of $15,296.4$ km^2, the sensor density is approximately 0.058 sensors/km^2. Additionally, they focus primarily on traffic flow data, with little (or even absent) integration of other types of data.

The METR-LA and PEMS-BAY datasets, both introduced in the same study [14], provide traffic data for Los Angeles and the San Francisco Bay Area, respectively. METR-LA includes data from 207 sensors with a sensor density of 0.0165 sensors/km^2, while PEMS-BAY comprises data from 325 sensors with a density of 0.018 sensors/km^2. Both datasets are limited in sensor count and temporal coverage, which restricts their applicability to large-scale urban networks or long-term forecasting. Furthermore, neither datasets integrates complementary data, such as weather conditions or urban characteristics, which reduces their ability to capture the complexity of urban traffic dynamics.

The LargeST dataset [29] consists of $8,600$ sensors distributed across the state of California, which covers an area of approximately $423,970$ km^2, resulting in a sensor density of 0.0203 sensors/km^2. It features a temporal coverage of five years and provides detailed metadata for each sensor, including information on road categories and the number of lanes, enabling more detailed traffic network analysis.

These existing benchmarks, despite their notable contributions, highlight a persistent need for datasets that not only offer substantial temporal and spatial coverage but also integrate a richer variety of data types, including granular environmental data and dynamic urban event information.

Such multifaceted datasets are essential for developing the next generation of traffic forecasting models capable of achieving higher accuracy and providing deeper insights into the complex interplay of factors affecting urban mobility.

B. MADRID TRAFFIC DATASET

In this work, we utilized an updated version of the *Madrid Traffic Dataset*, a rich dataset that we compiled [30], after extending and optimizing it for compatibility with various traffic prediction models. This revised dataset, collected from a network of 554 urban and highway sensors between June 2022 and November 2024, records traffic intensity in 15-minute intervals, expressed in vehicles per hour. Raw data from Madrid's Open Data Portal [31] were processed to remove inconsistencies, outliers, duplicates, and low-activity sensors. With a coverage area of approximately $8,030$ km^2 (Community of Madrid), the sensor density is about 0.069 sensors/km^2.

Additionally, it includes contextual variables, such as historical weather conditions, including temperature, precipitation, and wind speed, obtained from Madrid's Open Data Portal [32], as well as road infrastructure features obtained from OpenStreetMap [33] using tools like OSMnx [34] to extract and analyze street networks, and KD-Tree [35] to perform efficient spatial searches and determine the location of sensors on different roads. These include lane count, road directionality, street types, speed limits, and segment lengths. During preprocessing, temporal features (month, day of week, and hour) were extracted and encoded using trigonometric functions to capture cyclical patterns, as described in [30]. Categorical variables were transformed with techniques like OneHotEncoder [36] for nominal variables and OrdinalEncoder [36] for ordinal variables. Continuous variables were normalized with StandardScaler [36], ensuring equal contributions to the model without scale bias.

For this work, we extended the original version of our dataset with additional months of data and incorporated all available sensors. Additionally, adjustments were made to adapt the dataset to various types of traffic prediction models, each with specific requirements regarding data inputs and model structure. To achieve this, an adjacency matrix was generated based on the geographic distances between sensors, using the Haversine formula [37]. This matrix represents the distance from each sensor to every other sensor, limited to a maximum of 4 kilometers to focus on relevant spatial relationships. Subsequently, these distances were transformed into connectivity weights by applying a Gaussian kernel [38] and setting a threshold to eliminate insignificant connections. Additionally, input sequences and labels were generated for supervised models, incorporating features such as the time of day and day of the week, allowing the models to capture recurring temporal patterns and diurnal variations in traffic intensity. All configurations and processing details are available in the accompanying code repository, which is linked from TrafficDator's website [8] for reproducibility and further exploration.

C. COMPARISON OF TRAFFIC DATASETS: FOCUS ON THE MADRID TRAFFIC DATASET

Table 2 compares the *Madrid Traffic Dataset* with other datasets previously mentioned and explained, in terms of study region, sensor density, temporal coverage, and integrated data types. PeMS07 and METR-LA, with sensor densities of 0.058 and 0.0165 sensors/km^2 respectively, have limited temporal coverage (less than six months) and provide only basic sensor location data, restricting their applicability for long-term studies or complex urban networks. By contrast, the LargeST dataset represents a considerably more robust alternative for certain types of analysis. With an impressive count of $8,600$ sensors and five years of data, it is well-suited for extensive analyses due to its scale and the consideration of infrastructure data, such as road classification and traffic flow direction. However, its sensor density (0.0203 sensors/km^2) is lower than that of the *Madrid Traffic Dataset* (0.069 sensors/km^2) and lacks meteorological data and labor calendar information, making it less suitable for urban-specific dynamics.

In this comparative context, the *Madrid Traffic Dataset* emerges as a particularly outstanding proposal. With 554 sensors distributed across the Madrid region and 30 months of coverage (June 2022 – November 2024), its primary distinguishing value lies in the richness and variety of the data it integrates. This dataset not only includes traffic data but also detailed meteorological conditions (temperature, precipitation, and wind), urban characteristics (flow direction, lanes, road classification, speed limits, one-way constraints, and length), and labor calendar information. This comprehensive integration of data makes it particularly valuable for studies focused on European urban environments. Ultimately, its compelling combination of high sensor density, sufficient temporal coverage, and rich data, establishes it as a versatile and powerful resource for urban traffic studies.

V. EXPERIMENTAL EVALUATION

Accurate traffic prediction requires robust models capable of capturing the complex spatio-temporal dynamics inherent in urban traffic data. This section describes the models applied to the *Madrid Traffic Dataset*, including both baseline models based on traditional machine learning techniques and advanced deep learning architectures, as well as their adaptations to the dataset (Section V-A). Then, we describe the experimental settings and metrics used for evaluation (Section V-B). Additionally, we present the experimental results, highlighting the advantages of *TrafficDatorNet* in both short- and long-term predictions (Section V-C). Finally, we include a detailed example scenario (Section V-D) that demonstrates the robustness of *TrafficDatorNet* for long-term prediction scenarios, applied to a strategically located traffic sensor in Madrid.

TABLE 2. **Comparison of traffic forecasting datasets. We present several features (region, number of sensors, temporal coverage, types of variables included, total datapoints, and sensor density) for popular benchmarks and our proposed *Madrid Traffic Dataset.***

Dataset	Region	Sensors	Temporal Coverage	Types of Data Included	Data Points	Sensor Density (sensors/km^2)
PeMS07	California, USA	883	4 months (May–Aug 2017)	Sensor Location Data	28K	0.058
METR-LA	Los Angeles, USA	207	4 months (Mar–Jun 2012)	Sensor Location Data	34K	0.0165
PEMS-BAY	Bay Area, USA	325	6 months (Jan–May 2017)	Sensor Location Data	52K	0.018
LargeST	California, USA	8,600	5 years (Jan 2017–Dec 2021)	Sensor Location Data Road Information: district county road classification lanes type of sensor flow direction	4.52B	0.0203
Madrid Traffic Dataset (ours)	**Madrid, Spain**	**554**	**30 months (Jun 2022–Nov 2024)**	**Sensor Location Data** **Labor Calendar** **Weather:** **temperature** **precipitation** **wind** **Urban Characteristics:** **flow direction** **lanes** **road classification** **speed limits** **one-way** **length**	**45K**	**0.069**

A. MODELS USED FOR OUR EXPERIMENTAL COMPARISON

To establish a solid benchmark for traffic prediction, we tested a variety of traditional machine learning methods and recent deep learning models, selected for their effectiveness in prior studies. For a comprehensive analysis, we selected state-of-the-art baseline models, described in the following:

- *LSTM (Long Short-Term Memory Network)* [39] is a type of recurrent neural network (RNN) [40] designed to capture temporal patterns in sequential data. Our implementation builds on frameworks from previous research [41].
- *DCRNN (Diffusion Convolutional Recurrent Neural Network)* [14] is an advanced hybrid model that combines graph neural networks to capture spatial dependencies with recurrent networks to model temporal relationships. DCRNN simulates how traffic conditions propagate through a network of sensors using graph-based diffusion processes.
- *STGODE (Spatio-Temporal Graph Ordinary Differential Equation)* [15] is an advanced approach that uses ordinary differential equations [42] to model continuous spatio-temporal dynamics. Inspired by recent advances in spatio-temporal graph networks, STGODE has been customized to efficiently handle large-scale urban traffic data and is particularly effective at capturing nonlinear traffic patterns, offering flexibility and precision.
- *PatchSTG (Patch-based Spatio-Temporal Transformer)* [19] is a recent transformer-based approach that tackles the quadratic cost of dynamic spatial modeling by partitioning irregularly distributed sensors into balanced, nonoverlapping patches via KD-Tree partitioning; its dual-attention encoder alternates between intra-patch (depth) and inter-patch (breadth) attention to capture both local and global dependencies. PatchSTG has established itself as a strong reference model for urban traffic forecasting due to its balance between efficiency and accuracy.
- *DynAGS (Dynamic Localisation of Spatial-Temporal Graph Neural Network)* [18] is a dynamic localization framework designed to enhance the efficiency and flexibility of spatio-temporal graph networks. Built upon cross-attention mechanisms, DynAGS dynamically constructs adaptive and sparse graphs over time, allowing each node to selectively interact with the most relevant neighbors (i.e., other graph nodes that maintain significant spatio-temporal dependencies with

it). This dynamic localization mechanism balances accuracy and computational efficiency, and can be seamlessly integrated with various spatio-temporal forecasting backbones, establishing itself as a reference architecture for distributed urban data modeling.

These models represent the state-of-the-art and the most widely adopted approaches in traffic prediction, supported by a strong foundation of research and development in the field of intelligent traffic management and forecasting. To complement our analysis, traditional machine learning methods were also considered to provide a broad performance spectrum and evaluate the trade-offs between complexity and efficiency:

- *Random Forest (RF)* [43], an ensemble learning method that combines multiple decision trees. Although efficient for regression, it has limitations in modeling complex spatio-temporal dependencies. However, recent studies show that it is still used in combination with other models to improve the accuracy in problems like traffic prediction [44].
- *Gradient Boosting (GB)* [45], a boosting technique that builds trees sequentially, correcting errors progressively. It performs well on structured data, although it is limited in capturing spatio-temporal relationships. It has been used in recent studies as an example for traffic prediction applications [46].
- *K-Nearest Neighbors (KNN)* [47], a nonparametric algorithm that predicts traffic based on values of nearby neighbors. It is a simple and useful technique for comparison in traffic studies, especially in traffic prediction, supported even by recent research [48].

A unified framework was developed to standardize the processes of data preparation, model training, and evaluation, keeping all parameters constant except for the specific characteristics of each model's architecture. This facilitates and supports our experimental evaluation and comparison.

B. EXPERIMENTAL SETTINGS AND METRICS

The experiments were conducted on a computer with an 11$^{\text{th}}$ Gen Intel Core i7-1165G7 processor at 2.80 GHz and 16 GB of RAM (CPU-only, no discrete GPU). We used a tailored subset of the *Madrid Traffic Dataset* (described in Section IV-B) with data from 300 sensors over 17 months (June 2022–October 2023), chosen to balance computational efficiency and predictive performance. Models were implemented in Python: *TrafficDatorNet* with Keras [27] and TensorFlow [49]; LSTM, DCRNN, STGODE, PatchSTG and DynAGS with PyTorch [50]; and Random Forest, Gradient Boosting, and KNN with scikit-learn [36]; preprocessing (encoding/normalization) also used scikit-learn.

All implementations followed a common experimental pipeline described for *TrafficDatorNet*: the same subset of the *Madrid Traffic Dataset*, the same 60/20/20 partition (training/validation/test), and a unified preprocessing. Additionally, we used Adam, early stopping with best-model checkpointing, and fixed the random seed for reproducibility. Each family of models was minimally adapted to the specific characteristics of its architecture: sequential models operate with time windows, graph-based models rely on the same adjacency matrix, and classical models (RF/GB/KNN) must be trained on the transformed features.

Model performance was evaluated using several metrics: the *Mean Squared Error (MSE)*, which is the average of squared errors that penalizes larger deviations; the *Mean Absolute Error (MAE)*, which is the average of absolute errors and reflects the typical prediction error; the *Mean Absolute Percentage Error (MAPE)*, which is the average absolute error expressed in percentage terms to allow scale-free comparisons; and the *Coefficient of Determination* (R^2), which is the proportion of variance explained by the model. For MSE/MAE/MAPE, lower values indicate better performance, whereas for R^2 higher values are better. Additionally, training and inference times were used to assess computational efficiency and model feasibility. While initialization/training time reflects either setting up the model (e.g., for KNN) or actively training it (e.g., Random Forest or *TrafficDatorNet*), the inference time indicates suitability for real-time applications, providing a comprehensive view of practical performance.

All neural models were evaluated with Monte Carlo (MC) [51] at test time. We kept dropout active and ran $R = 5$ stochastic inference passes; for each prediction and metric horizon we reported the mean $\pm$ one standard deviation (SD), computed over these runs. For tree-ensemble baselines (Random Forest), predictive variability was estimated through bootstrap resampling of the base learners at inference time, averaging their outputs; results were summarized using the same statistics (mean and standard deviation). In contrast, deterministic baselines (Gradient Boosting, KNN) produced a single prediction (as there is no variation across repetitions, due to their lack of internal stochasticity), which was directly reported.

C. EXPERIMENTAL RESULTS

Selecting appropriate prediction horizons is crucial in urban traffic forecasting, as it directly impacts the projection accuracy and, consequently, strategic planning and decision-making. Table 3 presents the results in terms of MAE, RMSE, MAPE, and R^2 across different prediction horizons (each time step represents 1 hour) for the models evaluated on the *Madrid Traffic Dataset*.

Models such as LSTM, DCRNN, PatchSTG and STGODE have shown strong performance in short-term predictions, typically over 3, 6, and 12 time steps, as noted in previous studies [29], [52], [53]. However, predicting over longer time horizons is more challenging. There are two main approaches to address this: DirRec (Direct-Recursive) and MIMO (Multiple-Input Multiple-Output) [54].

In DirRec, the output of one step is used as input for the next, which can lead to error accumulation as the prediction horizon extends, significantly affecting accuracy.

TABLE 3. Experimental comparison of prediction methods: performance across horizons from 3 to 12 steps (1 step = 1 h); mean ± SD over 5 runs.

Horizon	MAE	RMSE	MAPE (%)	R^2
LSTM				
3	68.4 ± 0.7	137.7 ± 2.6	25.3 ± 0.5	0.84 ± 0.01
6	68.0 ± 0.3	148.0 ± 5.0	53.4 ± 0.9	0.53 ± 0.03
12	82.3 ± 1.0	157.1 ± 5.0	25.7 ± 0.4	0.76 ± 0.01
Avg	69.3 ± 1.2	144.0 ± 5.7	38.3 ± 0.7	0.62 ± 0.04
DCRNN				
3	85.2 ± 5.1	181.8 ± 14.2	27.7 ± 1.4	0.73 ± 0.05
6	59.9 ± 3.7	147.4 ± 8.0	39.9 ± 1.7	0.49 ± 0.06
12	116.1 ± 10.7	207.6 ± 33.0	33.2 ± 1.2	0.55 ± 0.14
Avg	77.4 ± 5.4	169.8 ± 15.0	32.3 ± 1.4	0.51 ± 0.07
STGODE				
3	53.4 ± 0.9	136.5 ± 0.5	20.4 ± 0.6	0.82
6	37.1 ± 0.4	125.8 ± 0.8	42.2 ± 1.7	0.66
12	55.3 ± 0.4	98.1 ± 1.6	14.5 ± 0.2	0.88
Avg	48.0 ± 0.5	127.3 ± 0.9	24.9 ± 0.8	0.70
PatchSTG				
3	60.2 ± 1.6	135.2 ± 1.4	27.6 ± 1.6	0.84
6	42.7 ± 0.8	128.4 ± 1.1	44.2 ± 3.7	0.64 ± 0.01
12	63.9 ± 0.9	122.8 ± 1.6	19.3 ± 1.0	0.85
Avg	54.4 ± 1.1	128.2 ± 1.9	37.0 ± 2.1	0.71 ± 0.01
DynAGS				
3	78.2 ± 8.1	200.9 ± 22.0	20.8 ± 0.9	0.65 ± 0.08
6	61.1 ± 4.7	146.2 ± 8.5	31.8 ± 1.2	0.47 ± 0.06
12	78.5 ± 9.4	170.5 ± 29.5	18.7 ± 1.5	0.71 ± 0.11
Avg	67.1 ± 6.5	164.5 ± 19.0	24.4 ± 1.3	0.53 ± 0.10
Random Forest				
3	176.2 ± 0.3	273.6 ± 0.3	232.8 ± 5.5	0.38
6	103.2 ± 0.1	181.6 ± 0.3	183.8 ± 0.7	0.29
12	198.7 ± 0.3	254.8 ± 0.3	69.7 ± 0.1	0.37
Avg	138.3 ± 0.2	212.6 ± 0.3	116.3 ± 0.8	0.30
Gradient Boosting				
3	103.6	182.4	99.8	0.73
6	69.5	129.0	77.1	0.64
12	92.6	135.1	30.8	0.82
Avg	91.5	154.8	77.2	0.59
KNN				
3	70.5	142.9	24.5	0.83
6	53.0	134.2	41.3	0.61
12	95.1	140.3	24.2	0.81
Avg	71.4	138.8	31.5	0.67
TrafficDatorNet (ours)				
3	$\mathbf{68.2 \pm 2.2}$	$\mathbf{140.3 \pm 12.2}$	$\mathbf{42.2 \pm 9.5}$	$\mathbf{0.85 \pm 0.03}$
6	$\mathbf{47.6 \pm 2.1}$	$\mathbf{125.3 \pm 7.4}$	$\mathbf{44.2 \pm 4.6}$	$\mathbf{0.66 \pm 0.04}$
12	$\mathbf{65.1 \pm 0.9}$	$\mathbf{100.5 \pm 2.9}$	$\mathbf{18.5 \pm 0.7}$	$\mathbf{0.92 \pm 0.01}$
Avg	$\mathbf{58.9 \pm 2.0}$	$\mathbf{127.9 \pm 8.0}$	$\mathbf{32.3 \pm 3.6}$	$\mathbf{0.71 \pm 0.04}$

In contrast, MIMO avoids error accumulation between steps but faces other challenges, such as increased computational demands and difficulty in capturing long-term dependencies in complex data, which can affect accuracy over extended prediction horizons, as discussed in [55], [56], [57], [58].

TABLE 4. *TrafficDatorNet* test performance across horizons from 3 to 384 steps (1 step = 1 h); mean ± SD over 5 runs.

Horizon	MAE	RMSE	MAPE (%)	R^2
3	68.2 ± 2.2	140.3 ± 12.2	42.2 ± 9.5	0.85 ± 0.03
6	47.6 ± 2.1	125.3 ± 7.4	44.2 ± 4.6	0.66 ± 0.04
12	65.1 ± 0.9	100.5 ± 2.9	18.5 ± 0.7	0.92 ± 0.01
24	76.2 ± 3.2	117.6 ± 4.7	22.4 ± 1.5	0.91 ± 0.01
48	64.7 ± 2.4	108.4 ± 8.9	19.7 ± 2.0	0.90 ± 0.02
96	80.8 ± 1.5	133.3 ± 5.1	23.6 ± 1.1	0.90 ± 0.01
192	63.5 ± 3.1	100.2 ± 6.0	25.9 ± 5.0	0.93 ± 0.01
384	63.9 ± 2.6	120.9 ± 3.4	24.3 ± 2.9	0.88 ± 0.01
Avg	66.3 ± 2.2	118.3 ± 6.3	27.6 ± 3.4	0.87 ± 0.02

TABLE 5. CPU-only efficiency on the *Madrid Traffic Dataset*: initialization/training time (min) and per-inference time (s). The computer was an Intel Core i7-1165G7, with 16 GB RAM.

Model	Initialization/ Training Time (min)	Inference Time (s)
Random Forest	28.42	0.21 ± 0.02
Gradient Boosting	19.85	0.01
KNN	260.64	52.51
LSTM	293.43	0.05 ± 0.02
DCRNN	15,560.08	2.35 ± 0.77
STGODE	2,791.10	0.14 ± 0.12
PatchSTG	1,288.952	0.12 ± 0.04
DynAGS	878.73	0.41 ± 0.15
TrafficDatorNet (ours)	**47.92**	$\mathbf{0.42 \pm 0.08}$

Unlike previous models, *TrafficDatorNet* does not experience significant degradation over longer horizons, demonstrating robustness and flexibility for both short- and long-term predictions. This is achieved through a feature-based model that performs direct forecasts for each horizon, avoiding error accumulation and maintaining accuracy over long prediction ranges, as shown in Table 4.

Table 5 summarizes the computational efficiency of the models, highlighting the initialization/training time and inference time. Traditional machine learning models like Random Forest are computationally efficient but fail to capture complex spatio-temporal dependencies, limiting their predictive performance. On the other hand, advanced models like DCRNN and STGODE offer high accuracy but are computationally expensive. *TrafficDatorNet* balances these trade-offs, offering superior accuracy while maintaining reasonable training and inference times. The results obtained yield several key insights:

- *Predictive accuracy*. *TrafficDatorNet* consistently outperforms other models across all standard prediction horizons, from 3 up to 12 steps. Its ability to model complex spatio-temporal relationships makes it highly effective for urban traffic forecasting.

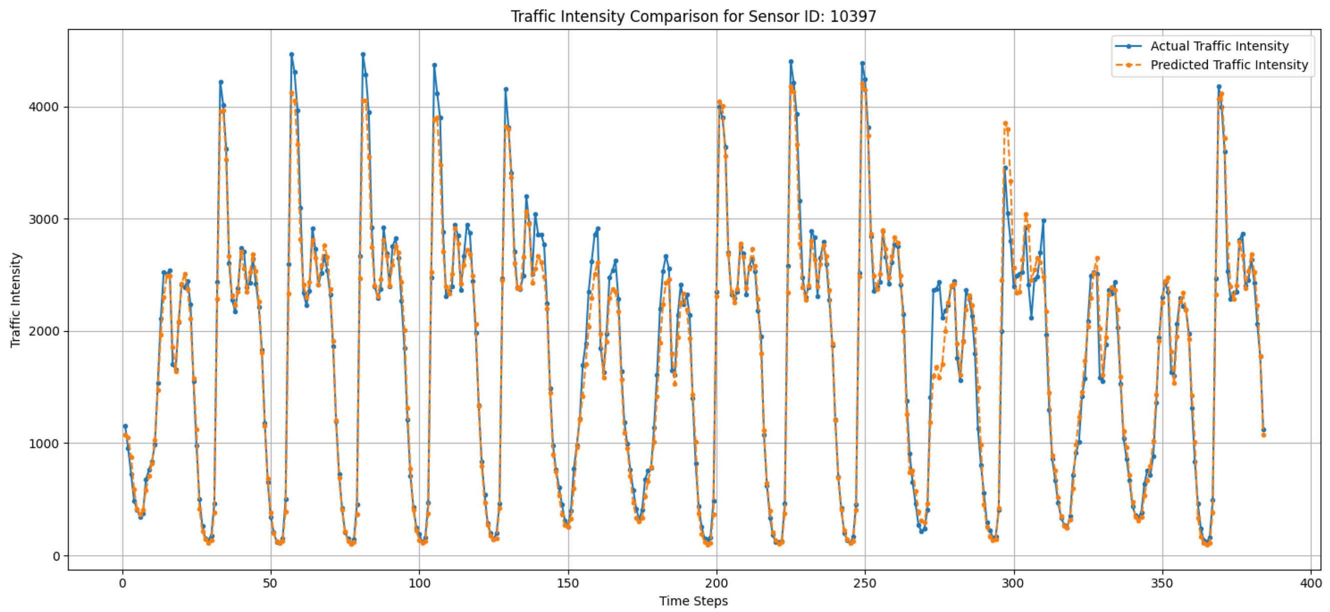

FIGURE 3. **Long-horizon forecast (384 steps; Sep 30–Oct 16, 2023) at sensor 10397 (Paseo de Extremadura, Madrid): real intensity (blue) vs. *TrafficDatorNet* prediction (orange).**

- *Computational efficiency and scalability*. By predicting features independently, *TrafficDatorNet* optimizes resource usage and maintains reasonable training and inference times. This makes it suitable for real-time applications, where speed and efficiency are critical. Compared with models based on recurrent neural networks, *TrafficDatorNet* achieves a better balance between computational cost and accuracy, offering a practical and scalable solution for operational implementations.
- *Extended horizon predictions*. *TrafficDatorNet's* performance over extended horizons, such as 192 and 384 steps, is notably stable, addressing a common limitation of traditional models. This makes it an ideal option for applications requiring long-term traffic forecasts, such as weekly urban planning and congestion management, as well as for short-term predictions.

In terms of scalability, the architecture contains a single component whose size grows with the number of sensors: the sensor-embedding table, with space complexity $\mathcal{O}(N_s \cdot d_e)$, where N_s denotes the number of sensors and d_e the number of embedding dimensions see Equation 2, whereas the remaining dense blocks remain constant. Consequently, both the number of parameters and the required memory grow linearly with the network size, while the inference cost, dominated by dense layers, is independent of the historical window length (there is no recurrence or self-attention over sequences). Empirically, on a CPU (Intel i7-1165G7, 16 GB RAM), *TrafficDatorNet* trains in 47.92 min and predicts in 0.42 ± 0.08 s per batch, whereas graph- and transformer-based references require $878–15,560$ min of training and 0.12–2.35 s for prediction (Table 5). Moreover, the performance remains stable for extended horizons (Table 4), confirming the model's practical scalability for operational deployments.

In summary, *TrafficDatorNet* strikes a unique balance between accuracy, efficiency, and scalability, making it a robust and practical solution for tackling the challenges of urban traffic forecasting in both short- and long-term scenarios.

D. EXAMPLE SCENARIO

To concretely illustrate the remarkable stability of *TrafficDatorNet* in long-range forecasting scenarios, we selected traffic sensor ID 10397 as a case study. This sensor is located at 393 "Paseo de Extremadura", in the southwestern area of Madrid, within the "Latina" district. This location holds strategic importance due to its proximity to the A-5 highway, one of the main access routes to the capital from the west. The surrounding area is characterized by intense urban activity, combining established residential neighborhoods with a diverse commercial fabric that includes small shops, supermarkets, cafés, and local services. Additionally, its closeness to major thoroughfares such as Illescas street and Boadilla road reinforces its role as a key transit point.

In terms of mobility and connectivity, this area is well served by public transportation, with several bus lines and nearby metro stations such as "Campamento" and "Empalme", facilitating movement for both residents and visitors. As such, this location is particularly relevant for analyzing urban dynamics, whether from an environmental or mobility perspective. The strategic location of the selected sensor allows for the collection of representative data reflecting the daily operations of a densely populated area along an urban corridor that links southwestern neighborhoods with the city center. This vibrant and accessible environment, characteristic of a major metropolitan area, is ideal for validating the model's effectiveness. It also demonstrates its practical utility in real-world conditions marked by high complexity and urban demand.

Within this context, Fig. 3 presents a comparison between the actual traffic intensity and the prediction generated by *TrafficDatorNet* at a 384-step horizon (i.e., predicting the traffic that will occur during the next 16 days) for the sensor with ID 10397, covering the period September 30 to October 16, 2023. As shown in Fig. 3, the prediction generated by *TrafficDatorNet* (orange line) for sensor ID 10397 closely follows the overall trends of the actual traffic data (blue line). The model's accuracy at this extended forecasting horizon for this specific sensor is supported by the following performance metrics: a coefficient of determination (R^2) of 0.9791, a Mean Absolute Error (MAE) of 108.07 vehicles/hour, a Root Mean Squared Error (RMSE) of 162.13 vehicles/hour, and a Mean Absolute Percentage Error (MAPE) of 8.11%. These results highlight the robustness and reliability of *TrafficDatorNet*, even in long-range predictive scenarios.

VI. CONCLUSION AND FUTURE DIRECTIONS

This work introduces *TrafficDatorNet*, a lightweight architecture that fuses sensor data, temporal signals, meteorology, and urban attributes using embeddings and dense layers to forecast traffic across short- and long-term horizons. On the *Madrid Traffic Dataset*, which offers broader coverage and richer context than standard benchmarking datasets, the model matches or surpasses sequential, graph-based, and traditional baselines while keeping training and inference costs low, with stable accuracy even at extended ranges. Moreover, we have released the dataset, code, and pipeline. Looking ahead, contextual signals may be enriched with exogenous factors (events, incidents, proximity features) and social-media cues to anticipate atypical demand, and forecasts could serve as demand generators in simulators (microsimulation or digital twins) to support what-if planning (e.g., road closures, disasters, pedestrianization, schedule shifts), enabling the assessment of their impact and the design of more effective and adaptive response strategies.

REFERENCES

[1] J. M. Bandeira, E. Macedo, P. Fernandes, M. Rodrigues, M. Andrade, and M. C. Coelho, "Potential pollutant emission effects of connected and automated vehicles in a mixed traffic flow context for different road types," *IEEE Open J. Intell. Transp. Syst.*, vol. 2, pp. 364–383, 2021.

[2] S. Ilarri, R. Trillo-Lado, and L. Marrodán, "Traffic and pollution modelling for air quality awareness: An experience in the city of Zaragoza," *SN Comput. Sci.*, vol. 3, no. 4, p. 281, 2022.

[3] A. Mystakidis, P. Koukaras, and C. Tjortjis, "Advances in traffic congestion prediction: An overview of emerging techniques and methods," *Smart Cities*, vol. 8, no. 1, p. 25, 2025.

[4] A. J. Huang and S. Agarwal, "Physics-informed deep learning for traffic state estimation: Illustrations with LWR and CTM models," *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 503–518, 2022.

[5] "A European AI on demand platform and ecosystem," AI4EU Consortium, 2019. Accessed: Oct. 6, 2025. [Online]. Available: https://cordis.europa.eu/project/id/825619

[6] "FATE: Fairness, accountability, transparency & ethics in AI," Microsoft Research. Accessed: Oct. 6, 2025. [Online]. Available: https://www.microsoft.com/en-us/research/group/fate/

[7] M. D. Wilkinson et al., "The FAIR guiding principles for scientific data management and stewardship," *Sci. Data*, vol. 3, pp. 1–9, Mar. 2016.

[8] I. Gómez and S. Ilarri, "TrafficDator: Traffic data analysis and prediction for better mobility," Accessed: Oct. 6, 2025. [Online]. Available: http://webdiis.unizar.es/ silarri/prot/TrafficDator/

[9] I. Gómez and S. Ilarri, "TrafficDator: Source code repository," 2025. [Online]. Available: https://github.com/TrafficDator/TrafficDator

[10] I. Gómez and S. Ilarri, "TrafficDatorNet: Code and baselines for traffic prediction with heterogeneous data (Madrid traffic dataset)," 2025. [Online]. Available: https://zenodo.org/records/10435154

[11] I. Gómez and S. Ilarri, "Enriched traffic datasets for Madrid," Version 2, Mendeley Data, 2025. Accessed: Oct. 6, 2025. [Online]. Available: https://data.mendeley.com/datasets/697ht4f65b/2

[12] S. A. Sayed, Y. Abdel-Hamid, and H. A. Hefny, "Artificial intelligence-based traffic flow prediction: A comprehensive review," *J. Elect. Syst. Inf. Technol.*, vol. 10, no. 1, p. 13, 2023.

[13] B. Gomes, J. Coelho, and H. Aidos, "A survey on traffic flow prediction and classification," *Intell. Syst. Appl.*, vol. 20, Nov. 2023, Art. no. 200268.

[14] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–16.

[15] Z. Fang, Q. Long, G. Song, and K. Xie, "Spatial-temporal graph ODE networks for traffic flow forecasting," in *Proc. 27th ACM SIGKDD Conf. Knowl. Disc. Data Min. (KDD)*, 2021, pp. 364–373.

[16] M. Méndez, M. G. Merayo, and M. Núñez, "Long-term traffic flow forecasting using a hybrid CNN-BiLSTM model," *Eng. Appl. Artif. Intell.*, vol. 121, May 2023, Art. no. 106041.

[17] W. Duan, X. He, Z. Zhou, L. Thiele, and H. Rao, "Localised adaptive spatial-temporal graph neural network," in *Proc. 29th ACM SIGKDD Conf. Knowl. Disc. Data Min. (KDD)*, 2023, pp. 448–458.

[18] W. Duan, S. Guo, Z. Zhou, W. Huang, H. Rao, and X. He, "Dynamic localisation of spatial-temporal graph neural network," in *Proc. 31st ACM SIGKDD Conf. Knowl. Disc. Data Min. (KDD)*, Aug. 2025, pp. 283–294.

[19] Y. Fang et al., "Efficient large-scale traffic forecasting with transformers: A spatial data management perspective," in *Proc. 31st ACM SIGKDD Conf. Knowl. Disc. Data Min. (KDD)*, 2025, pp. 307–317.

[20] A. Ali, Y. Zhu, and M. Zakarya, "Exploiting dynamic spatio-temporal graph convolutional neural networks for citywide traffic flows prediction," *Neural Netw.*, vol. 145, pp. 233–247, Jan. 2022.

[21] A. Ali, Y. Zhu, and M. Zakarya, "A data aggregation based approach to exploit dynamic spatio-temporal correlations for citywide crowd flows prediction in fog computing," *Multimedia Tools Appl.*, vol. 80, no. 20, pp. 31401–31433, 2021.

[22] N. Awan et al., "Modeling dynamic spatio-temporal correlations for urban traffic flows prediction," *IEEE Access*, vol. 9, pp. 26502–26511, 2021.

[23] A. Ali, Y. Zhu, and M. Zakarya, "Exploiting dynamic spatio-temporal correlations for citywide traffic flow prediction using attention based neural networks," *Inf. Sci.*, vol. 577, pp. 852–870, Oct. 2021.

[24] M. Zakarya, L. Gillam, A. A. Khan, O. Rana, and R. Buyya, "APMOVE: A service migration technique for connected and autonomous vehicles," *IEEE Internet Things J.*, vol. 11, no. 17, pp. 28721–28733, Sep. 2024.

[25] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. 30th Int. Conf. Mach. Learn. (ICML 2013)*, 2013, pp. 1–9.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[27] "Keras," Accessed: Oct. 13, 2025. [Online]. Available: https://keras.io/

[28] J. Chen, 2024, "Traffic flow," Dataset, IEEE Dataport. Accessed: Oct. 6, 2025. [Online]. Available: https://ieee- dataport.org/documents/traffic-flow

[29] X. Liu, H. Wang, Y. Zhang, and J. Chen, “LargeST: A benchmark dataset for large-scale spatiotemporal traffic prediction,” in *Proc. 37th Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2023, pp. 1–18.
[30] I. Gómez and S. Ilarri, “Enriched traffic datasets for the city of Madrid: Integrating data from traffic sensors, the road infrastructure, calendar data and weather data,” *Data Brief*, vol. 57, Dec. 2024, Art. no. 110878.
[31] “Madrid City council open data portal,” Accessed: Oct. 6, 2025. [Online]. Available: https://datos.madrid.es/portal/site/egob/menuitem.9e1e2f6404558187cf35cf3584f1a5a0/?vgnextoid=374512b9ace9f310VgnVCM100000171f5a0aRCRD&vgnextchannel=374512b9ace9f310VgnVCM100000171f5a0aRCRD&vgnextfmt=default
[32] “Meteorological data for Madrid—Hourly data since 2019,” Accessed: Oct. 6, 2025. [Online]. Available: https://datos.madrid.es/portal/site/egob/menuitem.c05c1f754a33a9fbe4b2e4b284f1a5a0/?vgnextoid=fa8357cec5efa610VgnVCM1000001d4a900aRCRD&vgnextchannel=374512b9ace9f310VgnVCM100000171f5a0aRCRD&vgnextfmt=default
[33] “Openstreetmap,” OpenStreetMap contributors. Accessed: Oct. 6, 2025. [Online]. Available: https://www.openstreetmap.org
[34] G. Boeing, “OSMNX: New methods for acquiring, constructing, analyzing, and visualizing complex street networks,” *Comput. Environ. Urban Syst.*, vol. 65, pp. 126–139, Jul. 2017.
[35] J. L. Bentley, “Multidimensional binary search trees used for associative searching,” *Commun. ACM*, vol. 18, no. 9, pp. 509–517, 1975.
[36] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.
[37] R. W. Sinnott, “Virtues of the haversine,” *Sky Telescope*, vol. 68, no. 2, pp. 158–159, 1984.
[38] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.
[39] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[40] J. L. Elman, “Finding structure in time,” *Cogn. Sci.*, vol. 14, no. 2, pp. 179–211, 1990.
[41] J. Smith, E. Johnson, W. Zhang, and M. García, “A survey of traffic flow prediction methods based on long short-term memory networks,” *IEEE Intell. Transp. Syst. Mag.*, vol. 16, no. 2, pp. 45–58, Sep./Oct. 2024.
[42] T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, “Neural ordinary differential equations,” in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 31, 2018, pp. 1–13.
[43] L. Breiman, “Random forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[44] S. Huang, Z. Niu, and J. Huang, “Road traffic congestion prediction based on random forest and DBSCAN combined model,” in *Proc. 5th Int. Conf. Smart Grid Elect. Autom. (ICSGEA)*, 2020, pp. 323–326.
[45] J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, 2001.
[46] S. Yang, J. Wu, Y. Du, Y. He, and X. Chen, “Ensemble learning for short-term traffic prediction based on gradient boosting machine,” *J. Sens.*, vol. 2017, pp. 1–15, May 2017.
[47] T. Cover and P. Hart, “Nearest neighbor pattern classification,” *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967.
[48] L. Cai, Y. Yu, S. Zhang, Y. Song, Z. Xiong, and T. Zhou, “A sample-rebalanced outlier-rejected k-nearest neighbor regression model for short-term traffic flow forecasting,” *IEEE Access*, vol. 8, pp. 22686–22696, 2020.
[49] “TensorFlow,” Accessed: Oct. 13, 2025. [Online]. Available: https://www.tensorflow.org/
[50] PyTorch Team, “PyTorch,” Accessed: Oct. 13, 2025. [Online]. Available: https://pytorch.org/
[51] Y. Gal and Z. Ghahramani, “Dropout as a Bayesian approximation: Representing model uncertainty in deep learning,” in *Proc. 33rd Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 1050–1059. [Online]. Available: https://proceedings.mlr.press/v48/gal16.html
[52] A. Wang, Y. Ye, X. Song, S. Zhang, and J. J. Q. Yu, “Traffic prediction with missing data: A multi-task learning approach,” *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 4, pp. 4189–4202, Apr. 2023.
[53] T. Mallick, J. MacFarlane, and P. Balaprakash, “Uncertainty quantification for traffic forecasting using deep-ensemble-based spatiotemporal graph neural networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 8, pp. 9141–9152, Aug. 2024.
[54] N. H. An and D. T. Anh, “Comparison of strategies for multi-step-ahead prediction of time series using neural network,” in *Proc. Int. Conf. Adv. Comput. Appl. (ACOMP)*, 2015, pp. 142–149.
[55] D. Sahoo, N. Sood, U. Rani, G. Abraham, V. Dutt, and A. Dileep, “Comparative analysis of multi-step time-series forecasting for network load dataset,” in *Proc. 11th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, 2020, pp. 1–7.
[56] D. H. Nguyen, T. A. Duong, and R. Jiang, “Strategies of multi-step-ahead forecasting for chaotic time series using autoencoder and LSTM neural networks: A comparative study,” in *Proc. Int. Conf. Image Process. Mach. Vis. (IPMV)*, 2023, pp. 45–50.
[57] E. Doğan, “Performance analysis of LSTM model with multi-step ahead strategies for a short-term traffic flow prediction,” *Sci. J. Silesian Univ. Technol. Series Transport*, vol. 111, pp. 15–31, Jun. 2021.
[58] B. Fernandes, F. Silva, H. Alaiz-Moreton, P. Novais, J. Neves, and C. Analide, “Long short-term memory networks for traffic flow forecasting: Exploring input variables, time frames and multi-step approaches,” *Informatica*, vol. 31, no. 4, pp. 723–749, 2020.

IVÁN GÓMEZ graduated in electronic and automatic engineering from the Polytechnic University School of Teruel, Spain. He received the master’s degree in big data and data analysis from UNIR, Spain. He is currently pursuing the Ph.D. degree in artificial intelligence and data science in the context of the NEAT-AMBIENCE Project, focusing on advanced traffic analysis and prediction to enhance urban mobility. In addition to his academic pursuits, his professional career extends to the private sector, where he addresses data engineering challenges, data analysis, and data science in dynamic business environments. He was awarded the best academic record and the best Final Degree Project from the Polytechnic University School of Teruel, which focused on fall risk estimation using accelerometers and machine learning techniques.

SERGIO ILARRI is a Full Professor of Computer Languages and Systems Engineering at the Universidad de Zaragoza. He is currently the Coordinator of the Computer Science Engineering degree at the School of Engineering and Architecture, Universidad de Zaragoza. He has also developed strong international collaborations, through research stays (e.g., the Mobile Computing Laboratory, University of Illinois at Chicago, Chicago, and also with the University of Valenciennes, France) and other actions. He has been leading the COSMOS Research Group (funded by the Government of Aragon) since the recognition period 2017–2019 (group references: T35_17D, T64_20R, and currently T64_23R), and leads the Research Project NEAT-AMBIENCE, funded by the Spanish State Research Agency (Agencia Estatal de Investigación). He has authored publications in relevant journals and conferences. His main research area is data management. His research interests include mobile computing, vehicular networks, context-aware recommender systems, data streams, information systems, and data-driven decision making.

Dr. Ilarri has participated in the organization and program committees of conferences, and as a Guest Editor of special issues of journals, such as *Transportation Research Part C-Emerging Technologies*, *IEEE Internet Computing*, *IEEE Multimedia*, the *Journal of Systems and Software*, *Distributed and Parallel Databases*, and *Personal and Ubiquitous Computing*.