

Article

Prediction and Modeling of Traffic Status at Road Intersection Using Deep-Learning Models

Chaymae Chouiekh ^{1,*}, Ali Yahyaouy ¹, My Abdelouahed Sabri ¹, Hicham Karmouni ²,
Mudasir Ahmad Wani ³, Kashish Ara Shakil ⁴ and Basma Abd El-Rahiem ⁵

¹ Faculty of Sciences Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, Fez 30000, Morocco; ali.yahyaouy@usmba.ac.ma (A.Y.); abdelouahed.sabri@usmba.ac.ma (M.A.S.)

² Higher School of Technology, Hassan II University of Casablanca, Casablanca 20470, Morocco; hicham.karmouni@usmba.ac.ma

³ College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 13318, Saudi Arabia; mawani@imamu.edu.sa

⁴ Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Riyadh 11671, Saudi Arabia; kashakil@pnu.edu.sa

⁵ Department of Mathematics and Computer Science, Faculty of Science, Menoufia University, Shebin El-Koom 32511, Egypt; basma.rahiem@science.menofia.edu.eg

* Correspondence: chaymae.chouiekh@usmba.ac.ma

Abstract

Efficient traffic-state prediction at urban intersections is a critical component of intelligent transportation systems (ITS), as traffic conditions are influenced by dynamic factors such as traffic demand variability, infrastructure constraints, and operational traffic-control policies. This study proposes a deep-learning-based approach for short-term traffic-state classification using real-world traffic data collected during 2022 at the Alésia intersection in Paris. The objective is to classify traffic conditions into five operational states: *Unknown*, *Flowing*, *Pre-saturated*, *Saturated*, and *Blocked*. To investigate the impact of temporal modeling on traffic-state recognition, four deep learning architectures were evaluated under identical experimental conditions: Artificial Neural Networks (ANN), Simple Recurrent Neural Networks (RNN), Long Short-Term Memory networks (LSTM), and Gated Recurrent Units (GRU). Considering the highly imbalanced nature of the dataset, model performance was assessed using complementary metrics including Accuracy, Precision, Recall, F1-score, Macro-F1 score, and Balanced Accuracy. Experimental results demonstrate that recurrent architectures substantially outperform the ANN baseline, highlighting the importance of temporal dependencies in traffic-state classification. While the conventional RNN achieves high overall accuracy, its performance on minority traffic states remains limited. Among the evaluated models, the LSTM achieves the highest Balanced Accuracy (70.91%), indicating superior recognition of underrepresented traffic conditions. The GRU attains the highest overall F1-score (0.9256) and Macro-F1 score (0.497), while maintaining competitive classification accuracy (91.01%), providing the most favorable trade-off between global predictive performance and balanced class-wise recognition. The analysis of learning curves, classification reports, and confusion matrices further confirms the effectiveness of gated recurrent architectures for handling highly imbalanced multiclass traffic-state classification problems. These findings provide practical insights for the deployment of intelligent traffic-monitoring systems capable of supporting real-time traffic management and decision-making in urban environments.

Keywords: intelligent transportation systems; traffic forecasting; time series analysis; recurrent neural networks; smart mobility



Academic Editors: Yanyan Xu, Yu Shen and Chunxiao Li

Received: 2 June 2026

Revised: 16 June 2026

Accepted: 23 June 2026

Published: 8 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Urban traffic management remains a critical issue in modern large cities, where congestion, delays, polluting emissions, and accidents are the direct consequences of inefficient planning. Road intersections, in particular, play a central role in the regulation of the flow of vehicles and pedestrians. Their operational complexity, reinforced by time variability, human behavior, and environmental conditions, makes them critical areas for decision-makers. The need for intelligent and adaptive solutions to predict, understand, and manage traffic behavior in these hotspots has never been more pressing.

In this context, artificial intelligence(AI) approaches, including deep learning models, offer promising potential to address the challenges of traffic prediction. Many studies have already explored the use of ANN to predict environmental variables such as air pollution (PM2.5, ozone) [1–3], or to anticipate temporal developments in health areas such as the spread of COVID-19 [4]. These studies demonstrate the ability of these models to capture non-linear trends in complex time series [5,6]. However, their practical application in the field of urban traffic, and particularly at intersections, remains under-exploited [7,8].

Despite the progress made, several limitations remain in the existing literature. On the one hand, the performance of models is often presented only in the form of quantitative measures (precision, recall, F1 score), without explanation on how the results can be used concretely by decision-makers [9]. On the other hand, the data used are sometimes simulated or collected in too generic contexts, which limits their application to real, dynamic, and heterogeneous urban environments [10].

In order to fill these gaps, our study proposes an intelligent approach for short-term traffic forecasting at a real urban intersection, using different deep learning models. We use real data collected in 2022 at the “Alésia” intersection of Paris, an area characterized by dense road and pedestrian activity. The objective is not only to obtain accurate forecasts, but also to assess how these forecasts can be interpreted and used in an intelligent traffic management approach.

The main objective of this article is to accurately predict short-term road traffic conditions at a real urban intersection using locally collected traffic data. The contributions of this research are summarized as follows:

- (1) The exploitation of a real-world traffic dataset collected during 2022 at the Alésia intersection in Paris, including occupancy rate, traffic flow, and temporal information;
- (2) The implementation and comparative evaluation of four deep learning architectures—ANN, RNN, LSTM, and GRU—for the multiclass classification of traffic states (*Unknown*, *Flowing*, *Pre-saturated*, *Saturated*, and *Blocked*);
- (3) A comprehensive performance assessment based on Accuracy, Precision, Recall, F1-score, Macro-F1 score, Balanced Accuracy, classification reports, learning curves, and confusion matrices, enabling a detailed analysis of both majority and minority traffic-state recognition;
- (4) The identification of the strengths and limitations of each architecture under highly imbalanced traffic conditions, highlighting the superior minority-class recognition capability of the LSTM model and the favorable overall performance trade-off achieved by the GRU model.

This study is deliberately focused on the predictive performance of deep learning models and provides a solid foundation for future integration into intelligent traffic monitoring and traffic management systems.

The rest of the article is organised as follows: Section 2 presents the work related to traffic prediction and the models used; Section 3 describes the data; Section 4 the proposed methodology; Section 5 presents the results and their interpretation; Finally, Section 6 concludes the study and opens up future perspectives.

The novelty of this study does not lie in proposing a new deep-learning architecture, but rather in addressing traffic-state prediction as a highly imbalanced multi-class classification problem using real-world operational traffic data collected from a major urban intersection. Unlike most previous studies that focus on traffic-flow forecasting as a regression task, this work investigates the classification of five operational traffic states (Unknown, Flowing, Pre-saturated, Saturated, and Blocked), which are directly relevant for intelligent traffic management and real-time decision-making.

A major challenge of the considered dataset is its severe class imbalance, where the dataset exhibits a highly imbalanced distribution, where the Flowing class represents the vast majority of observations, while the Blocked, Pre-saturated, Saturated, and Unknown classes are considerably underrepresented. Such distributions are commonly encountered in real traffic-monitoring systems and may lead to biased models if not properly evaluated.

Therefore, beyond conventional accuracy assessment, this study emphasizes robust evaluation metrics including Precision, Recall, F1-score, Macro-F1, and Balanced Accuracy to provide a fair assessment of model performance across both majority and minority traffic states. Furthermore, a comprehensive comparison of ANN, RNN, LSTM, and GRU architectures is conducted under identical experimental conditions to analyze their capability to learn temporal dependencies while maintaining reliable detection of rare traffic states. The findings provide practical guidelines for selecting deep-learning architectures in ITS operating under realistic and highly imbalanced traffic conditions.

2. Related Works

Traffic management is an important topic with major social and economic benefits, especially at metropolitan crossings. To improve traffic management and forecasting, numerous researchers have put up a number of strategies, most notably the use of intelligent transport systems (ITS) [11]. These systems, which offer increased comfort, safety, and efficiency, are crucial to the transition of conventional transportation networks [12–14].

Several recent studies have focused on urban congestion management and traffic forecasting. Ref. [15] proposed a deep learning model to predict congestion levels in real time, which can guide proactive urban traffic planning. Ref. [16] explored different smart approaches to reduce congestion in sustainable cities, including through the integration of intelligent transport systems (ITS). In addition, Ref. [17] demonstrated the importance of coordination between land use planning and traffic modelling for effective flow management. These approaches complement our algorithmic proposal by adding an essential predictive dimension to the optimization of journeys.

Traffic state prediction has emerged as a key area of study within the ITS. Its foundation is the gathering, processing, and analysis of extremely non-linear time data, the variability of which is influenced by a wide range of variables (e.g., events, peak hours, weather conditions, etc.). Traffic forecasting is a key component of Traffic Status Indicators (TSI), which are crucial tools for decision support.

Originally, attempts to forecast traffic began in the 1970s and used mostly standard statistical tools such as [18–20]. But the emergence of AI significantly changed the field of methodology, and by recent advances in machine learning and deep learning. Several approaches have been explored in the literature to model traffic behaviors, each with specific strengths and limitations.

In Ref. [21], a systematic review of microscopic traffic models based on machine learning is presented. The authors identify a variety of supervised and unsupervised methods used to simulate individual vehicle behaviors and track change models, identifying critical gaps in research and proposing future directions, focusing on vehicle behaviour and traffic patterns rather than driver physiological behaviours.

At a more operational level, Ref. [22] introduces a deep learning transfer approach for the classification of path movements at intersections from connected vehicle mass data. The capacity to anticipate vehicle movements at crossings becomes crucial as linked vehicles use cutting-edge IT technologies to revolutionise road safety. To increase the safety of drivers and other vulnerable road users, the project attempts to provide an efficient path movement categorisation tool that enhances predictions of connected vehicle (CV) movements at crossings. With an accuracy of 99.73%, the method demonstrated how convolutional network designs may be used to comprehend spatial traffic patterns.

Furthermore, by modelling demand and capacity by intervals, Ref. [23] suggests a novel approach to tackling unskilled transport (UTP) problems under uncertainty. This study presents the UTP with demand defined by interval and supplier capabilities (UT-PIDS), where these values fluctuate due to economic conditions. Although focused on transport planning, this innovative approach provides useful insights for managing uncertainties in flow prediction.

Additionally, recent evaluations like [24] offer a thorough study of the various machine learning approaches used in traffic prediction as well as a full overview of modern methodologies. The study evaluates the prediction accuracy and robustness of multiple models, such as SVM, Random Forest, XGBoost, and neural networks, across a range of data sets. This analysis demonstrates the importance of choosing the algorithm according to the nature of the data and the deployment objectives in a real environment. Along a similar lines, Ref. [25] presents an in-depth comparative study on various traffic modelling techniques, particularly their use in Intelligent Transport Systems (ITS) to predict and simulate traffic behaviour in different road environments. In addition, it [26] examines the behaviour of drivers when crossing red lights, identifying existing predictive models and associated technological countermeasures. Although focused on a specific behavioural aspect, this analysis highlights the importance of integrating precise predictions to improve road safety.

Although substantial progress has been achieved in traffic prediction, important challenges remain, particularly the requirement for large-scale high-quality datasets and the computational complexity of advanced deep-learning architectures. The results reported in this study suggest that memory-based recurrent models, namely LSTM and GRU, offer a more balanced classification performance than ANN and conventional RNN approaches. Their ability to capture temporal dependencies contributes not only to high predictive accuracy but also to improved recognition of underrepresented traffic states, which is a critical requirement in real-world urban traffic monitoring applications.

In the same dynamic of continuous improvement, several recent works have proposed targeted algorithmic optimizations. In particular, some research has highlighted the use of the GRU model, coupled with fine-tuning techniques for hyperparameters, allowing more accurate traffic forecasts and a reduction of errors in real time [27]. Other studies have explored machine learning to model gas emissions and fuel consumption from actual driving data, from an environmental monitoring perspective, and to encourage more environmentally friendly behaviors [28]. These complementary approaches demonstrate the adaptability of deep learning models to various intelligent transportation issues, extending beyond simple traffic modeling.

Table 1 summarizes and compares these major contributions, highlighting their methodologies, strengths, and limitations. Notably, our study differs by utilizing real data collected from the busy Alésia intersection in Paris, and by evaluating and comparing four models—ANN, RNN, GRU, and LSTM—in terms of their ability to classify traffic states into five meaningful categories. This not only enhances predictive accuracy but also improves interpretability and potential integration into intelligent traffic control systems.

Table 1. Comparative synthesis of recent traffic prediction and modeling studies.

Study	Methodology	Strengths	Limitations
Rowan et al. [21]	Review of ML-based car-following and lane-changing models in microscopic traffic flow simulation	Identifies critical research gaps, model transferability, and driver heterogeneity	Lacks quantitative comparisons and practical implementation guidance
Komol et al. [22]	Deep transfer learning applied to connected vehicle trajectory data for intersection classification	High classification accuracy (99.73%), effective for connected vehicle environments	Requires large-scale CV data, limited generalizability to different intersections
Silmi Juman et al. [23]	Algorithm for uncapacitated transport under uncertainty using interval values	Effectively models uncertainty; useful for strategic transport planning	Not designed for real-time traffic prediction, limited to planning scenarios
Alghamdi et al. [25]	Comparative review of modeling techniques for traffic simulation and prediction	Offers a broad view of ITS modeling, compares multiple methodologies	Focuses more on classification than specific algorithm performance in real cases
Komol et al. [26]	Review of red-light violation prediction models and prevention technologies	Emphasizes safety, integrates behavioral predictions and countermeasures	Focuses on red-light cases; not comprehensive for general traffic prediction
Parashuram & Vijayalakshmi [24]	Performance analysis of ML models (SVM, RF, XGBoost, NN) across datasets	Quantitative comparison of models; supports algorithm selection for deployment	Less emphasis on intersection-level behavior or signal timing applications
Hussain et al. [27]	GRU model with hyperparameter fine-tuning for real-time traffic forecasting	Improves prediction accuracy and reduces real-time forecasting errors	Requires careful tuning and high-quality data for optimal performance
Shahariar et al. [28]	ML-based modeling of gas emissions and fuel consumption using real driving data	Promotes eco-driving and environmental monitoring through data-driven insights	Focuses on environmental impact; less oriented toward traffic flow optimization

Although substantial progress has been achieved in ITS, previous studies have explored a wide range of approaches, including conventional machine-learning methods, deep-learning architectures, traffic-flow forecasting models, driving-risk assessment systems, and intelligent traffic-control strategies. Recent contributions have further extended these applications to real-time driving-risk prediction using vehicle trajectory data and traffic-control optimization in connected vehicle environments under cyber-attack scenarios [29,30]. These studies demonstrate the increasing importance of data-driven and AI-based solutions for transportation management.

Nevertheless, most existing works focus on traffic-flow forecasting, risk assessment, or traffic-control optimization, whereas comparatively fewer studies investigate multiclass traffic-state classification at the intersection level using real-world operational datasets. Moreover, limited attention has been given to the challenges posed by highly imbalanced traffic-state distributions, which may significantly affect model training and evaluation. Therefore, this study aims to evaluate and compare the performance of ANN, RNN, LSTM, and GRU models for multiclass traffic-state classification using real-world traffic data collected from the Alésia intersection in Paris. The study also investigates the effectiveness

of cost-sensitive learning through class weighting and employs Balanced Accuracy and Macro-F1 metrics to provide a more reliable evaluation under imbalanced-data conditions.

Recent advances in traffic prediction have increasingly focused on spatio-temporal deep learning architectures capable of jointly modeling spatial dependencies and temporal dynamics. In particular, Spatio-Temporal Graph Neural Networks (STGNNs) have demonstrated remarkable performance by exploiting the topological structure of transportation networks. For example, Pan et al. (2025) proposed a physics-guided stepwise framework based on STGNN algorithms for urban intersection traffic-flow prediction, showing improved forecasting accuracy through the integration of traffic-flow theory and graph-based learning [31]. Similarly, Transformer-based architectures have recently attracted significant attention due to their ability to capture long-range temporal dependencies. Liu and Wang (2025) introduced an improved Transformer-based traffic prediction model that achieved competitive performance in complex traffic forecasting scenarios [32]. These recent developments highlight the growing importance of graph-based and attention-based learning frameworks for ITS. However, such architectures typically require larger datasets, higher computational resources, and network-level spatial information. Therefore, the present study focuses on evaluating recurrent deep-learning architectures (RNN, LSTM, and GRU) under identical experimental conditions using real-world intersection-level traffic data. Future work will investigate the integration of STGNN and Transformer-based models to further extend the comparative analysis.

3. Data Collection and Preprocessing

In the context of this study, effective dataset preprocessing is crucial to ensure the quality and reliability of the subsequent analyses. This section focuses on the various steps taken to prepare the raw dataset for further investigation [33]. By employing appropriate techniques and methodologies, we aim to address data inconsistencies, outliers, and missing values, while also refining the dataset for accurate analysis. Before delving into the preprocessing procedures, it is essential to provide a comprehensive overview of the dataset itself [34]. The subsequent subsection, “Dataset Collection,” presents an in-depth exploration of the data’s characteristics, including its sources, dimensions, and initial features. Understanding the characteristics, structure, and content of the dataset is essential as we embark on various preprocessing steps to ready it for analysis.

3.1. Dataset Collection

The Directorate of Roads and Travel-Department of Travel, in conjunction with the Central Post of Lutetia Operations, focuses on various objectives related to traffic regulation, public transportation, user education (via the Sytadin website), and research. Their primary attention is directed towards the main arteries within the City of Paris, where they have installed stations for vehicle counting and evaluating occupancy rates. There are two types of stations on the network: one type that solely measures occupancy rates and another that measures both occupancy rates and vehicle counts. The rate measurement stations are particularly valuable as they provide accurate and detailed information about traffic conditions. The less frequent flow stations are typically located near busy intersections, where traffic flow tends to be maintained between two significant crossing points. For this study, traffic data were collected from permanent sensors installed at the Alésia urban intersection in Paris during the year 2022. The characteristics of the dataset are summarized in Table 2. The original dataset consisted of 144,264 traffic records containing traffic status information, Hourly Debit measurements, occupancy rates, and timestamp information collected over a 12-month period.

The traffic dataset contains several attributes describing the operational conditions of the Alésia intersection. These attributes include the Arc Identifier, Upstream Node Identifier, Downstream Node Identifier, Count_date_time, Hourly Debit, Occupancy Rate, Arc Status, and the corresponding Traffic State label. The Arc and Node identifiers provide spatial information about the monitored road segment, while Count_date_time records the timestamp associated with each traffic observation, enabling the analysis of temporal traffic dynamics.

Table 2. Traffic Data Description.

Data	Description
Arc identifier	Unique identifier of the reference arc object.
Label	Arc description.
Count_date_time	Timestamp (end of development period) in ISO format: YYYY-MM-DD HH:MM:SS.
Hourly Debit	The number of vehicles that have passed the counting point during a fixed one-hour interval.
Occupancy rate	Vehicle time on loop as a percentage of a fixed time interval (one hour for data provided).
Traffic status	Traffic status qualification: 0—Unknown, 1—Smooth/Flowing, 2—Pre-saturated, 3—Saturated, 4—Blocked
Upstream node identifier	Unique identifier of the upstream node of the arc.
Upstream node description	Description of the upstream node of the arc.
Downstream node identifier	Arc downstream node identifier.
Downstream node description	Description of the downstream node.
Arc status	Status at arc circulation: 0—Unknown, 1—Open, 2—Closed/Barred, 3—Invalid
Date start dispo data	Data availability start time for the object (axis opening or first data availability).
Date end Dispo data	End Data Availability Timestamp for the object (axis closure or last update of data).
Geo point 2d	Length of object (meters).
Geo shape	Geometry of the object.

Among the collected variables, Hourly Debit and Occupancy Rate constitute the primary traffic-related indicators used to characterize road conditions. Hourly Debit represents the number of vehicles detected during a one-hour interval, whereas Occupancy Rate measures the percentage of time that vehicles occupy the detection loop within the same period. The occupancy rate is particularly important because similar traffic volumes may correspond to different traffic conditions depending on the level of road occupancy. For example, a flow of 100 vehicles per hour may indicate free-flow traffic during nighttime periods but severe congestion during peak traffic hours.

Furthermore, the Count_date_time attribute provides valuable temporal information that can be exploited to capture daily, weekly, and seasonal traffic patterns. Consequently, temporal features such as Hour, DayOfWeek, and Month were extracted during the pre-processing stage to enrich the representation of traffic behavior. By combining spatial, temporal, and traffic-related information, each observation can be assigned to one of five traffic-state categories: Unknown, Flowing, Pre-saturated, Saturated, and Blocked.

Table 3 presents an example of a traffic record collected from the Alésia intersection. Each observation contains temporal information together with the corresponding traffic measurements and traffic-state label used in this study.

Table 3. Example of traffic data from the used dataset.

Field	Value 1	Value 2
Identifiant arc	4959	4959
Label	Alesia	Alesia
Date and Time of Counting	03-03-2022 08:00:00 +01:00	03-03-2022 07:00:00 +01:00
HourlyFlow	107	62
Occupancy Rate	4.56833	1.83833
Traffic State	Flowing	Flowing
Upstream Node Identifier	399	399
Upstream Node Label	Av Rene Coty-Alesia	Av Rene Coty-Alesia
Downstream Node Identifier	2626	2626
Downstream Node Label	Alesia-Sante-Glaciere-Tolbiac	Alesia-Sante-Glaciere-Tolbiac
Arc State	Invalid	Invalid
Count_date_time	01-01-2005 00:00:00	01-01-2005 00:00:00
geo point 2d	48.82691273165781, 2.3383679454922497	48.82691273165781, 2.3383679454922497
geo shape	{"coordinates": [[2.3351115965, 48.8273981509], [2.3416242945, 48.8264273124]], "type": "LineString"}	{"coordinates": [[2.3351115965, 48.8273981509], [2.3416242945, 48.8264273124]], "type": "LineString"}

3.2. Dataset Preprocessing

To ensure data quality and improve the predictive performance of the proposed models, several preprocessing steps were applied to the raw traffic dataset. These steps included data cleaning, feature engineering, and class imbalance handling. The following subsections describe each preprocessing stage in detail.

3.2.1. Data Cleaning

The original dataset contained 144,264 traffic records collected from the Alésia intersection during 2022. To ensure data quality and consistency, records containing missing values, incomplete measurements, duplicate entries, and invalid observations were removed during the preprocessing stage. In particular, occupancy-rate values exceeding 100% were considered inconsistent with the official dataset definition, according to which the occupancy rate represents the percentage of time that vehicles occupy the detection loop during a one-hour interval. Consequently, 37 observations with occupancy rates greater than 100% (approximately 0.12% of the cleaned dataset) were discarded. After the complete cleaning process, a total of 30,498 valid observations were retained for subsequent analysis, model training, and evaluation.

After data cleaning, a statistical analysis was conducted to better understand the distribution of the key numerical variables. Table 4 summarizes the descriptive statistics of the Hourly Debit and Occupancy Rate variables in the cleaned dataset.

The descriptive statistics reveal substantial variability in traffic conditions. Hourly traffic volumes range from 0 to 1488 vehicles per hour, while occupancy rates vary between 0% and 95.35%. The median occupancy rate of 3.10% and the third quartile value of 6.19% indicate that most observations correspond to uncongested traffic conditions, whereas a limited number of observations represent highly congested situations. This variability highlights the complexity of urban traffic dynamics and motivates the use of machine-learning and deep-learning models capable of accurately discriminating between multiple traffic states.

Table 4. Descriptive Statistics of the Main Variables After Data Cleaning.

Measure	Hourly Debit	Occupancy Rate
Count	30,498	30,498
Mean	169.43	4.86
Standard Deviation	111.82	7.76
Min	0.00	0.00
25% Percentile	74.00	1.20
50% Percentile (Median)	157.00	3.10
75% Percentile	247.00	6.19
Max	1488.00	95.35

3.2.2. Feature Engineering

To enhance the representation of temporal traffic patterns, additional features were extracted from the original Count_date_time attribute. Specifically, four temporal variables were generated: Hour, DayOfWeek, Weekend, and Month. These features enable the learning models to capture hourly, weekly, and seasonal variations in traffic conditions.

Figures 1 and 2 illustrate the evolution of the average Hourly Debit and Occupancy Rate according to the day of the week and the month of the year, respectively. The observed variations indicate that traffic conditions exhibit clear temporal patterns at both weekly and monthly levels. Such recurring trends suggest that temporal information plays an important role in traffic-state characterization and prediction.

Consequently, the extracted temporal features were incorporated into the input dataset to provide additional contextual information for the deep learning models. These engineered attributes allow the models to better learn temporal dependencies and improve their ability to distinguish between different traffic states.

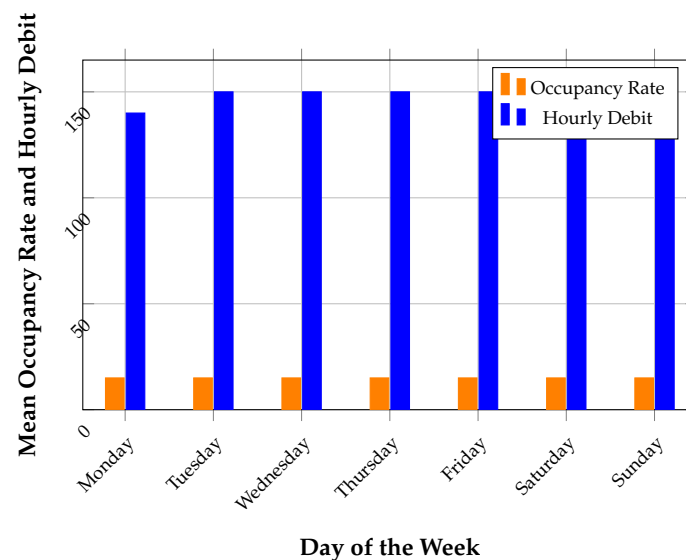


Figure 1. Evolution of the Occupancy Rate and Average Hourly Debit according to the days of the week.

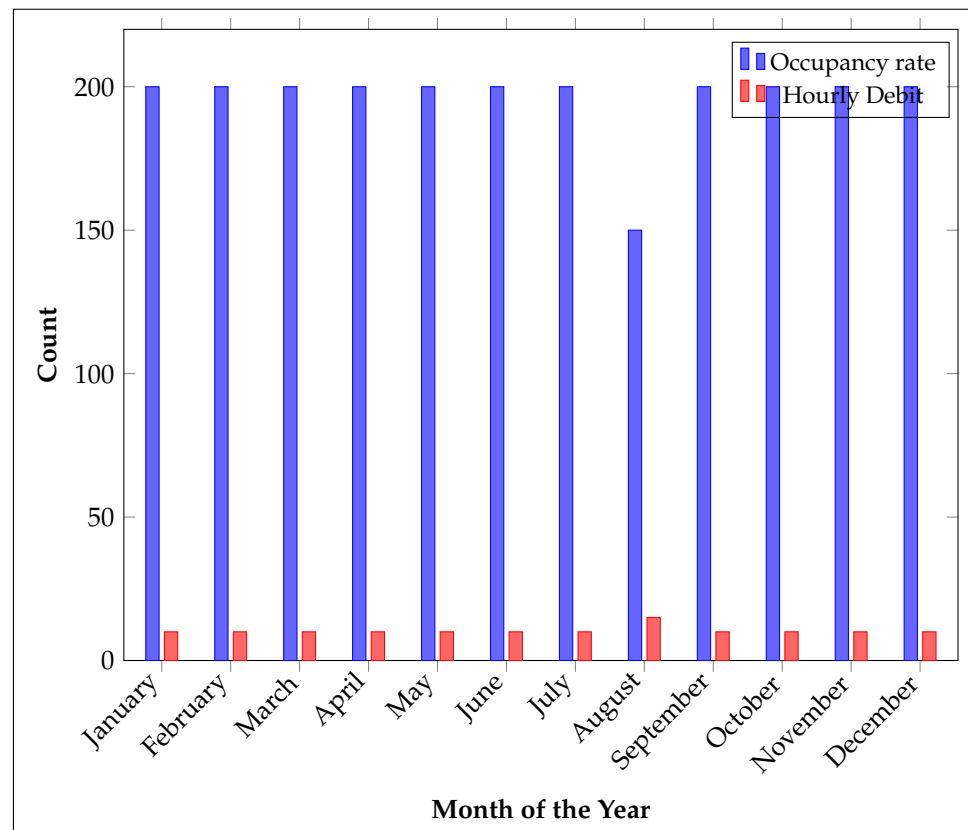


Figure 2. Occupancy Rate and Average Hourly Debit Grouped by Month of the Year.

3.2.3. Class Imbalance Handling

Figure 3 illustrates the distribution of traffic-state classes in the cleaned dataset. As can be observed, the dataset exhibits a highly imbalanced class distribution. The *Unknown* class represents the majority of observations, followed by the *Flowing* class, whereas the *Blocked*, *Pre-saturated*, and *Saturated* classes contain considerably fewer samples.

Such class imbalance may bias the learning process toward the majority classes and reduce the model's ability to correctly identify minority traffic states, which are often the most critical for traffic-management applications. To address this issue, different imbalance-handling strategies were investigated.

Initially, the Synthetic Minority Over-sampling Technique Synthetic Minority Over-sampling Technique (SMOTE) was applied to the training set in order to generate synthetic samples for the minority classes. However, experimental results showed that SMOTE did not improve the overall predictive performance and led to poorer generalization on the validation and testing datasets.

Consequently, a cost-sensitive learning strategy based on class weights was adopted. During model training, higher weights were assigned to minority classes and lower weights to majority classes, forcing the learning algorithms to pay greater attention to underrepresented traffic states without altering the original data distribution. This approach provided a better balance between overall accuracy and minority-class recognition and was therefore retained for all experiments reported in this study.

The effectiveness of this strategy is reflected in the balanced accuracy and Macro-F1 score obtained by the proposed models, which provide a more reliable assessment than overall accuracy when dealing with highly imbalanced multiclass traffic-state classification problems.

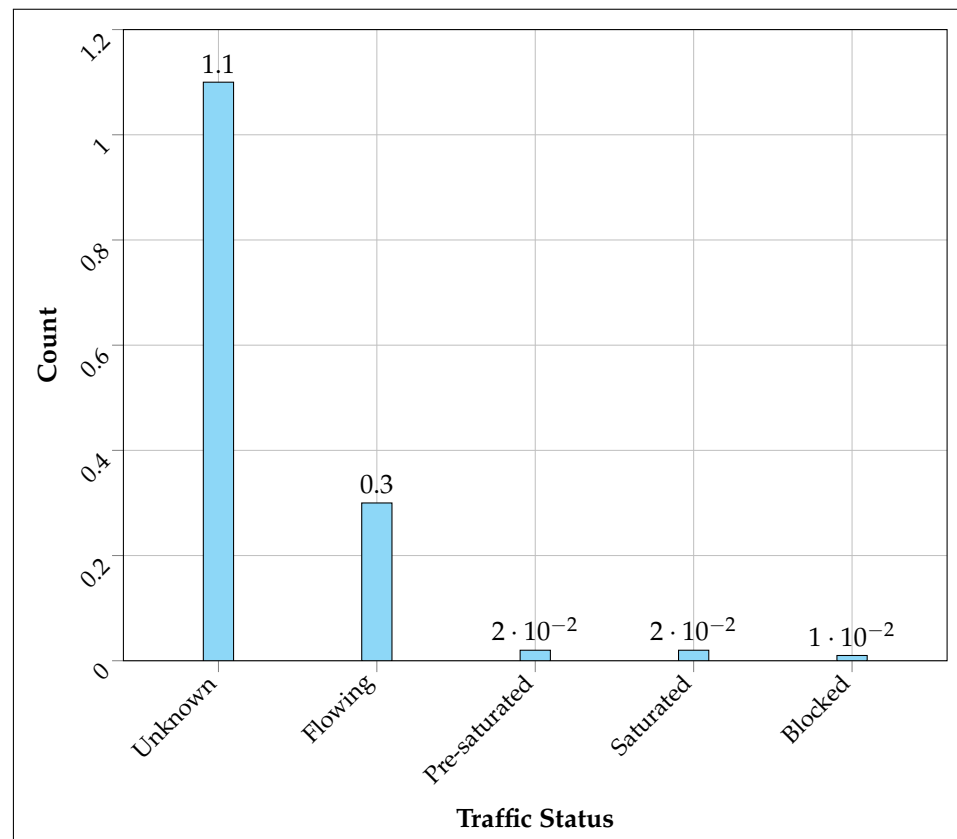


Figure 3. Distribution of Traffic Status in the Dataset.

4. Methodology

4.1. Overview of the Proposed Approach

The proposed study aims to classify urban traffic states from real-world traffic-sensor data collected at the Alésia intersection in Paris. To ensure reliable and reproducible traffic-state prediction, a complete data-processing and modeling pipeline was designed, including data preparation, temporal-sequence construction, class-imbalance handling, deep-learning model training, and performance evaluation. The overall workflow of the proposed approach is illustrated in Figure 4.

As illustrated in Figure 4, the proposed approach consists of a sequence of preprocessing, feature-engineering, temporal-modeling, and evaluation stages designed to address the challenges associated with urban traffic-state classification. Since traffic conditions evolve over time and exhibit strong temporal dependencies, the construction of temporal sequences represents a critical step in the proposed methodology. Rather than treating each observation as an independent sample, consecutive traffic measurements are organized into fixed-length sequences to capture short-term traffic dynamics and temporal correlations. The following subsection describes the temporal sequence construction process and the rationale behind the selected sequence length used as input to the deep-learning models.

4.2. Temporal Sequence Construction

Traffic conditions are inherently dynamic and evolve continuously over time. Consequently, treating individual observations as independent samples may fail to capture the temporal dependencies that characterize urban traffic behavior. To address this limitation, the cleaned dataset was first sorted chronologically according to the *Count_date_time* attribute, ensuring that the temporal order of traffic observations was preserved throughout the modeling process.

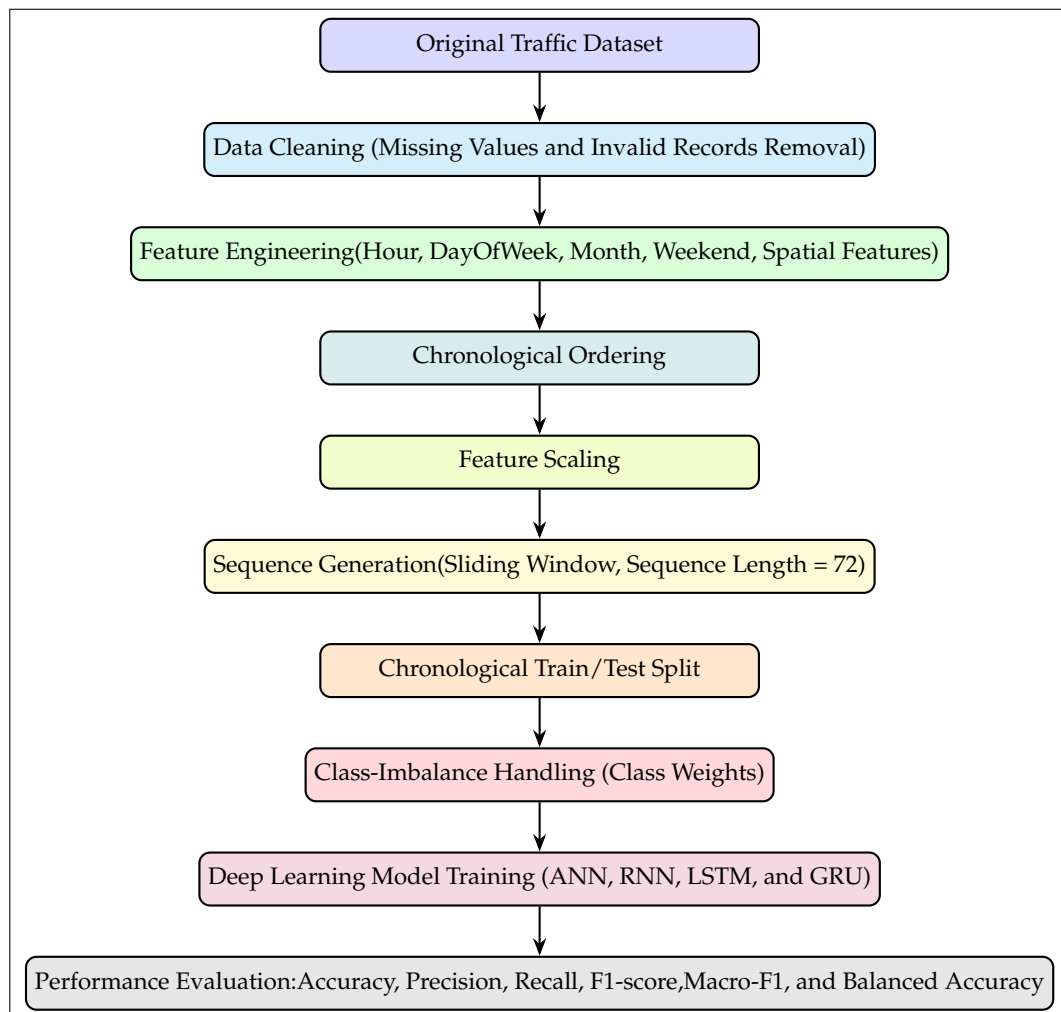


Figure 4. Overview of the proposed traffic-state classification approach, including data preprocessing, temporal sequence construction, class-imbalance handling, model training, and performance evaluation.

After chronological ordering, a temporal sequence construction procedure was applied using a sliding-window strategy. Instead of using a single observation as input, each sample was represented by a sequence of consecutive traffic measurements. This approach enables the deep-learning models to learn temporal patterns, traffic evolution trends, and short-term dependencies that may influence future traffic states.

Let X_t denote the feature vector observed at time t . A sequence of length L was generated by grouping the previous L consecutive observations:

$$S_t = X_{t-L+1}, X_{t-L+2}, \dots, X_t \quad (1)$$

where S_t represents the input sequence associated with the target traffic-state label at time $t + 1$.

In this study, a sequence length of $L = 72$ was adopted. The selected window size was empirically determined to provide an appropriate balance between temporal information retention and computational efficiency. By considering the previous 72 consecutive observations, the models are able to capture recurring traffic patterns, temporal dependencies, and short-term fluctuations that characterize urban traffic dynamics, while avoiding excessive model complexity and training time.

The sliding-window mechanism was applied sequentially across the entire chronologically ordered dataset. At each step, the window was shifted by one observation, generating

overlapping sequences and increasing the number of training samples available for model learning. This strategy preserves temporal continuity while maximizing the utilization of the available traffic data.

Following sequence generation, the resulting data were organized into a three-dimensional structure of the form:

$$(N, L, F) \quad (2)$$

where N denotes the number of generated sequences, $L = 72$ represents the sequence length, and F corresponds to the number of input features used for traffic-state classification. In the proposed approach, the input features include the traffic-related variables (*Hourly Debit* and *Occupancy Rate*), the engineered temporal features (*Hour*, *DayOfWeek*, *Month*, and *Weekend*), and the spatial descriptors (*Arc Identifier*, *Upstream Node Identifier*, and *Downstream Node Identifier*). This representation is particularly suitable for recurrent neural-network architectures, including RNN, LSTM, and GRU models, which are specifically designed to learn temporal relationships from sequential data.

The generated sequences were subsequently divided into training and testing subsets using a chronological train–test split procedure. Unlike random partitioning, chronological splitting preserves the natural temporal ordering of observations and prevents information leakage from future samples into the training process. This strategy provides a more realistic assessment of model performance under real-world traffic-monitoring and forecasting conditions.

4.3. Feature Representation and Data Preparation

To accurately characterize traffic conditions, each observation was represented using a combination of traffic-related, temporal, and spatial features extracted from the cleaned dataset. The traffic-related variables include *Hourly Debit* and *Occupancy Rate*, which provide direct information about traffic demand and road utilization. In addition, several temporal features were derived from the *Count_date_time* attribute during the feature-engineering stage, namely *Hour*, *DayOfWeek*, *Month*, and a binary *Weekend* indicator. Spatial information was incorporated through the *Arc Identifier*, *Upstream Node Identifier*, and *Downstream Node Identifier*, which describe the location and connectivity of the monitored road segment within the traffic network.

Consequently, each traffic observation was represented using nine input features combining traffic, temporal, and spatial information. This multidimensional representation enables the deep-learning models to capture complementary characteristics influencing traffic-state evolution and classification.

Although traffic conditions may also be influenced by external factors such as weather conditions, road incidents, special events, and network-level interactions, such information was not available in the official dataset used in this study. Consequently, the proposed models were developed using the traffic, temporal, and spatial variables provided by the traffic-monitoring system. Future work will investigate the integration of additional contextual variables and feature-importance analyses to further improve traffic-state classification performance.

Prior to sequence generation and model training, feature scaling was applied to normalize the input variables and reduce the influence of differences in measurement units and value ranges. Since the selected features exhibit heterogeneous numerical scales, normalization ensures a more stable optimization process and prevents variables with larger magnitudes from dominating the learning procedure. The normalized features were subsequently used to construct the temporal sequences described in the previous subsection.

After sequence generation, the resulting samples were divided into training and testing subsets using a chronological train–test split strategy. Unlike random partitioning,

chronological splitting preserves the temporal ordering of observations and prevents future information from being introduced into the training process. This procedure is particularly important for traffic forecasting and traffic-state classification applications, where the objective is to predict future conditions from historical observations. By maintaining the natural temporal structure of the dataset, the evaluation process more accurately reflects real-world deployment scenarios and provides a reliable assessment of model generalization capability.

4.4. Class Weight Strategy

As illustrated in Figure 3, the traffic-state dataset exhibits a highly imbalanced class distribution. The majority of observations belong to the *Flowing* and *Unknown* categories, whereas the *Blocked*, *Pre-saturated*, and *Saturated* classes contain considerably fewer samples. Such imbalance may bias the learning process toward majority classes and reduce the ability of deep-learning models to correctly recognize minority traffic states.

To address this issue, several imbalance-handling strategies were investigated during preliminary experiments. In particular, data-level resampling techniques were evaluated to improve the representation of minority classes. However, these approaches did not provide consistent improvements in model generalization and sometimes introduced additional noise into the learning process. Consequently, a cost-sensitive learning strategy based on class weights was adopted for all experiments reported in this study.

The class-weighting mechanism assigns a larger penalty to classification errors associated with minority classes and a smaller penalty to errors related to majority classes. During model training, these weights are incorporated into the loss function, encouraging the optimization process to pay greater attention to underrepresented traffic states while preserving the original class distribution.

Let N denote the total number of training samples, K the number of traffic-state classes, and n_i the number of samples belonging to class i . The class weight associated with class i is computed as:

$$w_i = \frac{N}{K \times n_i} \quad (3)$$

where larger weights are assigned to minority classes and smaller weights to majority classes.

Compared with data-resampling approaches, the class-weight strategy preserves the original characteristics of the traffic dataset while reducing the bias toward dominant classes. This property is particularly important for traffic-state classification problems, where minority classes often correspond to critical traffic conditions such as congestion, saturation, or blockage events. The effectiveness of this strategy is subsequently assessed using class-sensitive evaluation metrics, including Macro-F1 score and Balanced Accuracy.

4.5. Deep Learning Architectures

To evaluate the effectiveness of deep-learning techniques for multiclass traffic-state classification, four neural-network architectures were investigated in this study: ANN, RNN, LSTM, and GRU. These architectures were selected because they represent different levels of complexity and memory capabilities for modeling traffic dynamics.

The ANN model serves as a baseline deep-learning approach and processes traffic observations without explicitly modeling temporal dependencies. In contrast, the RNN, LSTM, and GRU architectures are specifically designed to learn sequential patterns from temporal data. Since traffic conditions evolve continuously over time, these recurrent architectures are expected to better capture the temporal relationships embedded within the generated traffic sequences.

To ensure a fair comparison, all models were trained using the same preprocessed dataset, identical train–test partitions, the same class-weight strategy, and con-

sistent evaluation metrics. The architectural details of each model are described in the following subsections.

4.5.1. Artificial Neural Network (ANN)

The Artificial Neural Network (ANN) was employed as a baseline classifier for traffic-state prediction. The architecture consists of fully connected hidden layers followed by a Softmax output layer that produces the probability distribution over the five traffic-state categories. Unlike recurrent architectures, the ANN processes each input independently and therefore does not explicitly model temporal dependencies between consecutive observations. Nevertheless, it provides a useful benchmark for assessing the contribution of temporal learning mechanisms in recurrent models.

4.5.2. Simple Recurrent Neural Network (RNN)

The Simple Recurrent Neural Network (RNN) extends the ANN architecture by incorporating recurrent connections that enable information from previous observations to influence the current prediction. This memory mechanism allows the network to capture short-term temporal dependencies within traffic sequences. However, standard RNNs may suffer from gradient-vanishing problems when processing long sequences, limiting their ability to learn long-range temporal patterns.

4.5.3. Long Short-Term Memory Network (LSTM)

To overcome the limitations of conventional RNNs, a Long Short-Term Memory (LSTM) architecture was implemented. LSTM networks incorporate memory cells and gating mechanisms that regulate the flow of information through time, enabling the learning of both short-term and long-term temporal dependencies.

The proposed LSTM architecture consists of two stacked LSTM layers containing 128 and 64 memory units, respectively. Batch Normalization layers were introduced after each LSTM layer to stabilize the learning process, while Dropout layers with a dropout rate of 0.3 were employed to reduce overfitting. The extracted temporal features were subsequently processed through two fully connected layers containing 64 and 32 neurons before being fed into a Softmax output layer responsible for multiclass traffic-state classification.

4.5.4. Gated Recurrent Unit Network (GRU)

The Gated Recurrent Unit (GRU) network represents a simplified variant of the LSTM architecture. GRUs employ update and reset gates to control information flow while using fewer parameters than LSTM networks. This reduced architectural complexity often leads to faster training while maintaining competitive predictive performance.

Similar to the LSTM model, the GRU architecture was designed to capture temporal traffic patterns from sequential observations and evaluate whether a lighter recurrent structure could achieve comparable classification performance under highly imbalanced traffic conditions.

4.6. Training Configuration

To ensure a consistent and fair comparison among the investigated deep-learning architectures, all models were trained under similar experimental conditions. The same training and testing datasets, sequence length, feature representation, and class-weight strategy were employed throughout the experiments.

Model optimization was performed using the Adam optimizer, which is widely adopted in deep-learning applications because of its adaptive learning-rate mechanism and computational efficiency. The Sparse Categorical Cross-Entropy loss function was selected to address the multiclass traffic-state classification problem.

A maximum of 50 training epochs was specified for all recurrent architectures, with a batch size of 16 samples as shown in Table 5. To improve model generalization and reduce the risk of overfitting, an Early Stopping strategy was implemented. During training, the validation loss was continuously monitored, and the training process was automatically terminated when no further improvement was observed for a predefined number of epochs.

To prevent overfitting and improve the generalization capability of the deep learning models, an Early Stopping strategy was employed during training. The maximum number of training epochs was set to 50; however, training was automatically terminated when the validation loss did not improve for a predefined number of consecutive epochs. The model weights corresponding to the best validation performance were restored for the final evaluation.

In addition, a ReduceLROnPlateau mechanism was incorporated to dynamically adjust the learning rate whenever the validation performance stagnated. This strategy enables a more stable optimization process and facilitates convergence toward better-performing solutions.

Given the highly imbalanced nature of the traffic-state dataset, the class-weight values computed in the previous subsection were integrated into the training process. By assigning larger penalties to minority classes and smaller penalties to majority classes, the models were encouraged to pay greater attention to underrepresented traffic states while preserving the original class distribution.

The final model performance was evaluated on the unseen testing dataset using Accuracy, Precision, Recall, F1-score, Macro-F1 score, and Balanced Accuracy metrics. These complementary evaluation measures provide a comprehensive assessment of classification performance, particularly in the presence of class imbalance.

The hyperparameter values reported in Table 5 were selected based on preliminary exploratory experiments and commonly adopted practices in deep-learning applications for time-series classification. The objective was to obtain stable convergence while maintaining a reasonable computational cost. Formal hyperparameter optimization techniques such as Grid Search, Random Search, or Bayesian Optimization were not considered within the scope of the present study. Future work will investigate automated hyperparameter optimization strategies to further improve model performance.

Table 5. Training Hyperparameters.

Parameter	Value
Sequence Length	72
Batch Size	16
Maximum Epochs	50
Optimizer	Adam
Loss Function	Sparse Categorical Cross-Entropy
Output Activation	Softmax
Early Stopping	Yes
ReduceLROnPlateau	Yes
Class Weights	Yes

The proposed approach integrates several complementary components to address the challenges of real-world traffic-state classification. By combining temporal sequence construction, spatial and temporal feature representation, chronological data partitioning, and cost-sensitive learning through class weighting, the methodology is specifically designed to handle the dynamic and highly imbalanced nature of urban traffic data. Furthermore, the comparative evaluation of ANN, RNN, LSTM, and GRU architectures under identical experimental conditions provides a comprehensive assessment of their ability to model

complex traffic patterns and recognize both majority and minority traffic states. Through this unified approach, the study aims to establish a reliable and reproducible benchmark for multiclass traffic-state classification using real-world traffic observations. The effectiveness of the proposed methodology and the comparative performance of the investigated deep-learning models are presented and discussed in the following section.

5. Results and Discussion

This section presents a comprehensive evaluation of the four deep learning architectures investigated in this study, namely ANN, RNN, LSTM, and GRU, for multiclass traffic-state classification. Model performance is assessed using several complementary metrics, including Accuracy, Precision, Recall, F1-score, Balanced Accuracy, confusion matrices, and class-wise classification reports. In addition, the learning dynamics of each architecture are analyzed through training and testing accuracy and loss curves in order to evaluate convergence behavior, model stability, and generalization capability.

Figure 5 presents the learning curves of the ANN model. The training accuracy increases progressively from approximately 24% to 62.2%, while the testing accuracy follows a similar trend and reaches 61.3% at the final epoch. Although both curves exhibit a consistent upward progression, the overall performance remains considerably lower than that achieved by the recurrent architectures. The relatively small gap between training and testing accuracy indicates that the model does not suffer from severe overfitting. However, the modest classification performance suggests that a feedforward ANN is unable to fully capture the temporal dependencies and sequential patterns inherent in traffic-state evolution.

The loss curves further support this observation. Both training and testing losses decrease steadily throughout the learning process, indicating stable optimization and effective parameter updates. Nevertheless, the final loss values remain relatively high compared with those obtained by the recurrent models, suggesting that important temporal information is not adequately represented by the ANN architecture. These findings demonstrate that although the ANN provides a reasonable baseline, it is not sufficiently powerful for complex traffic-state prediction tasks involving temporal dynamics.

The performance of the Simple RNN model is illustrated in Figure 6. Compared with the ANN, the RNN achieves substantially higher predictive performance, with training and testing accuracies stabilizing around 91.4% and 90.3%, respectively. The improvement can be attributed to the recurrent connections that enable the model to exploit temporal dependencies between consecutive traffic observations. Throughout training, the two accuracy curves remain relatively close, indicating satisfactory generalization and limited overfitting.

The corresponding loss curves exhibit a continuous decrease, with the training loss dropping from approximately 1.35 to 0.72 and the testing loss stabilizing near 0.30. Although several fluctuations can be observed in the testing accuracy, these variations remain limited and do not indicate unstable learning behavior. The results confirm that incorporating temporal information significantly enhances classification performance compared with the ANN model. However, the oscillatory behavior of the testing accuracy suggests that the Simple RNN may still encounter difficulties in preserving long-term contextual information due to the well-known vanishing-gradient problem.

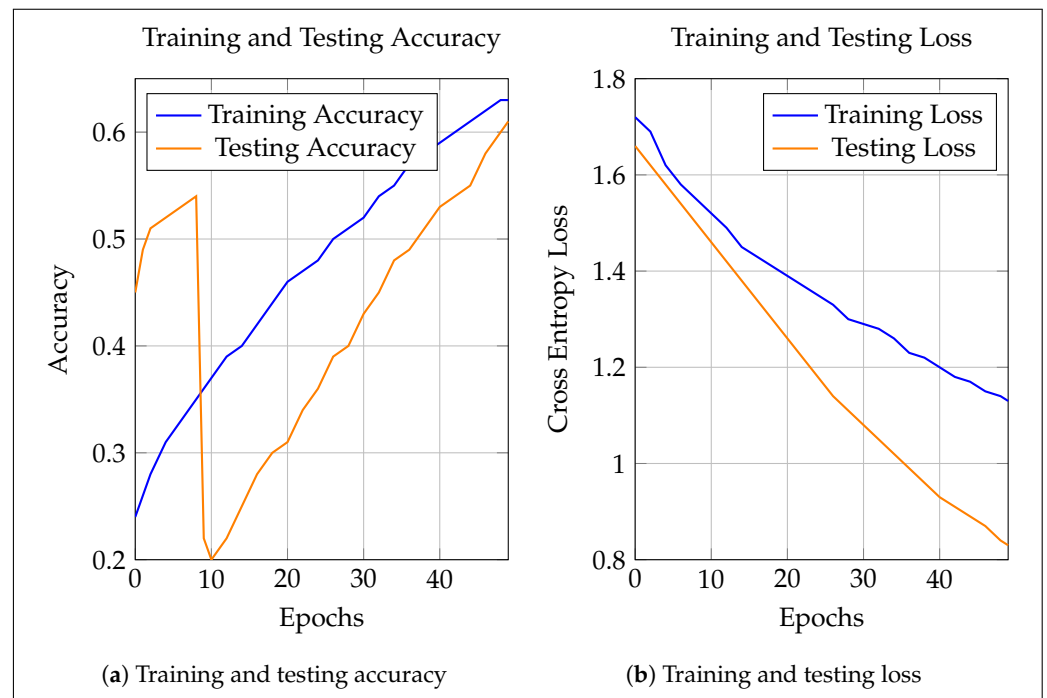


Figure 5. Learning curves of the ANN model over 50 epochs: (a) accuracy; (b) loss.

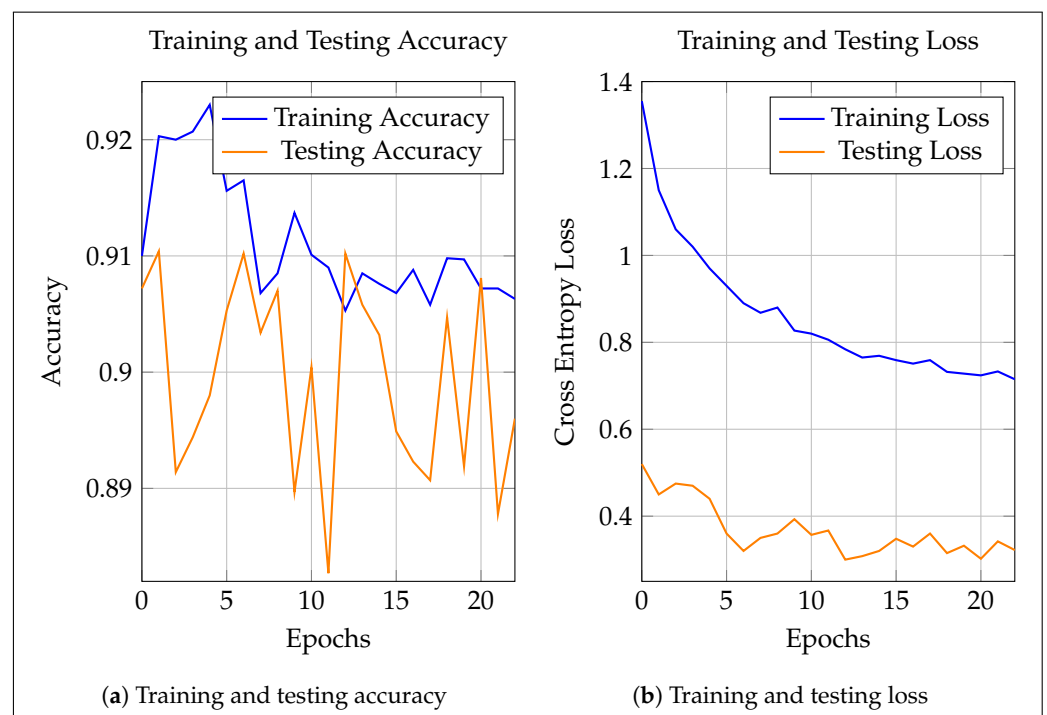


Figure 6. Learning curves of the RNN model over 23 epochs: (a) accuracy and (b) loss.

Figure 7 presents the results obtained using the LSTM architecture. The training accuracy remains highly stable throughout the learning process, converging to approximately 91.4%, while the testing accuracy stabilizes around 90.3%. The close agreement between training and testing curves indicates excellent generalization capability. Furthermore, the Early Stopping mechanism halted the training process after ten epochs, demonstrating rapid convergence and preventing unnecessary iterations that could potentially lead to overfitting.

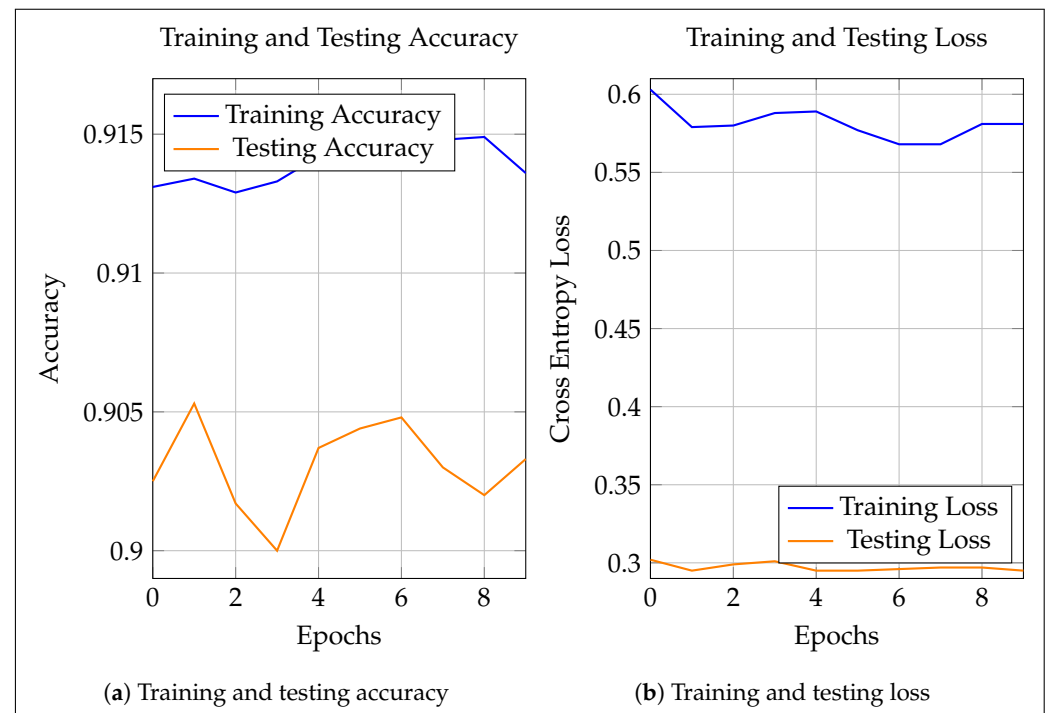


Figure 7. Learning curves of the LSTM model over 50 epochs: (a) accuracy and (b) loss.

The loss curves reveal a similar behavior. Training loss remains stable around 0.58, whereas testing loss rapidly converges and stabilizes near 0.30. The absence of significant divergence between training and testing losses indicates that the model effectively learns representative traffic patterns without memorizing the training data. The memory-cell mechanism of LSTM enables the preservation of long-term temporal dependencies, resulting in a more robust representation of traffic evolution and improved stability compared with the Simple RNN architecture.

The GRU model results are shown in Figure 8. Among all evaluated architectures, GRU achieves the highest classification performance, reaching approximately 91.96% training accuracy and 90.74% testing accuracy. The training and testing curves exhibit a highly consistent behavior throughout the learning process, with only minor fluctuations. The limited gap between the two curves indicates excellent generalization and demonstrates that the model successfully captures the underlying traffic dynamics without overfitting.

The corresponding loss curves show a substantial reduction during training, with the training loss decreasing from approximately 1.19 to 0.47 and the testing loss converging near 0.30. Compared with the LSTM model, GRU achieves slightly better accuracy while maintaining a simpler architecture and lower computational complexity. This result can be explained by the efficient gating mechanism of GRU, which effectively models temporal relationships while requiring fewer parameters and shorter training times than LSTM.

A comparative analysis of the four architectures highlights the importance of temporal modeling for traffic-state classification. The ANN model exhibits the lowest performance because it processes observations independently and therefore cannot exploit sequential dependencies. In contrast, the recurrent architectures consistently achieve substantially higher classification accuracy. Among them, the Simple RNN improves predictive capability but remains sensitive to temporal information loss. Both LSTM and GRU provide superior stability and generalization through their gating mechanisms, which alleviate the vanishing-gradient problem and facilitate the learning of long-term traffic patterns.

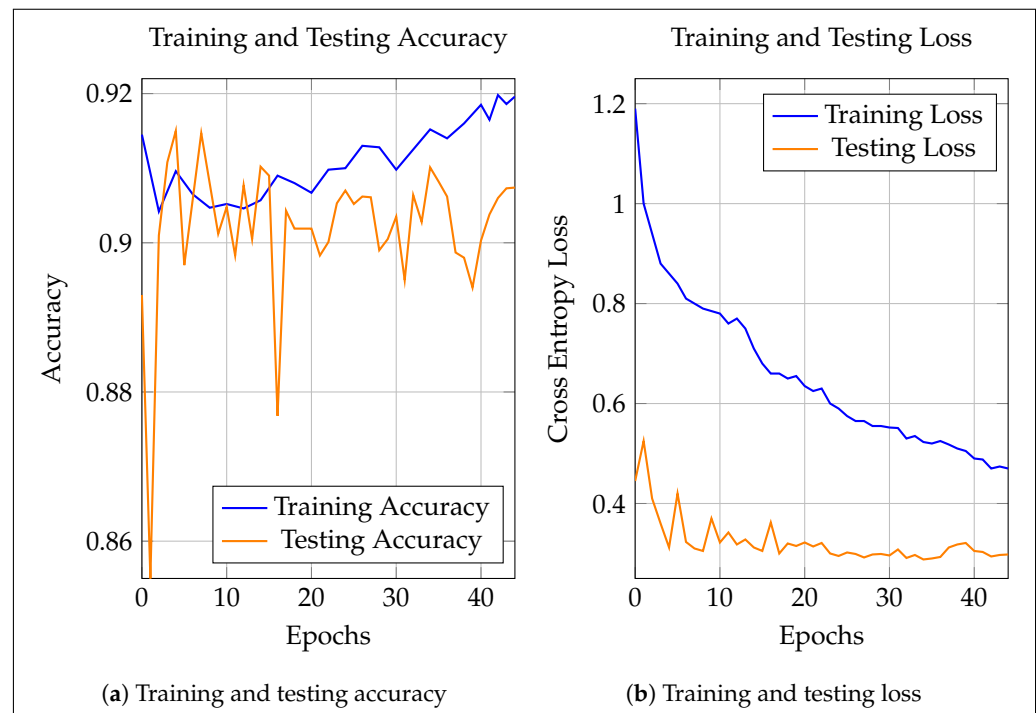


Figure 8. Learning curves of the GRU model across 45 epochs: (a) accuracy and (b) loss.

To further investigate the classification capability of each architecture, the following subsection presents a detailed quantitative evaluation based on the classification reports obtained for both training and testing datasets. Subsequently, confusion matrices are analyzed to identify class-specific strengths and misclassification patterns, providing deeper insight into the behavior of each model under real-world traffic conditions.

Tables 6 and 7 present the classification performance of the ANN model on the training and testing datasets. The model achieves comparable results on both datasets, with an accuracy of 62% for training and 61% for testing, indicating stable learning and satisfactory generalization capability. The highest performance is obtained for the *Blocked* and *Flowing* traffic states, which achieve relatively high precision and recall values. In contrast, the *Pre-saturated*, *Saturated*, and *Unknown* classes exhibit lower F1-scores, highlighting the difficulty of correctly identifying minority traffic conditions. Moreover, the close agreement between the training and testing metrics confirms that the ANN model does not exhibit significant overfitting, although its ability to discriminate between underrepresented classes remains limited.

Table 6. ANN Training Classification Report.

Class	Precision	Recall	F1-Score	Support
Blocked	0.68	0.90	0.77	202
Flowing	0.99	0.62	0.76	23,153
Pre-saturated	0.07	0.69	0.13	402
Saturated	0.46	0.45	0.46	161
Unknown	0.05	0.53	0.09	510
Accuracy	—	—	0.62	24,428
Macro Avg	0.45	0.64	0.44	24,428
Weighted Avg	0.95	0.62	0.74	24,428

Table 7. ANN Testing Classification Report.

Class	Precision	Recall	F1-Score	Support
Blocked	0.64	0.88	0.74	50
Flowing	0.98	0.61	0.75	5788
Pre-saturated	0.08	0.76	0.15	101
Saturated	0.46	0.30	0.36	40
Unknown	0.04	0.50	0.08	128
Accuracy	–	–	0.61	6107
Macro Avg	0.44	0.61	0.42	6107
Weighted Avg	0.94	0.61	0.73	6107

Tables 8 and 9 summarize the classification performance of the RNN model on the training and testing datasets. The model achieves an overall accuracy of approximately 91% on both datasets, with a weighted F1-score of 0.92, indicating a substantial improvement compared with the ANN model. The close agreement between the training and testing metrics demonstrates stable learning behavior and strong generalization capability, with no evidence of significant overfitting.

Table 8. RNN Training Classification Report.

Class	Precision	Recall	F1-Score	Support
Blocked	0.26	0.84	0.40	202
Flowing	0.98	0.94	0.96	23,095
Pre-saturated	0.19	0.03	0.05	402
Saturated	0.17	0.40	0.24	161
Unknown	0.31	0.76	0.44	510
Accuracy	–	–	0.91	24,370
Macro Avg	0.38	0.59	0.42	24,370
Weighted Avg	0.94	0.91	0.92	24,370

Table 9. RNN Testing Classification Report.

Class	Precision	Recall	F1-Score	Support
Blocked	0.20	0.68	0.31	50
Flowing	0.98	0.94	0.96	5774
Pre-saturated	0.09	0.01	0.02	101
Saturated	0.08	0.20	0.11	40
Unknown	0.34	0.76	0.47	128
Accuracy	–	–	0.91	6093
Macro Avg	0.34	0.52	0.37	6093
Weighted Avg	0.94	0.91	0.92	6093

At the class level, the RNN model performs particularly well in recognizing the dominant *Flowing* traffic state, achieving an F1-score of 0.96 for both training and testing datasets. Furthermore, the *Blocked* and *Unknown* classes are identified more effectively than with the ANN model. However, lower precision and F1-score values are still observed for the *Pre-saturated* and *Saturated* classes, indicating that the classification of minority traffic states remains challenging. Despite these limitations, the results confirm that incorporating temporal dependencies through recurrent neural networks significantly enhances traffic-state classification performance.

Tables 10 and 11 present the classification performance of the LSTM model on the training and testing datasets. The model achieves high and consistent performance, with

accuracies of 91% and 90% on the training and testing sets, respectively, and weighted F1-scores exceeding 0.92. The close agreement between training and testing metrics indicates stable learning and good generalization capability without significant overfitting.

Table 10. LSTM Training Classification Report.

Class	Precision	Recall	F1-Score	Support
Blocked	0.38	0.84	0.52	202
Flowing	0.99	0.92	0.95	23,095
Pre-saturated	0.25	0.39	0.30	402
Saturated	0.22	0.89	0.35	161
Unknown	0.38	0.87	0.53	510
Accuracy	–	–	0.91	24,370
Macro Avg	0.44	0.78	0.53	24,370
Weighted Avg	0.96	0.91	0.93	24,370

Table 11. LSTM Testing Classification Report.

Class	Precision	Recall	F1-Score	Support
Blocked	0.32	0.76	0.45	50
Flowing	0.99	0.92	0.95	5774
Pre-saturated	0.19	0.35	0.25	101
Saturated	0.18	0.72	0.29	40
Unknown	0.38	0.80	0.52	128
Accuracy	–	–	0.90	6093
Macro Avg	0.41	0.71	0.49	6093
Weighted Avg	0.95	0.90	0.92	6093

Compared with the ANN and standard RNN models, the LSTM provides a more balanced classification performance across traffic states. In particular, improvements are observed for the minority classes *Pre-saturated*, *Saturated*, and *Unknown*, resulting in higher recall and F1-score values. Furthermore, the model maintains excellent performance for the dominant *Flowing* class, achieving an F1-score of 0.95 on both datasets. These results demonstrate the effectiveness of the LSTM architecture in capturing temporal dependencies and improving the recognition of complex traffic-state patterns.

Tables 12 and 13 summarize the classification performance of the GRU model on the training and testing datasets. The results show highly consistent performance across both datasets, indicating stable learning behavior and good generalization capability. The *Flowing* traffic state achieves the highest classification performance, with an F1-score of 0.96 and 0.95 on the training and testing sets, respectively.

Table 14 presents a comparative evaluation of the four deep learning architectures using overall performance metrics computed on the testing dataset. The results reveal substantial differences between the evaluated models. The ANN model achieves the lowest performance, with an accuracy of 61.26% and an F1-score of 0.7281, indicating limited capability to capture the temporal dependencies inherent in traffic-state evolution. Although its balanced accuracy remains comparable to that of some recurrent models, its overall predictive performance is significantly lower.

Table 12. GRU Training Classification Report.

Class	Precision	Recall	F1-Score	Support
Blocked	0.41	0.93	0.57	202
Flowing	0.99	0.93	0.96	23,095
Pre-saturated	0.32	0.48	0.38	402
Saturated	0.31	0.92	0.47	161
Unknown	0.39	0.92	0.55	510
Accuracy	–	–	0.92	24,370
Macro Avg	0.48	0.84	0.59	24,370
Weighted Avg	0.96	0.92	0.93	24,370

Table 13. GRU Testing Classification Report.

Class	Precision	Recall	F1-Score	Support
Blocked	0.33	0.76	0.46	50
Flowing	0.99	0.93	0.95	5774
Pre-saturated	0.24	0.37	0.29	101
Saturated	0.20	0.60	0.29	40
Unknown	0.36	0.76	0.49	128
Accuracy	–	–	0.91	6093
Macro Avg	0.42	0.68	0.50	6093
Weighted Avg	0.95	0.91	0.93	6093

Table 14. Comparison of the overall performance metrics obtained by the evaluated deep learning models on the testing dataset.

Model	Accuracy	Balanced Accuracy	F1-Score	Macro F1-Score
ANN	0.6126	0.6109	0.7281	0.4200
RNN	0.9102	0.5168	0.9204	0.3700
LSTM	0.9025	0.7091	0.9217	0.4900
GRU	0.9101	0.6821	0.9256	0.4970

The recurrent architectures consistently outperform the ANN model, highlighting the importance of temporal information for traffic-state classification. While the conventional RNN achieves a high accuracy of 91.02%, its balanced accuracy (51.68%) and Macro F1-score (0.37) remain relatively low, suggesting that its predictions are strongly influenced by the dominant traffic class and that minority traffic states are less effectively recognized.

Among the advanced recurrent architectures, both LSTM and GRU provide superior and more balanced performance. The LSTM achieves the highest balanced accuracy (70.91%), demonstrating its strong ability to handle class imbalance and correctly identify minority traffic states. In contrast, the GRU attains the highest overall F1-score (0.9256) and Macro F1-score (0.497), while maintaining a competitive accuracy of 91.01%. These results indicate that the GRU offers the most favorable compromise between overall predictive performance and class-wise recognition capability.

Overall, the comparative analysis confirms that incorporating gated recurrent mechanisms significantly improves traffic-state classification performance. While the LSTM demonstrates greater robustness to class imbalance, the GRU achieves the best overall trade-off between accuracy, generalization capability, and balanced recognition of different traffic conditions, making it the most suitable architecture for the considered traffic-state classification task.

Figures 9–12 provide additional insights into the class-wise behavior of the evaluated models. For the ANN model, substantial confusion can be observed between minority traffic states and the dominant classes, which explains its relatively low overall classification performance. Although the model correctly identifies a large proportion of the *Flowing* observations, it struggles to distinguish the *Pre-saturated*, *Saturated*, and *Unknown* traffic states.

The RNN model significantly improves the recognition of the dominant *Flowing* class and achieves higher overall accuracy. However, the confusion matrices reveal that the model still encounters difficulties in correctly identifying the *Pre-saturated* and *Saturated* classes, resulting in frequent misclassifications of these minority traffic conditions.

In contrast, the LSTM and GRU architectures exhibit noticeably better discrimination of minority traffic states. The confusion matrices show that both models reduce the misclassification of the *Blocked*, *Saturated*, and *Unknown* classes, demonstrating their ability to capture more complex temporal dependencies and traffic evolution patterns. Although some confusion remains between neighboring traffic states, the GRU model achieves the most balanced classification behavior overall, while the LSTM provides the strongest recognition of minority classes, which is reflected by its superior Balanced Accuracy score. These findings further confirm the advantage of gated recurrent architectures for multiclass traffic-state classification under highly imbalanced real-world traffic conditions.

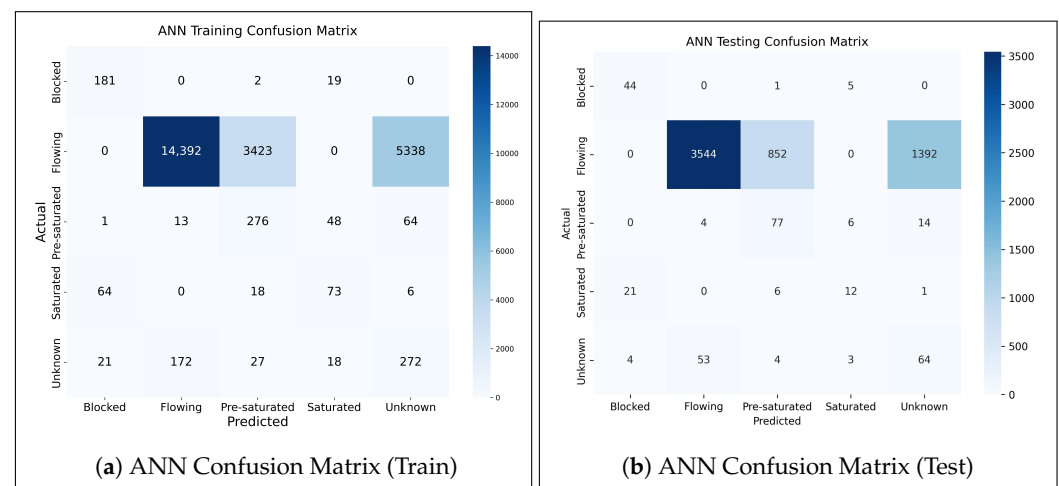


Figure 9. Confusion matrices of the ANN model for training and testing phases.

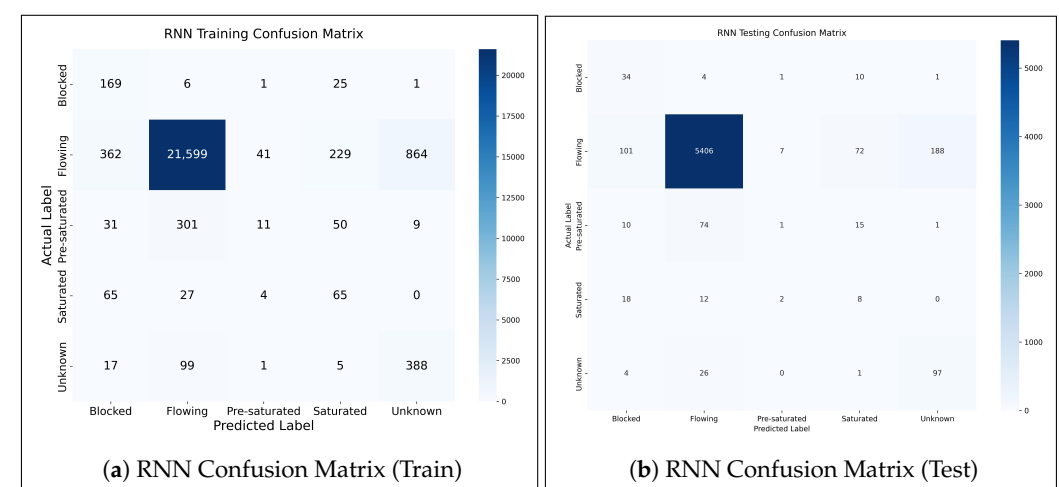


Figure 10. Confusion matrices of the RNN model for training and testing phases.

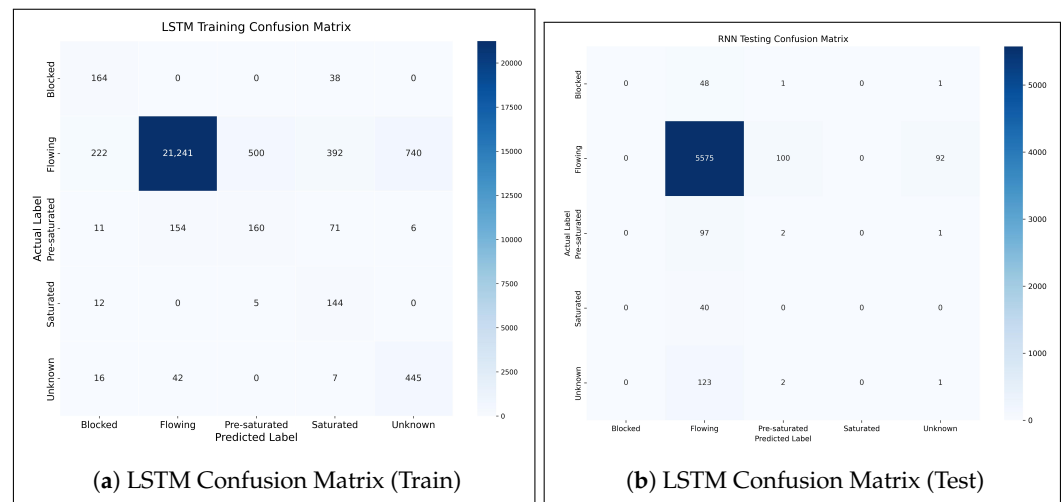


Figure 11. Confusion matrices of the LSTM model for training and testing phases.

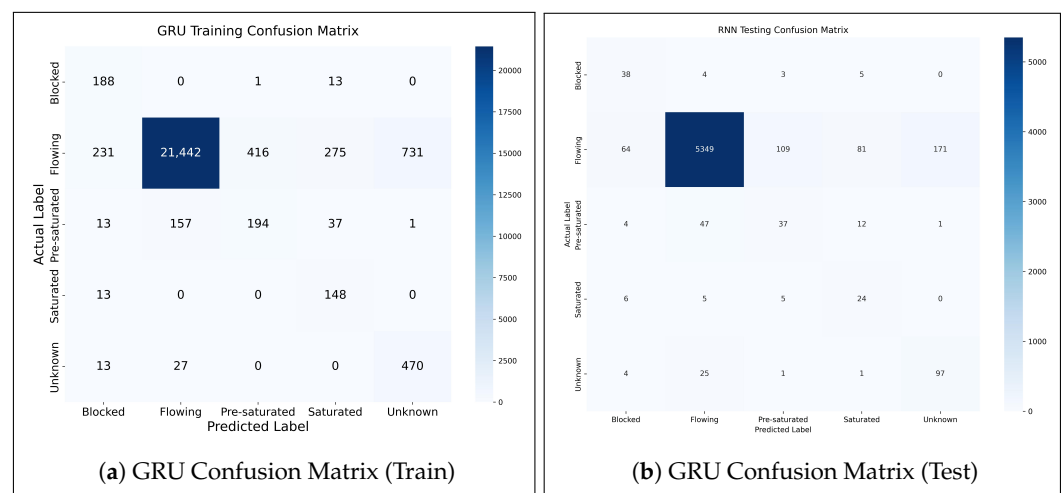


Figure 12. Confusion matrices of the GRU model for training and testing phases.

Overall, the experimental results highlight the importance of temporal modeling for traffic-state classification. While the ANN model provides acceptable performance for dominant traffic conditions, its limited ability to capture temporal dependencies leads to weaker recognition of minority traffic states. The introduction of recurrent architectures significantly improves classification performance, as evidenced by the substantial increase in Accuracy and F1-score obtained by the RNN, LSTM, and GRU models.

However, the comparative analysis demonstrates that overall Accuracy alone is insufficient for evaluating performance on highly imbalanced traffic datasets. Although the RNN achieves a high overall Accuracy, its lower class-wise performance indicates difficulties in recognizing underrepresented traffic states. In contrast, both LSTM and GRU exhibit a more balanced classification behavior, as confirmed by their improved recognition of minority classes in the classification reports and confusion matrices.

Among the evaluated architectures, the LSTM achieves the highest Balanced Accuracy, demonstrating a stronger capability to correctly identify different traffic conditions regardless of class frequency. Meanwhile, the GRU attains the highest overall F1-score and Macro F1-score while maintaining competitive Accuracy, indicating a favorable trade-off between global predictive performance and class-wise discrimination. These findings suggest that gated recurrent architectures are better suited for real-world traffic-state classification, where both dominant and minority traffic conditions must be accurately recognized.

The consistency observed between the learning curves, classification reports, confusion matrices, and comparative performance metrics further confirms the robustness of the obtained results and supports the effectiveness of recurrent deep-learning architectures for urban traffic-state prediction and monitoring applications.

6. Conclusions and Perspectives

This study investigated the effectiveness of four deep-learning architectures, namely ANN, RNN, LSTM, and GRU, for multiclass traffic-state classification using real-world traffic observations collected at the Alésia urban intersection in Paris. Unlike many traffic prediction studies that focus primarily on overall predictive accuracy, the present work addressed a highly imbalanced classification problem involving five traffic states (Flowing, Pre-saturated, Saturated, Blocked, and Unknown). To provide a comprehensive assessment of model performance, multiple complementary evaluation criteria were considered, including Accuracy, Precision, F1-score, Macro-F1 score, Balanced Accuracy, class-wise classification reports, learning curves, and confusion matrices.

The experimental results demonstrated clear differences among the evaluated architectures. The ANN model exhibited limited capability in handling complex traffic-state patterns and minority classes, resulting in the lowest overall performance. The conventional RNN substantially improved global accuracy; however, its lower Macro-F1 score and Balanced Accuracy revealed difficulties in correctly identifying underrepresented traffic conditions. These findings highlight the limitations of simple recurrent architectures when dealing with highly imbalanced traffic datasets and complex temporal dynamics.

The memory-based recurrent architectures achieved more robust and balanced results. The LSTM model obtained the highest Balanced Accuracy and showed superior recognition of minority traffic states, indicating a stronger ability to capture long-term temporal dependencies and preserve important contextual information. The GRU model achieved the best overall compromise between Accuracy, F1-score, and Macro-F1 score while maintaining a simpler architecture and lower computational complexity. Furthermore, the confusion matrices confirmed that both LSTM and GRU significantly reduced the misclassification of critical traffic states compared with ANN and conventional RNN models.

The analysis of the learning curves revealed stable convergence behavior and limited evidence of overfitting for the recurrent architectures. In addition, the use of Early Stopping contributed to efficient training while preserving generalization capability. These observations emphasize the importance of temporal modeling for traffic-state classification and demonstrate the effectiveness of gated recurrent mechanisms in handling real-world traffic data characterized by substantial class imbalance.

Overall, the results indicate that no single metric is sufficient to evaluate model performance in multiclass traffic-state classification. While GRU achieved the strongest overall predictive performance, LSTM provided the most balanced recognition of minority traffic conditions. Consequently, the choice between these architectures should depend on the operational objective, whether maximizing global predictive performance or improving the detection of less frequent but operationally critical traffic states.

It should also be noted that the conclusions of this study are restricted to the evaluated ANN, RNN, LSTM, and GRU architectures within the considered traffic-state classification framework. Recent approaches based on STGNNs and Transformer architectures have shown promising results for large-scale traffic-flow forecasting tasks. However, these methods are generally designed for traffic-flow regression problems and exploit explicit spatial relationships within transportation networks. In contrast, the present work focuses on traffic-state classification at a real urban intersection under highly imbalanced conditions.

Therefore, the findings presented here provide complementary insights into the behavior of recurrent deep-learning models in realistic urban traffic-management scenarios.

Future work will investigate the integration of additional contextual information, including weather conditions, traffic incidents, roadwork events, and signal-control parameters, to further improve prediction robustness. Moreover, extending the proposed framework to recent architectures such as STGNNs and Transformer-based models, as well as integrating the prediction system into adaptive traffic-management and ITS, represents a promising direction for real-time urban traffic optimization.

Author Contributions: Conceptualization, C.C. and A.Y.; methodology, C.C. and A.Y.; software, C.C.; validation, C.C., A.Y. and M.A.S.; formal analysis, C.C.; investigation, C.C.; data curation, C.C.; writing—original draft preparation, C.C.; writing—review and editing, A.Y., M.A.S., H.K., M.A.W., K.A.S. and B.A.E.-R.; visualization, C.C.; supervision, A.Y. and M.A.S. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to acknowledge the Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R757), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Data Availability Statement: The data used in this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. He, Z.; Guo, Q.; Wang, Z.; Li, X. Prediction of Monthly PM2.5 Concentration in Liaocheng in China Employing Artificial Neural Network. *Atmosphere* **2022**, *13*, 1221. [CrossRef]
2. Guo, Q.; He, Z.; Wang, Z. Predicting of Daily PM2.5 Concentration Employing Wavelet Artificial Neural Networks Based on Meteorological Elements in Shanghai, China. *Toxics* **2023**, *11*, 51. [CrossRef] [PubMed]
3. Guo, Q.; He, Z.; Wang, Z. Simulating daily PM2.5 concentrations using wavelet analysis and artificial neural network with remote sensing and surface observation data. *Chemosphere* **2023**, *340*, 139886. [CrossRef] [PubMed]
4. Guo, Q.; He, Z. Prediction of the confirmed cases and deaths of global COVID-19 using artificial intelligence. *Environ. Sci. Pollut. Res.* **2021**, *28*, 11672–11682. [CrossRef] [PubMed]
5. Guo, Q.; He, Z.; Li, S.; Li, X.; Meng, J.; Hou, Z.; Liu, J.; Chen, Y. Air Pollution Forecasting Using Artificial and Wavelet Neural Networks with Meteorological Conditions. *Aerosol Air Qual. Res.* **2020**, *20*, 1536–1546. [CrossRef]
6. Guo, Q.; He, Z.; Wang, Z. Monthly Climate Prediction Using Deep Convolutional Neural Network and Long Short-Term Memory. *Sci. Rep.* **2024**, *14*, 17748. [CrossRef] [PubMed]
7. Guo, Q.; He, Z.; Wang, Z.; Qiao, S.; Zhu, J.; Chen, J. A Performance Comparison Study on Climate Prediction in Weifang City Using Different Deep Learning Models. *Water* **2024**, *16*, 2870. [CrossRef]
8. He, Z.; Guo, Q. Comparative Analysis of Multiple Deep Learning Models for Forecasting Monthly Ambient PM2.5 Concentrations: A Case Study in Dezhou City, China. *Atmosphere* **2024**, *15*, 1432. [CrossRef]
9. Guo, Q.; He, Z.; Wang, Z. Assessing the Effectiveness of Long Short-Term Memory and Artificial Neural Network in Predicting Daily Ozone Concentrations in Liaocheng City. *Sci. Rep.* **2025**, *15*, 6798. [CrossRef] [PubMed]
10. Guo, Q.; He, Z.; Wang, Z. Prediction of Monthly Average and Extreme Atmospheric Temperatures in Zhengzhou Based on Artificial Neural Network and Deep Learning Models. *Front. For. Glob. Change* **2023**, *6*, 1249300. [CrossRef]
11. Shaaban, K.; Elamin, M.; Alsoub, M. Intelligent Transportation Systems in a Developing Country: Benefits and Challenges of Implementation. *Transp. Res. Procedia* **2021**, *55*, 1373–1380. [CrossRef]
12. Tettamanti, T.; Török, Á.; Varga, I. Dynamic road pricing for optimal traffic flow management by using nonlinear model predictive control. *IET Intell. Transp. Syst.* **2019**, *13*, 1139–1147. [CrossRef]
13. Pathak, P.; Singhal, A.; Bhutani, S. Intelligent Transportation System in India. *Telecom Bus. Rev. SITM J.* **2018**, *11*, 25–29. Available online: <http://www.publishingindia.com> (accessed on 25 April 2026).
14. Zang, J.; Jiao, P.; Liu, S.; Zhang, X.; Song, G.; Yu, L. Identifying Traffic Congestion Patterns of Urban Road Network Based on Traffic Performance Index. *Sustainability* **2023**, *15*, 948. [CrossRef]

15. Harrou, F.; Zeroual, A.; Kadri, F.; Sun, Y. Enhancing road traffic flow prediction with improved deep learning using wavelet transforms. *Results Eng.* **2024**, *23*, 102342. [[CrossRef](#)]
16. Zeroual, A.; Harrou, F.; Sun, Y.; Messai, N. Monitoring road traffic congestion using a macroscopic traffic model and a statistical monitoring scheme. *Sustain. Cities Soc.* **2017**, *35*, 494–510. [[CrossRef](#)]
17. Zeroual, A.; Harrou, F.; Sun, Y. Road traffic density estimation and congestion detection with a hybrid observer-based strategy. *Sustain. Cities Soc.* **2019**, *46*, 101411. [[CrossRef](#)]
18. Lana, I.; Ser, J.D.; Velez, M.; Vlahogianni, E.I. Road Traffic Forecasting: Recent Advances and New Challenges. *IEEE Intell. Transp. Syst. Mag.* **2018**, *10*, 93–109. [[CrossRef](#)]
19. Tian, K.; Tzigieras, A.; Wei, C.; Lee, Y.M.; Holmes, C.; Leonetti, M.; Merat, N.; Romano, R.; Markkula, G. Braking behaviour of automated vehicles as critical implicit communication for pedestrians' vehicle behaviour estimation and road crossing. *PsyArXiv* **2023**. [[CrossRef](#)]
20. Kamarianakis, Y.; Prastacos, P. Forecasting Traffic Flow Conditions in an Urban Network: Comparison of Multivariate and Univariate Approaches. *Transp. Res. Rec. J. Transp. Res. Board* **2003**, *1857*, 74–84. [[CrossRef](#)]
21. Rowan, D.; He, H.; Hui, F.; Yasir, A.; Mohammed, Q. A systematic review of machine learning-based microscopic traffic flow models and simulations. *Commun. Transp. Res.* **2025**, *5*, 100164. [[CrossRef](#)]
22. Komol, M.M.R.; Elhenawy, M.; Masoud, M.; Glaser, S.; Rakotonirainy, A.; Wood, M. Deep Transfer Learning Based Intersection Trajectory Movement Classification for Big Connected Vehicle Data. *IEEE Access* **2021**, *9*, 141830–141842. [[CrossRef](#)]
23. Juman, Z.A.M.S.; Masoud, M.; Bhuiyan, H.; Elhenawy, M.; Komol, M.M.R.; Battaia, O. A New Algorithm for Solving Uncapacitated Transportation Problem with Interval-Defined Demands and Suppliers Capacities. *J. Intell. Fuzzy Syst.* **2021**, *41*, 625–637. [[CrossRef](#)]
24. Parashuram, N.; Vijayalakshmi, K. A Comprehensive Analysis of Road Traffic Prediction Using Machine Learning Algorithms. In Proceedings of the 2024 First International Conference on Software, Systems and Information Technology (SSITCON), Tumkur, India, 18–19 October 2024; pp. 1–5. [[CrossRef](#)]
25. Alghamdi, T.; Mostafi, S.; Abdelkader, G.; Elgazzar, K. A Comparative Study on Traffic Modeling Techniques for Predicting and Simulating Traffic Behavior. *Future Internet* **2022**, *14*, 294. [[CrossRef](#)]
26. Komol, M.M.R.; Pinnow, J.; Elhenawy, M.; Yasmin, S.; Masoud, M.; Glaser, S. A Review on Drivers' Red Light Running Behavior Predictions and Technology-Based Countermeasures. *IEEE Access* **2022**, *10*, 25309–25326. [[CrossRef](#)]
27. Hussain, B.; Afzal, M.K.; Ahmad, S.; Mostafa, A.M. Intelligent Traffic Flow Prediction Using Optimized GRU Model. *IEEE Access* **2021**, *9*, 100736–100746. [[CrossRef](#)]
28. Shahariar, G.M.H.; Bodisco, T.A.; Surawski, N.; Komol, M.M.R.; Sajjad, M.; Chu-Van, T.; Ristovski, Z.; Brown, R.J. Real-driving CO₂, NO_x and Fuel Consumption Estimation Using Machine Learning Approaches. *Next Energy* **2023**, *1*, 100060. [[CrossRef](#)]
29. Luo, Q.; Lu, X.; Zang, Z.; Gong, H.; Guo, X.; Chen, X. A Real-Time Early Warning Approach for Multi-Dimensional Driving Risk of Heavy-Duty Trucks Using Trajectory Data. *Systems* **2026**, *14*, 204. [[CrossRef](#)]
30. Zhai, C.; Wu, W.; Xiao, Y.; Zhang, J.; Zhai, M.; Wu, Y. A Novel Throttle-Based Self-Stabilizing Control Scheme Integrated into an Anisotropic Continuum Model to Mitigate Cyber-Attacks in Connected Vehicle Scenarios. *Chaos Solitons Fractals* **2025**, *201*, 117319. [[CrossRef](#)]
31. Pan, Y.A.; Li, F.; Li, A.; Niu, Z.; Liu, Z. Urban Intersection Traffic Flow Prediction: A Physics-Guided Stepwise Framework Utilizing Spatio-Temporal Graph Neural Network Algorithms. *Multimodal Transp.* **2025**, *4*, 100207. [[CrossRef](#)]
32. Liu, S.; Wang, X. An Improved Transformer-Based Traffic Flow Prediction Model. *Sci. Rep.* **2025**, *15*, 8284. [[CrossRef](#)] [[PubMed](#)]
33. Joelianto, E.; Utami, F.P.; Sutarto, H.Y.; Gautama, S.; Semajski, I.; Fathurrahman, M.F. Performance Analysis of Max-Pressure Control System for Traffic Network Using Macroscopic Fundamental Diagram. *Int. J. Artif. Intell.* **2022**, *20*, 1–23.
34. Patel, H.; Gore, N.; Easa, S.; Arkatkar, S. Novel Traffic Conflict-Based Framework for Real-Time Traffic Safety Evaluation Under Heterogeneous and Weak Lane-Discipline Traffic *Transp. Res. Rec. J. Transp. Res. Board* **2023**, *2678*, 118–134. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.