

Review

Advances in Traffic Congestion Prediction: An Overview of Emerging Techniques and Methods

Aristeidis Mystakidis , Paraskevas Koukaras  and Christos Tjortjis * 

School of Science and Technology, International Hellenic University, 14th km Thessaloniki-Moudania, 57001 Thessaloniki, Greece; a.mystakidis@ihu.edu.gr (A.M.); p.koukaras@ihu.edu.gr (P.K.)

* Correspondence: c.tjortjis@ihu.edu.gr

Highlights:

What are the main findings?

- Traffic prediction involves multiple steps, including gathering and processing data while considering factors like weather, seasonality, and past traffic trends.
- While statistical models handle simple patterns well, machine learning, deep learning and ensemble techniques are better suited for complex and changing traffic conditions, though they come with challenges like overfitting and high computational demands.

What is the implication of the main finding?

- More accurate traffic predictions can improve urban mobility, reduce congestion, and minimize environmental impact by optimizing traffic flow.
- Advanced technologies and prediction approaches, such as new sequence-to-sequence modelling, may improve and balance accuracy with interpretability.

Abstract: The ongoing increase in urban populations has resulted in the enduring issue of traffic congestion, adversely affecting the quality of life, including commute duration, road safety, and local air quality. Consequently, recognizing and forecasting underlying traffic congestion patterns have become essential, with Traffic Congestion Prediction (TCP) emerging as an increasingly significant area of study. Advancements in Machine Learning (ML) and Artificial Intelligence (AI), as well as improvements in Internet of Things (IoT) sensor technologies have made TCP research crucial to the development of Intelligent Transportation Systems (ITSs). This review examines advanced TCP, emphasizing innovative forecasting methods and technologies and their importance for the ITS sector. This paper provides an overview of statistical, ML, Deep Learning (DL) approaches, and their ensembles that compose TCP. We examine several forecasting methods and discuss relative and absolute evaluation metrics from regression and classification perspectives. Finally, we present an overall step-by-step standard methodology that is often utilized in TCP problems. By combining these elements, this review highlights critical advancements and ongoing challenges in TCP, providing robust and detailed information for state-of-the-art ITS solutions.

Keywords: traffic congestion prediction; time series analysis; forecasting; machine learning; artificial neural networks; statistical methods



Academic Editor: Pierluigi Siano

Received: 13 December 2024

Revised: 28 January 2025

Accepted: 4 February 2025

Published: 7 February 2025

Citation: Mystakidis, A.; Koukaras, P.; Tjortjis, C. Advances in Traffic Congestion Prediction: An Overview of Emerging Techniques and Methods. *Smart Cities* **2025**, *8*, 25. <https://doi.org/10.3390/smartcities8010025>

Copyright: © 2025 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) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

There has been a notable global increase in urbanization rates in recent times. UN estimates indicate that by 2030, there will be around 4.9 billion people living in urban areas

worldwide, and by 2050, about 70% of people will be urban residents [1]. Traffic congestion has significantly increased as a result of the continued development of the urban area. This has far-reaching effects on road accidents, noise pollution, local air quality, and commute times [2]. Intelligent Transportation Systems (ITSs) are a well-established technology in the field of intelligent transportation that are used to improve the operational efficiency of transportation systems and optimize traffic flow. They are an essential component of the Internet of Things (IoT) framework.

Enhancing traffic movement efficiency and ensuring safety, while reducing travel times and fuel consumption, is the main goal of ITSs [3]. By decreasing the duration of time automobiles spend idling at red lights or intersections, ITSs may have a favorable effect, particularly in relation to local air quality [4]. This is due to the fact that cars often release more air pollutants when they stop and have their combustion engines in running status [5]. ITSs can forecast intersection density to regulate traffic signal systems and lessen traffic congestion by precisely counting the number of vehicles [6]. In order to create sustainable ITSs, it is imperative that IoT infrastructures are used more extensively, and that Information and Communication Technologies (ICTs) are used effectively.

An increasing quantity of traffic-related data is currently produced by these equipment and applications. This makes it possible to apply Machine Learning (ML) and Deep Learning (DL), which are cutting-edge approaches that provide improved dependability, when producing and generating traffic flow predictions [7,8]. Using a range of techniques and methods, Traffic Congestion Prediction (TCP) aims to forecast future traffic patterns. The information provided by these forecasts is crucial for decision-makers in several industries, including business, government, utilities, and Smart Cities (SCs) [9]. There are many effective ways to forecast traffic congestion, and to get the best predictive performance, with most of them using either a DL model, such as a Recurrent Neural Network (RNN), or an ML model, such as a tree-based approach.

The complex field of TCP is investigated in this review, which explores the fundamental concepts, different algorithms, and innovative strategies that have been addressed in recent research. We investigate how methods from ML, DL, and statistics are applied to TCP. The efficacy of ensemble techniques in enhancing prediction accuracy and reliability is also investigated. A deeper understanding of the complexities involved in forecasting models' comparative analysis can be gained by examining the assessment metrics that assess their effectiveness.

TCP is crucial for supporting efficient traffic management and decision-making procedures. It gives SCs the necessary flexibility, enabling ITSs to efficiently coordinate and control future traffic demand. Moreover, it helps traffic management systems to forecast traffic trends precisely and get ready for increased traffic congestion [6].

This review offers insights into the latest advancements and state-of-the-art techniques, methodologies, and approaches that facilitate the field's further advancement, as we explore state-of-the-art TCP. Identifying gaps, challenges, and opportunities through a critical study of the corpus of existing literature lays the groundwork for future research.

The remainder of the manuscript is organized as follows. Section 2 provides information on the methodological instruments used in this review. Section 3.1 discusses the commonly used data parameters and time horizons for TCP. The techniques, models, and algorithms used for TCP are examined in Section 4 including statistical, ML, and DL models, while Section 5 showcases commonly used performance evaluation/validation metrics. Section 7 addresses the implications and challenges, and Section 8 discusses future directions of TCP. Lastly, Section 9 provides an overview of the work conducted, along with take-away lessons.

2. Materials and Methods

The methodologies employed in this research are designed to systematically explore the current state, challenges, and potential advancements of TCP technologies in ITSs. A combination of primary and secondary data from an extensive literature review of case studies is the basis of this work. This section outlines the data sources and methodological approach to create the research structure employed to achieve a comprehensive understanding of TCP's state of the art. We present here our methodological approach along with the structure of this research, starting with data collection.

2.1. Primary and Secondary Data Source Collection

This study used primary data from targeted case studies on TCP applications in SCs within ITS frameworks. The studies were selected as a result of their relevance to TCP architectures and potential to offer valuable insights on TCP implementation in ITSs. A thorough literature review that included peer-reviewed publications and industry reports was another method used to collect secondary data. Together, these data sources provided a solid foundation for understanding TCP technologies, their current uses in ITSs, and how they contribute to the efficiency of ITS processes.

2.2. Methodological Approach

This research employed a qualitative approach, utilizing thematic analysis to explore the operational possibilities and challenges of TCP. This method facilitated the identification of patterns, key trends, and the strengths and weaknesses of TCP structures in the context of ITSs. The literature review process, illustrated in Figure 1, was designed to ensure a comprehensive and systematic exploration of relevant studies. Over a hundred scholarly articles were reviewed, focusing on TCP's integration into ITS techniques, with priority given to studies that provided novel insights or substantial contributions to the existing knowledge base.

Table 1 presents an overview of the primary sources, detailing the specific databases, keywords, and search parameters used to compile the relevant literature. To ensure a rigorous selection process, we applied predefined inclusion and exclusion criteria, as detailed in Table 2. These criteria were established to filter studies based on relevance, methodological robustness, and contribution to TCP research. The search strategy involved querying multiple indexing databases, including Scopus, Web of Science, and IEEE Xplore, using carefully selected keywords and Boolean operators to refine the results.

Additionally, to enhance the reliability of our review, a citation analysis was employed to identify influential papers and assess the impact of specific studies. Yet, no software tools or AI-driven methods were utilized. This methodological framework allowed us to systematically integrate theoretical foundations with empirical case studies, ensuring a balanced and well-supported discussion of TCP's role in enhancing ITS efficiency in SCs.

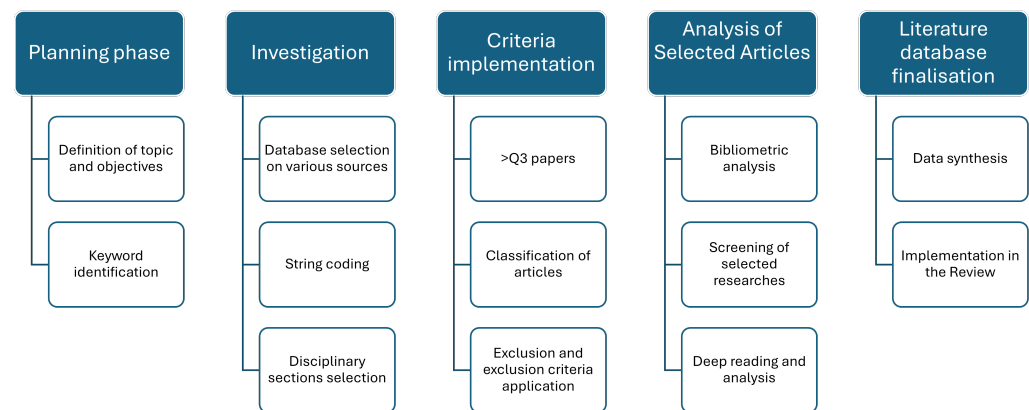


Figure 1. Literature review process.

Table 1. Literature collection methodology.

Criteria	Details
Sources	Springer, Elsevier, IEEE Xplore, MDPI, Wiley, Taylor & Francis, ResearchGate, Google Scholar
Search Strings	‘Traffic Congestion Prediction’, ‘Traffic Flow Prediction’, ‘Urban Traffic Prediction’, ‘Traffic Congestion Forecasting’, ‘Time Series Regression and Traffic Prediction’, ‘Time Series Classification and Traffic Prediction’, ‘Ensemble Models for Traffic Prediction’, ‘Deep Learning and Traffic Prediction’, ‘Machine Learning and Traffic Prediction’, ‘Statistical Models and Traffic Prediction’
Keywords	‘Prediction’, ‘Forecasting’, ‘Traffic Congestion’, ‘Traffic Demand’, ‘Traffic Load’, ‘Smart Cities’, ‘Traffic Management’, ‘Intelligent Transportation Systems’, ‘Statistical Models’, ‘Tree-Based’, ‘Deep Learning’, ‘Machine Learning’, ‘Time Series’, ‘Ensemble’, ‘Regression’, ‘Classification’

Table 2. Criteria for including and excluding literature.

Inclusion Criteria	Exclusion Criteria
Articles published between 1970 and 2024	Articles published before 1970
Publications from scholarly and reputable journals and conference papers	Opinion articles, introductory sections, abstracts, book critiques, and other informal publications
Works authored in English	Works authored in other languages
Research addressing the integration of TCP in SCs and ITSs	Research outside the specified scope of TCP in SCs and ITSs, or lacking strong evidence

2.3. Motivation and Comparison with Existing Review Papers

Recent technological advances in TCP have been extensively reviewed, particularly concerning the application of artificial intelligence [10], statistical [11], machine [12], and deep learning approaches [13]. These works, despite their thorough review, cover partial aspects of algorithmic TCP providing depth only to statistical [11], machine [12], and deep learning approaches [13,14], partial combinations of these [15] or generic artificial intelligence [10] to transportation systems. Our work illustrates the depth of all of these aspects combined.

It offers distinct advantages, such as a detailed comparative analysis of various ML models, including traditional algorithms and advanced neural networks for TCP, as well

as practical case studies. It demonstrates the application of different predictive models in diverse real-world urban pilots, and discusses the interpretability of complex ML models. Finally, our work is differentiated from other reviews by offering practical guidance on their implementation, as it provides a detailed step-by-step view of TCP implementation, including data collection and engineering, data preprocessing and TCP model selection, serving as a valuable tool for both researchers and practitioners.

2.4. Research Structure

This research progressed through discrete, yet interlinked phases:

1. Theoretical foundation: this step identified research gaps and constructed the theoretical basis through an initial examination of secondary data (Section 2).
2. Data collection and analysis: primary and secondary data on TCP applications were gathered and analyzed to inform the study, as discussed in Sections 2 and 3).
3. Evaluation of methods and case study analysis: A comprehensive examination of forecasting methods, including statistical, ML, and DL models, was conducted. This corresponds to Section 4, which presents case studies of model types to clarify real-world implementations of TCP in ITSs inside SCs, and Section 5, which presents their evaluation metrics.
4. Discussion, implications and challenges: the strengths, weaknesses, suitability and implications of these methods for ITS were analyzed, and challenges in implementation were identified, as elaborated in Sections 6 and 7.
5. Future directions and conclusions: the findings were synthesized, highlighting opportunities for future research and practical implementation (see Sections 8 and 9).

3. Traffic Congestion Prediction

By learning, evaluating, and predicting future changes in flow over a certain period of time, traffic managers can make better decisions on traffic control. By employing sophisticated forecasting models, the use of ITSs may aid in the development of particular traffic congestion analyses. The rapid advancement of technology has motivated a lot of studies focusing on data-driven approaches.

At the street level, there are different types of traffic loads; the most common presenting to multiple (high, medium, low) [16] or binary congestion levels [17], referring to classification problems, others on vehicles per hour/minute, etc. [18], or average vehicle speed [19] referring to regression problems. Depending on the characteristics of the dataset and the nature of the problem to be solved, all these traffic problems can be addressed either as time series or basic regression and classification problems.

3.1. Forecasting Horizons in Traffic Congestion Prediction

Traffic prediction forecasting horizons refer to the time intervals in the future for which traffic conditions are predicted. According to the literature, these horizons are typically categorized into Short-Term TCP (STTCP), Medium-Term TCP (MTTCP), and Long-Term TCP (LTTCP), each serving different purposes and employing various methodologies [20].

STTCP targets ranges from a few seconds up to 30 min ahead. It supports real-time traffic management, incident detection, dynamic route guidance, and traveler information systems. STTCP requires data with high frequency to capture rapid changes in traffic conditions, resulting in high temporal resolution. This results in emphasizing quick computation for real-time applicability [21,22].

MTTCP studies showcase time horizons ranging from 30 min up to several hours ahead, and their purpose is to aid in traffic control strategies, congestion management, and resource allocation. Compared to STTCP, MTTCP requires a balance between temporal resolution and computational efficiency, with the model implementation needing to account

for temporal patterns and possibly daily traffic cycles. In general, MTTCP utilizes hybrid models combining statistical methods with ML, spatiotemporal models, and some DL techniques [23,24].

LTTCP's time horizon ranges from several hours up to days or even weeks ahead. The purpose of LTTCP is to support infrastructure planning, policymaking, event planning, and long-term traffic management strategies. Characteristics of LTTCP include low temporal resolution with a focus on broader trends, rather than immediate fluctuations. Also, it often requires considering variables like seasonal trends, economic factors, and planned events. Usually, LTTCP implementations include time series analysis, statistical models, and ML models that can handle long-term dependencies [25,26]. The overall forecasting horizon classes are illustrated in Figure 2.

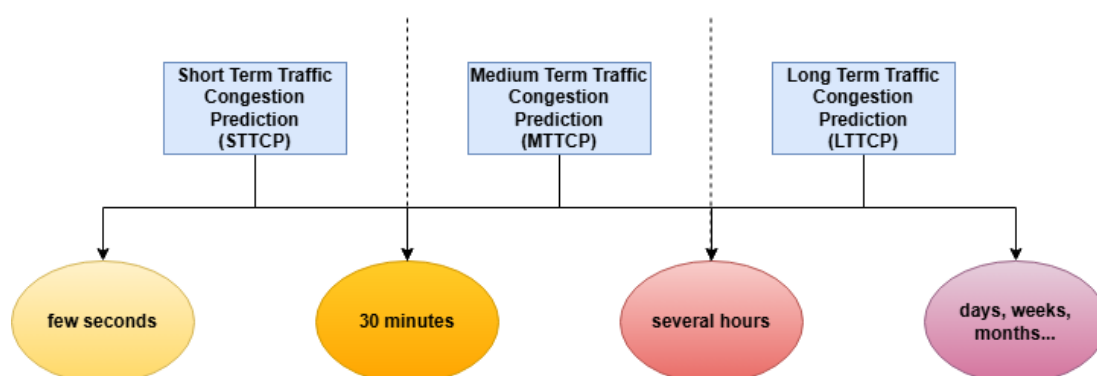


Figure 2. TCP horizons.

3.2. Input Parameters for Traffic Congestion Prediction

Three distinct input types are frequently utilized when using TCP, including seasonal input variables, historical traffic load, and weather parameters. Seasonal input variables, such as the month of the year, season, weekday, hour of day, etc., and historical traffic load data like hourly loads for the hour prior, the day prior, and the day prior to the previous week are examples. Weather parameters include air temperature, relative humidity, precipitation, wind speed, and cloud cover.

According to [27], there is a variety of parameters that affect traffic flow, including meteorological conditions. Their research outcomes illustrated that integrating external factors into forecasting algorithms somewhat enhanced predictive accuracy, whereas implementing modeling innovations, like vector and Bayesian estimation, significantly strengthened the models.

To enhance prediction accuracy with weather data, the researchers in [28] combined decision-level data fusion methods and deep belief networks (DBNs) for traffic and weather forecasting. The experimental findings, using traffic and weather data from California's San Francisco Bay Area, confirmed the effectiveness of the proposed approach.

Moreover, research on Tokyo's traffic congestion demonstrated a reduction in capacity between 4–7% during light precipitation and up to 14% under heavy rainfall [29]. Additionally, rainfall slows down free-flow speed as drivers need to adapt to slippery roads and diminished visibility. According to [30], adverse weather conditions reduce road capacity without affecting traffic demand. It would be interesting to examine additional weather elements, such as relative humidity or temperature, to enhance congestion management and make use of precise traffic congestion forecasting algorithms.

Furthermore, in the work of [17], the authors used seasonality, weather, street characteristics, and the traffic flow of nearby locations to solve a problem of filling missing traffic information, based on a classification approach. Results indicated that the information gained from the next and the two previous junctions, combined with weather data

like relative humidity, temperature, cloud cover, wind speed, and precipitation, along with the length of the road, maximum speed limit in kilometers per hour, direction, and street category (main, residential, highway, with or without strip) could significantly boost models' performance.

In the work of [31], the authors refer to two different traffic data sources: stationary and probe data. Stationary data include sensor data and fixed cameras, while probe data that were used in the studies were GPS data mounted on vehicles. Specific traffic parameters in the study included traffic volume, density, occupancy, speed, and congestion index.

Authors in [32] utilized a TCP model during holidays, incorporating a hybrid prediction methodology combining discrete Fourier transform (DFT) with Support Vector Regression (SVR). The suggested methodology illustrated a higher accuracy compared with the traditional methods, showcasing an efficient method for TCP during the holidays.

Other works also include other factors, like COVID-19 [33], vehicle trajectories and information [34], or location criteria [35]. A generic presentation of the input parameters regarding TCP is illustrated by categories in Table 3.

Table 3. TCP's essential input parameters summary.

Weather Information	Time Features-Seasonality	Street Information	Miscellaneous	Traffic Information
Temperature	Hour of day	Length of the road	COVID quarantine	Historical traffic load
Rainfall	Weekday	Flow direction	Vehicle trajectory	Traffic volume
Wind velocity	Minute of hour	Main, residential, highway	Holidays (yes/no)	Traffic density
Relative humidity	Day of month	With/without strip	Occupancy	Traffic occupancy
Clouds	Month	Maximum speed limit	Vehicle details	Traffic congestion index
Wind direction	Year	-	Location criteria	Nearby locations/junctions

3.3. TCP Process Flow Analysis

A standard generic flow for TCP problems usually comprises three main phases. These include (i) data collection and engineering, (ii) data preprocessing, and (iii) model selection for regression or classification for TCP, as illustrated in Figure 3.

3.3.1. Data Collection and Engineering

Regarding data collection and engineering, in most cases, TCP records are combined with datasets from the same period, including information like weather, seasonality, sensors, KPIs, or miscellaneous information [17]. For the integration of these datasets, it is crucial that they share the same temporal resolution (e.g., 15 min, 30 min, 1 h), which necessitates the use of normalization techniques [36]. In addition, a comprehensive Exploratory Data Analysis (EDA) [37] is essential to enhance the interplay between visual analytics and statistical summaries. To conduct these analyses and subsequent tasks, a variety of tools and technologies can be applied, including MATLAB [38], SQL [39], or Python with libraries such as scikit-learn [36], TensorFlow [40], PyTorch [41], Pandas [42], Numpy [43], and Matplotlib [44].

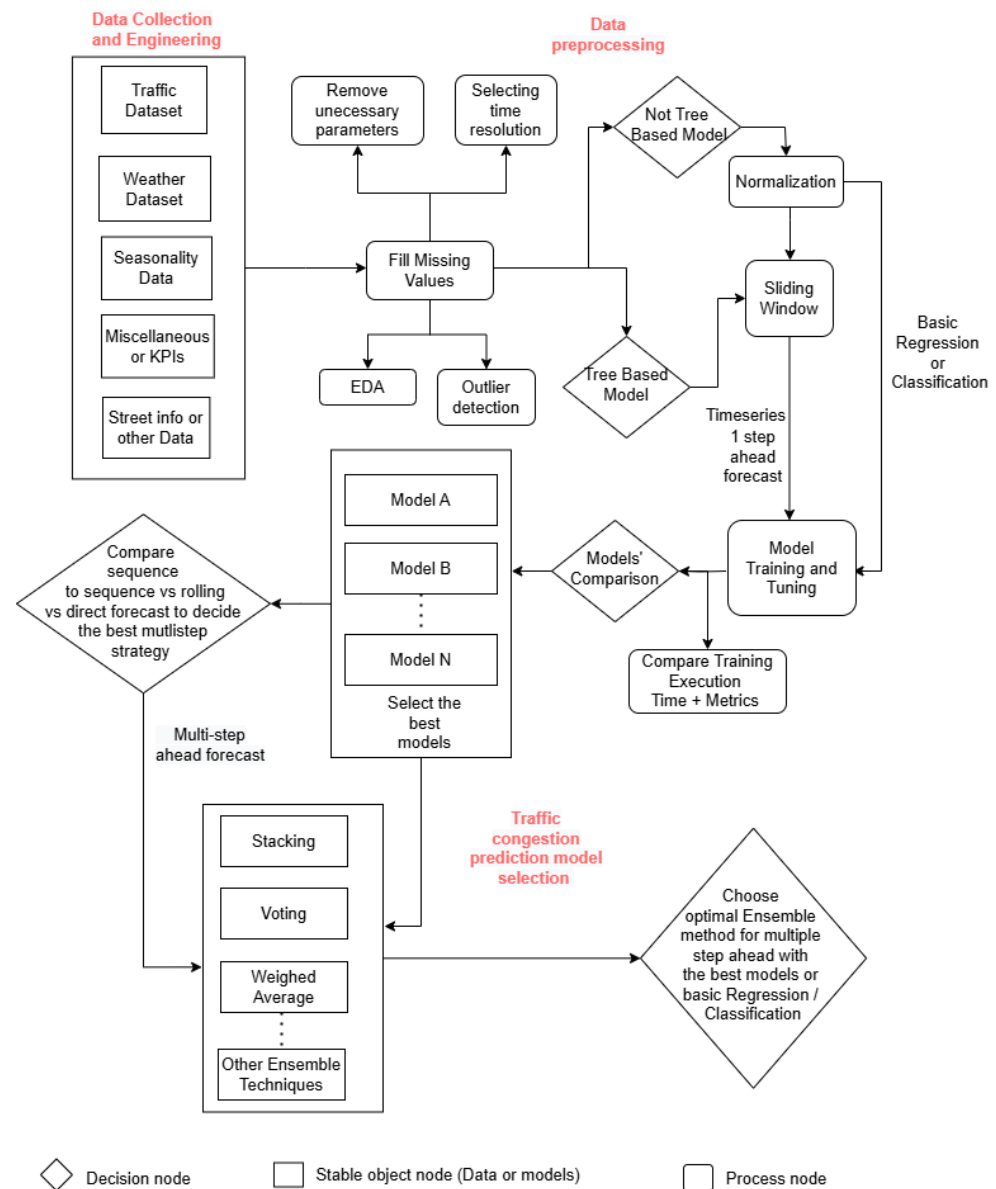


Figure 3. TCP generic step-by-step process flow of data collection, preprocessing, and prediction pipeline.

3.3.2. Data Preprocessing

Regarding data preprocessing, numerous previously mentioned steps, such as EDA, outlier detection/removal, and addressing missing values, can also be considered components of data collection and engineering [45]. Nevertheless, one can differentiate the eventually compiled dataset preceding these steps, distinctly presenting all values with their specific timestamps during the intermediary phase (post-EDA and pre-outlier detection/removal or filling of missing values). Subsequently, the process typically advances with a transformed dataset that is prepared for training, testing, and validation [17].

Other techniques supported EDA and involved in basic classification, regression, or time series forecasting, with substantial impact on the performance of both one-step-ahead [45] and multi-step-ahead [46] predictions, include Principal Component Analysis (PCA) [47], decomposition [48], feature selection [17], feature extraction [47], denoising [49], residual error modeling, outlier detection [45], and filter-based correction [50].

An EDA can uncover critical insights about the dataset, indicating whether operations like identifying and removing outliers or imputing missing values are warranted [51]. For missing information, several techniques have been explored. For instance, leveraging data

from nearby road segments to estimate missing values was proven effective [17]. Other implementations include models that can handle missing data by incorporating temporal and spatial correlations [52], imputation using CNNs [53], or graph-based approaches, which can handle missing data, by incorporating both temporal and spatial dependencies [54].

Moreover, an EDA provides valuable information on the methodology for feature selection. Tree-based models (TBMs) and non-tree-based models (non-TBMs) generally employ distinct methods post-missing data imputation, before applying time series regression techniques. TBMs often bypass normalization due to their inherent design. In contrast, non-TBMs may require normalization, typically achieved using the MinMaxScaler function from the Python sklearn.preprocessing package [36], thereby ensuring the transformed dataset adheres to a specific range.

Prior to conducting time series regression, the Sliding-Window (SW) technique [55] is applied. This is characterized by data shifted ahead by multiple steps (e.g., 96 steps for 24 h with a 15 min interval), with the resultant data becoming the input for the models. RNN-based models incorporate a three-dimensional SW comprising timesteps, rows, and parameters, while other models use a two-dimensional format encompassing rows and parameters plus timesteps. Subsequently, the dataset is commonly divided into test and training sets, often reserving 20% for testing and 80% for training, a common split based on the Pareto principle [56,57], though occasionally allocating an additional 5–10% for validation purposes [58].

3.3.3. TCP Model Selection and Next Steps

The next steps include the model selection and the forecasting implementation that are analyzed further in Section 4, depending on the problem's nature (classification, regression), data synthesis (time series or baseline prediction), and model type (statistical, ML, DL, or ensemble).

4. Forecasting Methods Based on Models and Algorithms

This section reports on statistical, ML, and DL algorithms and ensemble methods utilized for TCP problems as summarized in Figure 4.

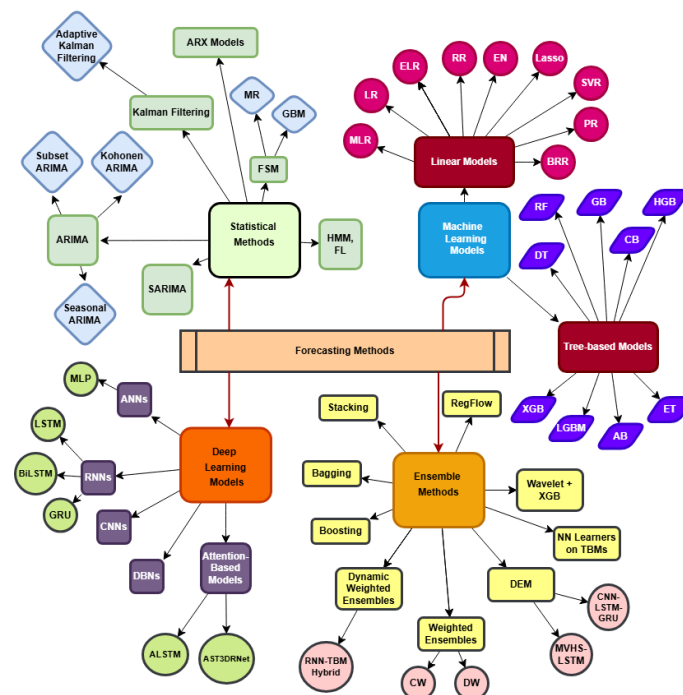


Figure 4. Summary of most-used prediction models and methods for TCP.

4.1. Statistical Models

The foundation of TCP was established using methods of statistical analysis. They provide resources for simulating and forecasting patterns of traffic flow. These techniques use past data to find seasonality, trends, and other significant trends in ITSs. A survey of their contributions to the subject is given in Table 4, which summarizes important statistical techniques, their uses in TCP, and noteworthy discoveries.

Predicting traffic flow has been a significant field of research since the 1970s. Early efforts in traffic congestion prediction primarily relied on statistical (or parametric) models, due to their simplicity and interpretability. These fixed-structure models used empirical data to train their parameters. One of the basic time series prediction techniques in statistics-based approaches is ARMA (Autoregressive Moving Average) [59], and its extended version, ARIMA (Autoregressive Integrated Moving Average) [60].

The ARIMA mode is considered as one of the most popular statistical parametric models and has also been integrated with time series models for STTCP [61]. The original ARIMA [62] or modified variants of the original model were employed in this category, including Subset ARIMA [63], Seasonal Auto-Regressive Integrated Moving Average (SARIMA) [64], and Kohonen ARIMA [65]. The mean and variance of the data must be consistent for these models to work, yet traffic data have nonlinear and stochastic features. Although Linear Regression (LR) and statistical models can perform well in normal situations, they are less adaptable when the external system changes and are limited in handling nonlinear patterns inherent in traffic data [23].

The SARIMA approach [66], which is an extension of the ARMA method, is particularly well suited for modeling seasonal, stochastic time series that are constantly present in traffic flow data. It is capable of detecting underlying correlations in time series data. The spatial influence in urban flow prediction problems cannot be represented by these traditional time series approaches, despite their ability to capture temporal dependencies in time series data.

Kalman filtering techniques were also employed for real-time traffic estimation and prediction [67]. While effective for STTCP, these methods struggled with the non-stationarity and volatility of traffic flows during peak hours or incidents.

The authors of [68], taking into account that the recent development of sensor technology connected to the Internet facilitated the possibility of developing forecasting models more effectively, proposed a method based on adaptive Kalman filters to diagnose the future flow of traffic, merging the data supplied by connected vehicles and Bluetooth devices. The model used a test bed of a scheme of more than 100 connected vehicles around an urban catchment area in Australia. The results showed that as vehicle volume increased, more data were generated, and therefore, the Kalman filter performed flow forecasting more effectively. The proposed method worked best with a significant margin, close to 11%.

The uses of Auto-Regressive Model with Exogenous Input (ARX) models in predicting actual traffic flow in New York City were examined in the study of [69]. The ARX models were built using Neural Networks (NNs) or linear/polynomial algorithms. Extensive comparative analyses were conducted based on the algorithms' accuracy, efficiency, and computational requirements for training.

The study of [70] evaluates the most popular traffic forecasting techniques and suggests a novel strategy that could fix issues with the estimation techniques currently used in the literature. As a result, the study contrasts Fourier Series Models (FSMs), Mean Reversion (MR), and Geometric Brownian Motion (GBM). Given the long-horizon nature of toll road concessions and uncertainty, the comparison assesses the best-fit patterns in traffic demand estimates. The models make use of an actual concession initiative's historical traffic demand from the Fourth-Generation Roads Concession Program (4G) in Colombia. The findings

show that when it comes to predicting seasonal traffic, the Fourier series performs better than GBM and MR.

The work of [71] suggested the use of a Hidden Markov Model (HMM), a stochastic method, to anticipate short-term freeway traffic during peak hours. The study data came from real-time traffic monitoring devices on a 60.8 km (38 mile) stretch of Interstate-4 in Orlando, Florida, over a six-year period. The HMM used first-order statistics (mean) and second-order statistics (contrast) of speed data to construct traffic states in a two-dimensional space. State transition probabilities were used to account for the dynamic changes in highway traffic circumstances. HMMs determined the most likely order of traffic states for a series of traffic speed observations. Prediction errors, which were quantified by the relative length of the distance between the observed state and the anticipated state in two dimensions, were used to assess the model's performance. HMMs produced reasonable prediction errors of less than or about 10%. Additionally, location, travel direction, and peak period time had no discernible effects on the model's performance. Two naïve prediction techniques were contrasted with the HMMs. The outcomes demonstrated that HMMs outperformed the naïve approaches in terms of performance and robustness.

Other studies, like [72,73], employed fuzzy logic and HMMs, respectively, to predict the rate of traffic congestion using traffic camera data, with accuracy scores ranging from 95% in [73] and from 75% to 88% in [72].

Table 4. Overview of TCP's statistical models.

Model/Method/Reference(s)	Applications	Findings
ARMA [59]	STTCP	Early application for STTCP with moderate accuracy in linear patterns.
ARIMA, Subset ARIMA, Seasonal ARIMA, Kohonen ARIMA [60–63,65]	STTCP	Demonstrated effectiveness in STTCP, yet struggled with nonlinear patterns.
SARIMA [64,66]	Seasonal traffic flow prediction	Captured seasonal and stochastic components but did not account for spatial dependencies.
Kalman filtering [67]	Real-time traffic estimation	Effective in STTCP but struggled with peak-hour volatility and non-stationarity.
Adaptive Kalman filtering [68]	Real-time traffic flow prediction using connected vehicles	Showed superior accuracy with increasing data from connected vehicles, improving by 11%.
ARX Models (NNs, linear/polynomial) [69]	Traffic flow prediction in urban environments	Comparative analysis revealed high accuracy in urban traffic predictions using ARX with NNs.
Fourier Series, Mean Reversion, GBM [70]	LTTCP	Fourier series outperformed GBM and MR in modeling seasonal traffic demand for toll roads.
Hidden Markov Model (HMM) [71]	Short-term freeway traffic prediction	Achieved robust predictions during peak hours with error margins less than 10%.
Fuzzy logic and HMM [72,73]	Traffic congestion prediction	Used traffic camera data to predict congestion rates, with accuracy ranging from 75% to 95%.

4.2. Machine Learning Approaches

Linear models and TBMs are fundamental for predicting traffic congestion. Linear models, like LR and its regularized forms, analyze relationships in data effectively, while TBMs capture complex patterns and handle nonlinear relationships. Tree methods, including Random Forests (RFs) and Gradient Boosting (GB), are especially suited for real-time traffic prediction due to their robustness and accuracy.

4.2.1. Linear Models

As the most fundamental and thoroughly studied regression technique, LR can be regarded as both a statistical model and an ML process. Simple LR is used when just one input variable is available, but multiple Linear Regression (MLR) is used when many variables are present, usually referred to as a matrix of features. Both LR and MLR have often been utilized in TCP problems [74].

Using an MLR model, a prediction method for STTCP in urban areas is proposed in [75]. The corresponding data attributes of short-term traffic flow in urban areas are chosen based on traffic operation status and used as the original data of traffic flow prediction. Based on the selected attributes, data on spatial static attributes and traffic flow dynamic attributes are gathered, and fault data are identified and fixed. The experimental results demonstrate that, in comparison to other methods, the average prediction accuracy of the proposed method is as high as 98.48%, and the prediction time is consistently less than 0.7 s.

In an LR model, predictions are made by adding the weighted sum of the input characteristics to the bias term, also known as the intercept term. There is no need for further complexity if the LR model correctly represents the facts. A cost function is commonly used to confirm the efficacy of LR. The goal is to reduce the discrepancy between the linear model's predictions and the training observations. For non-stationary time series with an irregular, periodic trend, an extended LR model with periodic components was proposed in [76]. This model was implemented using a sine function of various frequencies.

Ridge Regression (RR) is a regularized version of LR. Regularizing the LR model is an effective method to reduce the problem of overfitting. This is achieved by reducing the number of polynomial degrees and restricting the degree of freedom of the model, which makes overfitting more challenging. Another regularized form of LR is the least absolute shrinkage and selection operator regression, or lasso. Like RR [77], it adds a regularization component to the cost function.

SVR is based on a high-dimensional feature space that is created by changing the original variables and adding a term to the error function to penalize the complexity that occurs, depending on the implementation (i.e., using a linear kernel) [78]. The SVR-produced model only depends on a subset of the training data, since the cost function used to build the model rejects any training data that are significantly close to the model prediction [79].

Similarly, the study in [80] was carried out on the 1019.38-acre campus of Jawaharlal Nehru University (JNU), which is situated in New Delhi, India. This study took into account the previously thoroughly examined real-time vehicle traffic at JNU that was personally tracked, gathered, computed, and examined. It included digitized data from January 2013 for the campus's north entrance, which covered 31 days with 24 h each. SVR with linear kernel was used for TCP, because it provided global minima for training samples and exhibited superior generalization ability.

Other approaches involve Lasso Least Angle Regression [81] and Elastic Net (EN) regression [82]. Additionally, algorithms such as Polynomial Regression (PR) [83] and

Bayesian Ridge Regression (BRR) [84] may fall under this category, depending on their implementation and context.

For example, lasso-type techniques were suggested in the work of [85] as an alternative to standard least squares. Lasso methods carry out variable selection by adding an L1 penalty term, which may help to lower part of the variation in the threshold parameter estimation. Initially, the findings of the study that simulated two distinct underlying model architectures were covered. While the second was a self-exciting threshold autoregressive model, the first was a regression model with correlated predictors. Lastly, an application to urban traffic data compared the suggested lasso-type algorithms to traditional approaches.

Based on data from a country fact survey on traffic laws, an international questionnaire on traffic safety attitudes, and other statistical databases, the authors of [86] created an EN regression model to assess the variables that affected a person's traffic infractions and accidents. Initially, it was discovered that individual characteristics and attitudes on road safety had an impact on the experience of traffic violations and accidents, in addition to elements at the national level, and secondly, the variables chosen for country factors varied across traffic infractions and accidents, even if the same variables pertaining to personal characteristics and attitudes were chosen for both.

The linear models illustrated in this review are presented in Table 5.

Table 5. TCP ML Linear Models overview.

Model/Method/Reference(s)	Applications	Findings
LR, MLR [74]	General TCP problems	Applied LR and MLR for traffic congestion prediction.
MLR [75]	STTCP	Achieved 98.48% accuracy with under 0.7 s computation time.
Extended LR with periodic components [76]	Non-stationary time series	Improved modeling of non-stationary traffic trends using sine functions.
Ridge Regression (RR) [77]	Regularized traffic flow prediction	Reduced overfitting and improved generalization through regularization.
Support Vector Regression (SVR) [78,79]	Real-time traffic flow prediction	Utilized SVR with a linear kernel for high-dimensional traffic prediction.
SVR with linear kernel [80]	Real-time traffic prediction at JNU Campus	Achieved global minima for real-time traffic flow predictions.
Lasso Least Angle Regression [81]	Variable selection in linear models	Improved variable selection in high-dimensional data using LARS.
Elastic Net (EN) regression [82]	High-dimensional data analysis	Combined L1 and L2 penalties for improved performance with correlated predictors.
Polynomial Regression (PR) [83]	Nonlinear traffic flow prediction	Modeled nonlinear relationships in traffic flow.
Bayesian Ridge Regression (BRR) [84]	Regularized traffic prediction	Provided probabilistic traffic forecasting with regularization to prevent overfitting.
Lasso regression [85]	Threshold parameter estimation	Improved threshold estimation in urban traffic data.
EN regression [86]	Traffic violations and accidents prediction	Assessed factors affecting traffic violations and accidents.

4.2.2. Tree-based Models (TBMs)

Near zero values, TBMs typically behave better than other types of models [87–90]. Improved metric scores can be obtained using TBMs, since TCP datasets can offer distributions with a large number of zero values or zero-inflated (ZI) distributions [91]. Tree models are far superior to linear models because of their capacity to handle outliers. It was projected that regression trees would perform better than linear models when attributes do not show a linear relationship with the target variable [92]. These are solid algorithms that successfully fit complicated datasets.

Besides the basic Decision Tree (DT) [93], there are many popular Boosting/TBMs utilized for TCP. These include RF [94], Light Gradient Boosting Machine (LGBM) [95], Extreme Gradient Boosting (XGB) [96], GB [97], Histogram-Based Gradient Boosting (HGB) [97], Categorical Boosting (CB) [98], AdaBoost (AB) [99], Extra Trees (ETs) [100] and possibly others.

The authors of [101] used an ML approach that was more appropriate for the data structure and produced an accuracy of 91%. In addition to CDT data, GPS measurements may yield other reliable traffic information. In that study, a DT (J48/C4.5) classifier with mobile sensors was used to measure the amount of traffic congestion from GPS data and photos of traffic conditions. It achieved accuracy scores of 92%, enabling the surveillance of far greater traffic zones. The DT algorithm could detect patterns regarding vehicle movement. A fixed SW method was implemented.

To predict traffic conditions, the study of [102] used a dataset of more than 66,000 GTFS records, employing several TBM classifiers, such as RF, XGB, CB, and DT models, respectively. The inherent imbalance in the dataset was addressed by SMOTE, which guaranteed the increased representation of minority classes, and feature scaling improved model convergence. For this assignment, RF was the most accurate model, with a 98.8% accuracy rate. According to the results, the system could accurately predict traffic in real time, which helped with route planning, traffic management, and improving urban mobility.

A comparative study and methods for TCP were presented in the work of [16]. To choose the best method, a number of ML algorithms and techniques were compared. The approach was evaluated using data from one of the most troublesome streets in terms of traffic congestion in Thessaloniki, the second most populous city in Greece, employing Data Mining and Big Data approaches in addition to Python, SQL, and GIS technology. According to the evaluation and findings, the most important elements influencing algorithmic accuracy were data quantity and quality. A comparison of the results revealed that DT outperformed Logistic Regression in terms of accuracy.

The work of [103] proposes a TCP model that uses the XGB algorithm in conjunction with wavelet decomposition and reconstruction STTCP. First, the high- and low-frequency information of the target traffic flow is obtained during the training phase using the wavelet denoising technique. Second, the threshold approach is used to process high-frequency traffic flow data. The training label is then created by reconstituting the high- and low-frequency data. Lastly, the XGB algorithm is trained to predict traffic flow using the denoised target flow. This lessens the impact of short-term high-frequency noise, while maintaining the traffic flow pattern for each sample period. Based on traffic flow detector data gathered in Beijing, the suggested approach for TCP is evaluated and contrasted with the Support Vector Machine (SVM) algorithm. The outcome demonstrates that the suggested algorithm's prediction accuracy is significantly greater than SVM's, which is crucial for TCP.

Using a specially designed Android app to efficiently combine road and vehicle data, the study of [104] presents an improved, novel data fusion technique based on the safe route mapping methodology with the combined use of historical crash data and real-time

data, demonstrating significant improvements in real-time risk assessment. The enhanced safe route mapping framework closely observes road conditions and driver behavior. To evaluate overall driving competency, data gathered from drivers are evaluated on a central server using a facial recognition algorithm to identify indications of fatigue and diversions. Roadside cameras simultaneously record real-time traffic data, which are then processed by a sophisticated video analytics technique to monitor vehicle patterns and speeds. By combining various data streams, an LGBM prediction model is utilized, which helps drivers anticipate possible problems in the near future. Using a fuzzy logic model, predicted risk scores are combined with past crash data to define risk categories for various road segments. Using real-world data and a driving simulation, the enhanced safe route mapping model's performance is evaluated. It shows impressive accuracy, particularly when it comes to taking into consideration the real-time integration of traffic circumstances and driver behavior. Authorities can use the resulting visual risk heatmap to arrange trips intelligently based on current risk levels, find safer routes, and deploy law enforcement in a proactive manner. In addition to highlighting the value of real-time data for road safety, this study opens the door for data-driven, dynamic risk assessment algorithms that could lower road accidents.

The study of [105] used a set of multisource data, including land use, signal control, and roadway layout, to examine the effects of attributes on traffic order using Shapley additive explanation (SHAP) and a CB model. Application programming interfaces, field research, and navigation systems were used to gather traffic data for Beijing's intersection entrances. According to the model results, CB had an 81.1% F1 score, an 83.5% recall, and an 83.5% prediction accuracy. Additionally, SHAP was used to examine the significance, overall effects, main effects, and interaction effects of the influence factors. It was discovered that traffic order was significantly impacted negatively by the congestion index (CI). It was discovered that more electronic traffic management and additional lanes improved traffic order. Traffic order was improved via off-peak intersection entries or intersection entrances with three-phase signals. Furthermore, when the CI was between 1.1 and 1.4, a high green ratio for through vehicles could lessen the beneficial effect of the CI on traffic order. A signal management scheme with a high left-turn green ratio would produce a traffic flow that was both safe and orderly.

The ET ensemble is a forest of incredibly random trees. When creating a tree in a random field, only a random subset of the characteristics are taken into account for splitting at each node [100]. It is possible to construct trees with even more random entries for each attribute, in contrast to conventional DTs that look for ideal thresholds. This method trades a larger bias for a smaller variance. As a result, ETs train considerably faster than conventional RFs. This is due to the fact that it takes very little time to determine the optimal threshold for every characteristic at every tree node.

An overall overview of TBMs is presented in Table 6.

Table 6. Overview of TCP's ML tree-based models.

Model/Method/Reference(s)	Applications	Findings
Tree-based models (TBMs) [87–91]	Zero-inflated TCP data	TBMs improved performance in handling zero values and outliers.
Regression trees [92]	Nonlinear TCP relationships	Outperformed linear models for nonlinear attributes.
Decision Tree (DT) [93]	General TCP problems	Effectively handled nonlinear relationships and outliers.

Table 6. Cont.

Model/Method/Reference(s)	Applications	Findings
Random Forest (RF) [94]	Zero-Inflated TCP Data	Improved accuracy in zero-inflated datasets.
Light Gradient Boosting Machine (LGBM) [95]	Real-time risk assessment	Enhanced real-time risk assessment.
Extreme Gradient Boosting (XGB) [96]	STTCP	Achieved high accuracy in complex traffic patterns.
Gradient Boosting (GB), Histogram-Based Gradient Boosting (HGB) [97]	Traffic prediction	Efficient for large datasets with complex relationships.
Categorical Boosting (CB) [98]	Traffic order analysis	Achieved 83.5% prediction accuracy for traffic order.
AdaBoost (AB) [99]	Traffic flow prediction	Improved robustness by reducing bias.
Extra Trees (ETs) [100]	High-dimensional TCP data	Faster training for complex datasets.
RF, XGB, CB, DT [102]	Traffic condition prediction	RF achieved 98.8% accuracy in real-time predictions.
DT, Logistic Regression [16]	Comparative study for TCP	DT outperformed Logistic Regression.
XGB with wavelet denoising [103]	STTCP	Outperformed SVM with wavelet denoising.
LGBM with data fusion [104]	Real-time risk assessment	Enhanced risk assessment with real-time traffic data.
CB with SHAP analysis [105]	Traffic order examination	Achieved 83.5% accuracy; SHAP revealed key factors.

4.3. Deep Learning Frameworks

With their ability to capture intricate nonlinear correlations in huge datasets, DL regression models are effective tools for TCP. They represent the state of the art in predicting accuracy and model sophistication and vary from Convolutional Neural Networks (CNNs) for spatial data interpretation to RNNs for sequential data analysis or even Multi-Layer Perceptron (MLP) for basic regression and classification tasks. A summary of the literature presented in this section can be found in Table 7.

There are several DL architectures capable of TCP. These include Artificial Neural Networks (ANNs) like Feedforward Neural Networks (FNNs), also called MLPs [106], RNN architectures [107] like Long Short-Term Memory (LSTM) [108], Gated Recurrent Units (GRUs) [109] or bidirectional RNNs [107]. Also, there are CNNs [110], DBNs [111], and Radial Basis Function Networks (RBFNs) [112], among others.

Starting with earlier works, in the study [113], ANNs were used to analyze video data of traffic congestion from the driver's perspective in conjunction with cellular data and Cell Dwell Time (CDT), which is the time a cell phone connects to a mobile telecommunications antenna and provides an estimated travel speed.

In more recent studies, a multi-step prediction model based on a CNN and bidirectional LSTM (biLSTM) model was proposed by [114] to address a classic ITS problem: the use of dynamic traffic statistics to accurately estimate traffic flow due to the exponential growth of traffic data. The biLSTM model used the geographic properties of the traffic data as input to extract the time series characteristics of the traffic. The experimental results

verified that the biLSTM model improved prediction accuracy in comparison to the GRU and SVR models.

The study of [17] examined various ML algorithms to detect which ones were best suited for predicting and filling in the present traffic congestion values for road segments with inadequate or missing historical data for timestamps with incomplete information. The methodology was subsequently validated over a second time period after being assessed on a number of open-source datasets from one of the busiest streets in Thessaloniki, Greece, with reference to traffic. Comparing the results of experiments with different scenarios showed that using road segments close to those with incomplete data, along with an MLP, made it more effective to accurately fill in the missing information. The results showed that importing weather characteristics and addressing data imbalance concerns improved algorithmic performance for nearly all classifiers, with the MLP being the most accurate.

The paper of [115] proposed a method for TCP based on LSTM that corrected for missing temporal and geographical values. The suggested prediction approach used preprocessing before generating predictions. This included correcting temporal and spatial values using temporal and spatial trends and pattern data, as well as removing outliers using the median absolute deviation for traffic data. Data having time-series aspects had not been properly learned in earlier research. The suggested prediction method employed an LSTM model for time-series data learning in order to solve this issue. The Mean Absolute Percentage Error (MAPE) was computed for comparison with other models in order to assess the efficacy of the suggested approach. At about 5%, the MAPE of the suggested approach was determined to be the best among the models that were compared.

Following training on digital tachograph data, the suggested method of [116] provided highway speed predictions using the GRU model. In a single month, the digital tachograph data were collected, yielding over 300 million records. Vehicle locations and speeds on the roadway were among these data. According to experimental results, the GRU-based DL method performed better in terms of prediction accuracy than the state-of-the-art alternatives, the LSTM model, and the ARIMA model. Furthermore, compared to the LSTM, the GRU model had a reduced computational cost. ITSs and TCP can both benefit from the suggested approach.

The authors of [117] developed a DBN algorithm-based model for TCP. The target road segment was gathered and preprocessed, together with its historical traffic flow data in Tianjin. The DBN was then trained as a generative model by stacking multiple Restricted Boltzmann Machines. Lastly, the simulation experiment analyzed its performance. The suggested algorithm model was contrasted with several DL architectures like CNN, and Neuro Fuzzy C-Means models. According to the findings, the suggested algorithm model's Root-Mean-Square Error (RMSE), Mean Absolute Error (MAE), and MAPE were 4.42%, 6.21%, and 8.03%, respectively. It had a substantially higher prediction accuracy than the other three models.

In the research of [118], a biLSTM method is used for TCP in order to alleviate the escalating traffic congestion and lower traffic strain. Based on the gathered road traffic flow data, a biLSTM-based urban road short-term traffic state algorithm network is first created. Next, the network's internal memory unit structure is improved. It develops into a top-notch prediction model following training and optimization. After that, the prediction performance is assessed, and the experimental simulation is verified. Lastly, the real data, the data predicted by the LSTM algorithm model, and the data projected by the biLSTM algorithm model are compared. The simulation comparison demonstrates that while the LSTM and biLSTM forecast findings agree with the actual traffic flow pattern, the LSTM data significantly differ from the actual scenario, and the mistake is especially pronounced during peak hours. Although biLSTM differs significantly from the real situation during

peak times, it can still be used as a reference because it aligns well with the real situation during stationary periods and low-peak phases.

The authors of [119] propose (i) a cost-effective and efficient citywide data acquisition scheme by capturing a traffic congestion map snapshot from the Seoul Transportation Operation and Information Service, an open-source online web service, and (ii) a hybrid NN architecture that combines LSTM, Transpose CNN, and CNN to extract spatial and temporal information from the input image and predict the network-wide congestion level. Their test demonstrates that the suggested model is capable of learning temporal and geographical correlations for traffic congestion prediction in an efficient and effective manner. The suggested model performs better in terms of prediction performance and computational economy than two other DNNs (Auto-encoder and ConvLSTM).

Other recent DL methods include Attention-Based NNs that have been effectively applied to TCP problems, enhancing the modeling of complex spatiotemporal dependencies. In the work of [120], the authors introduce an Attention-Based LSTM model designed for STTCP. The proposed model captures features at different time intervals, leveraging time-aware traffic data to improve prediction performance. Similarly, the authors of [121] propose an Attention-Based Spatio-Temporal 3D Residual NN (AST3DRNet) for TCP. The AST3DRNet model integrates 3D residual networks and self-attention mechanisms to forecast traffic congestion levels. This approach, by stacking 3D residual units and utilizing 3D convolution, effectively captures spatiotemporal relationships. The incorporation of self-attention mechanisms improve the model's capacity to concentrate on important features across spatial and temporal dimensions, resulting in enhanced prediction performance.

Table 7. Overview of TCP's Deep Learning models.

Model/Method/Reference(s)	Applications	Findings
Multi-Layer Perceptron (MLP) [17,106]	Filling missing data for traffic congestion	Accurately filled missing traffic data; weather attributes and addressing data imbalance improved performance.
Artificial Neural Networks (ANNs) [113]	Real-time traffic speed estimation	Used video data and Cell Dwell Time (CDT) for driver-perspective travel speed estimation.
Long Short-Term Memory (LSTM) [115]	Temporal and spatial data prediction	Corrected missing temporal and spatial values with preprocessing; achieved a MAPE of 5%, outperforming compared models.
BiLSTM + CNN hybrid model [114]	Dynamic traffic statistics prediction	Combined CNN and biLSTM to model temporal and geographic traffic data; improved accuracy over GRU and SVR.
Gated Recurrent Units (GRUs) [116]	Highway speed prediction	Outperformed LSTM and ARIMA in accuracy with reduced computational cost using digital tachograph data.
Deep Belief Network (DBN) [117]	Traffic flow prediction	Achieved higher prediction accuracy compared to CNN and Neuro Fuzzy C-Means with an RMSE of 4.42%.

Table 7. Cont.

Model/Method/Reference(s)	Applications	Findings
BiLSTM [118]	Urban traffic flow prediction	Better than LSTM for stationary periods; challenges during peak hours with some alignment to real traffic patterns.
Hybrid NN (LSTM + CNN) [119]	Citywide traffic congestion prediction	Combined spatial and temporal data effectively; outperformed DNNs and ConvLSTM.
Attention-Based LSTM [120]	STTCP	Improved modeling of spatiotemporal dependencies, leveraging time-aware traffic data.
AST3DRNet [121]	Spatio-temporal traffic prediction	Enhanced spatiotemporal feature extraction using 3D residual networks and self-attention mechanisms.

4.4. Ensemble Strategies

In general, the literature includes numerous publications on ensembles in various fields, including TCP. Starting with weighted ensembles, these techniques are categorized into constant and dynamic weighting approaches. Researchers of [122] first introduced the idea of utilizing multiple models within a single procedure, leading to the development of ensemble learning. Since then, a variety of strategies has been suggested, including stacking and voting methods, which use automatic weights for individual models [123,124], as well as bagging and boosting techniques [125]. Other ensembles include ensemble models where NN learners train on the output of TBMs [126] for TCP or dynamically weighted ensembles utilizing a combination of RNNs and TBMs [127]. Apart from the conventional time series techniques, ZI regression techniques can also be applied to datasets that contain a significant quantity of zero data in their target parameter [128,129].

A prediction approach based on the combination of MLR and LSTM (MLR-LSTM) is proposed in the work of [130]. It makes use of the incomplete traffic flow data from the previous period of the target prediction section, as well as continuous and complete traffic flow data from each nearby section. The objective is to quickly and collaboratively forecast changes in the traffic flow in the target section.

Additionally, the authors of [103] provide a TCP model that combines the XGB approach with a wavelet decomposition and reconstruction for STTCP. During the training phase, high-frequency and low-frequency data on the target traffic flow are first collected using the wavelet denoising technique. Then, the high-frequency traffic flow data are processed using the threshold method. The high-frequency and low-frequency data are then reconstituted to form the training label. Finally, the denoised target flow is fed into the XGB algorithm, which then utilizes it to train its TCP model.

The authors of [131] forecasted the taxi traffic resulting from the number of tourists who would visit Beijing Capital International Airport using a variety of RNN-LSTM architectures. According to the study's findings, an LSTM-RNN prediction approach for tourist visits was constructed using three architectures. The three LSTM models that were employed were sequence-to-sequence (seq2seq) multi-step-ahead LSTM, a basic LSTM regression, and an LSTM network utilizing SW. Their conclusion was that different models provided the highest results for simultaneous training and testing, depending on the situation. The highest training results for predicting tourist visits came from regression models, with the lowest RMSE. On the contrary, during the testing phase, the SW model produced the lowest RMSE value.

In the work of [132], a TCP model using ensembling was developed to solve the complexity of multi-step traffic speed prediction, given the tight link between traffic speeds and traffic congestion. Detrending, which divides the dataset into mean trends and residuals, and direct forecasting, which reduces cumulative prediction errors, were the two main methodologies that the model incorporated. According to the study, the ensemble-based model performed better than other models like SVM and CB.

In the study of [133], the authors presented a probabilistic framework that was both versatile and robust, allowing for the modeling of future predictions with almost no limitations on the underlying probability distribution or modality. They used a hypernetwork architecture and trained a continuous normalizing flow model to achieve this. The resultant technique, known as RegFlow, outperformed rival methods by a large margin and produced state-of-the-art results on many benchmark datasets.

The authors of [134] offered a unique Deep Ensemble Model (DEM) with an emphasis on LTTCP. To construct this ensemble model, they first created the basic learners, which were a CNN, an LSTM network, and a Gated Recurrent Unit (GRU) network, as DL models. The outputs of various models were then combined based on each model's forecasting success in the following step. To assess each model's performance, the authors employed a different DL model. Their suggested ensemble prediction model was adaptable and could be modified in response to traffic data. They used a publicly accessible dataset to assess the suggested model's performance. The created DEM model achieved a mean square error of 0.25 and an MAE of 0.32 for multi-step prediction, according to experimental data, whereas the mean square error and MAE for single-step prediction were 0.06 and 0.15, respectively. The suggested model was compared to numerous models in various categories, including traditional ML models like k-nearest-neighbors regression, DT regression, LR, and other ensemble models like RF regression, as well as individual DL models like LSTM, CNN, and GRU.

Multiple Variables Heuristic Selection LSTM (MVHS-LSTM) is a unique prediction architecture that is presented in the research of [135]. Its capacity to choose informative parameters and remove extraneous elements to lower computational expenses while striking a balance between prediction performance and processing efficiency is the main novelty. The Ordinary Least Squares (OLS) approach is used by the MVHS-LSTM model to optimize cost efficiency and eliminate factors in an intelligent manner. Furthermore, it uses a heuristic iteration approach that involves epoch, learning rate, and window length to dynamically choose hyperparameters, guaranteeing flexibility and increased accuracy. Using actual traffic flow data from Shanghai, extensive simulations were run to assess the improved performance of MVHS-LSTM. Comparing the outcomes with those of the ARIMA, SVM, and PSO-LSTM models showed the potential and benefits of the suggested approach.

An overview of TCP's ensemble strategies presented in this section is illustrated in Table 8.

Table 8. Overview of TCP's ensemble strategies.

Model/Method/Reference(s)	Applications	Findings
Weighted ensembles, stacking, voting, bagging, and boosting [122–124]	General ensemble learning	Introduced multiple model integration, including stacking, voting, bagging, and boosting for TCP.
NN Learners on TBM outputs [126]	TCP on sparse data	NN learners trained on tree-based model outputs for improved performance on sparse datasets.
Dynamic weighted ensembles [127]	RNN-TBM Hybrid Prediction	Combined RNNs and TBMs dynamically for enhanced prediction accuracy and adaptability.

Table 8. Cont.

Model/Method/Reference(s)	Applications	Findings
Zero-inflated regression techniques [128,129]	Data with zero targets	Addressed datasets with significant zero data for robust traffic predictions.
MLR-LSTM hybrid [130]	Traffic flow prediction	Combined incomplete traffic data with continuous data from nearby sections for collaborative forecasting.
Wavelet + XGB [103]	STTCP	Used wavelet decomposition and XGB for denoised traffic flow predictions; improved accuracy.
RNN-LSTM architectures [131]	Tourist traffic prediction	Compared seq2seq, regression, and SW models; found SW model produced lowest RMSE during testing.
Ensemble for multi-step prediction [132]	Multi-step traffic speed prediction	Detrending and direct forecasting reduced cumulative errors; outperformed SVM and CB.
RegFlow [133]	Probabilistic TCP	Hypernetwork-based probabilistic framework achieved state-of-the-art results on benchmark datasets.
Deep Ensemble Model (DEM) [134]	LTTCP	Combined CNN, LSTM, GRU; achieved an MAE of 0.32 for multi-step and 0.15 for single-step predictions.
MVHS-LSTM [135]	Variable selection and forecasting	Used OLS and heuristic iteration for parameter optimization; outperformed ARIMA, SVM, and PSO-LSTM.

5. Metrics

There are several evaluation criteria for evaluating TCP performance. Depending on the nature of the problem (regression or classification), different evaluation metrics can be applied. In this subsection, the most popular metrics are provided in terms of classification and regression. These are outlined in Figure 5.

The selection of evaluation metrics is vital in determining the effectiveness of predictive models in TCP, impacting the interpretation and comparison of results. Although metrics like the MAE and RMSE are commonly utilized for regression tasks (Section 5.1), their usage and consequences differ, based on the situation. The MAE offers a simple metric for the average absolute error, causing it to be less affected by significant deviations, while the RMSE imposes stricter penalties on larger errors, which can be advantageous when emphasizing models that reduce extreme outliers. Moreover, metrics like the R^2 score provide information on the fraction of variance elucidated by a model. However, they may occasionally be deceptive when misapplied, especially in scenarios involving nonlinear correlations.

In addition to these standard measures, researchers frequently enhance them with specialized metrics, like the ones presented in Section 5.2), to more accurately reflect the subtleties of specific applications. In the analyzed studies, the choice of metrics frequently aligns with the particular demands of the task, underscoring the necessity for thoughtful interpretation instead of depending only on traditional statistical measures. This highlights the significance of not just presenting standard evaluation metrics, but also comprehending their advantages and drawbacks within the larger framework of model evaluation.

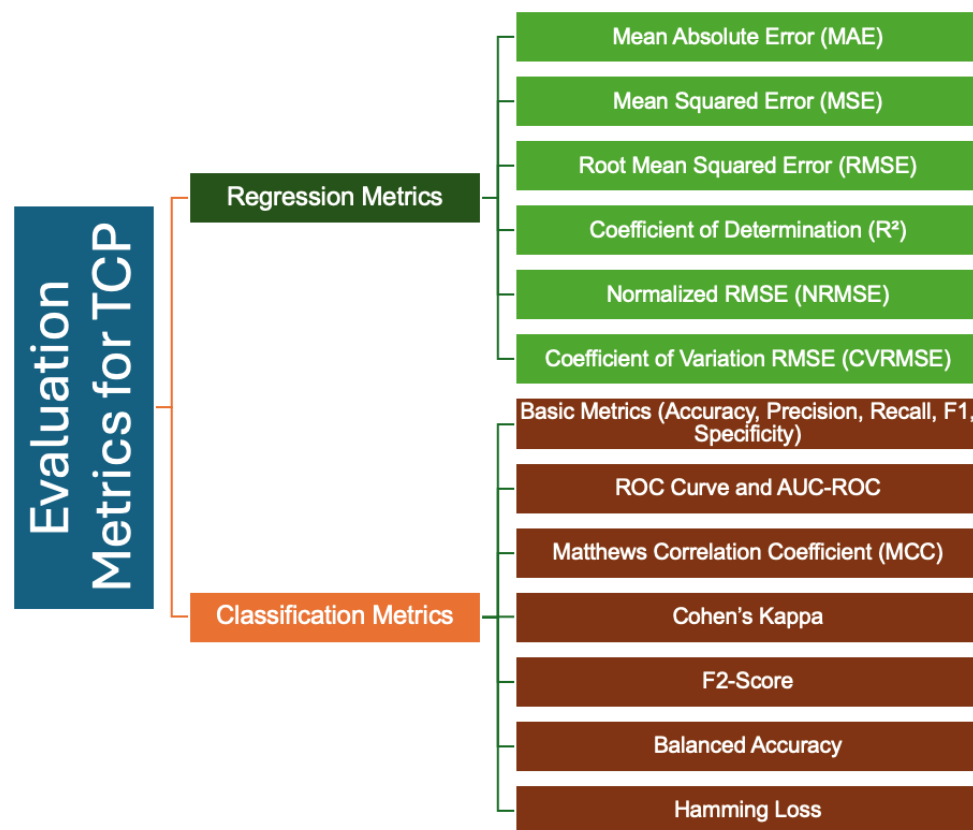


Figure 5. TCP evaluation metrics.

5.1. Regression

The most common regression metrics are the following:

5.1.1. Mean Absolute Error (MAE)

Calculating the MAE is fairly straightforward (Equation (1)), as it requires summing the absolute differences between observed and predicted values, then dividing this sum by the number of observations. Unlike other statistical techniques, the MAE assigns equal importance to all errors. The MAE is considered an absolute metric as it is counted in units and not in percentage.

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - f_i| \quad (1)$$

where y_i is the actual and f_i is the predicted value for the target, and N is the number of values.

5.1.2. Mean Squared Error (MSE)

The MSE assesses the quality of a fit by measuring the squared differences between each observed value i and its model prediction, followed by computing the average of these squared errors (see Equation (2)). Squaring not only eliminates negative values but also accentuates larger discrepancies. Clearly, a smaller MSE corresponds to a more accurate prediction. The MSE is considered an absolute metric.

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - f_i)^2 \quad (2)$$

where y_i is the actual and f_i is the predicted value for the target, and N is the number of values.

5.1.3. Root-Mean-Square Error (RMSE)

The RMSE is the square root of the mean of the squared differences between the observed and predicted values, essentially the square root of the MSE. Although the RMSE and MSE have nearly identical formulas (Equations (2) and (3)), the RMSE is preferred, because it is expressed in the same units as the dependent variable. The RMSE emphasizes larger errors more heavily, as the significance of each error on the overall measure is related to its square rather than its absolute value. Similarly to the MAE and MSE, the RMSE is also an absolute metric.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - f_i)^2} \quad (3)$$

5.1.4. Coefficient of Determination (R^2)

R^2 represents a ratio comparing the variance of prediction errors to the total variance of the data being analyzed. It quantifies the proportion of data variance “explained” by the predictive model. Unlike other metrics based on error, a higher R^2 value corresponds to a better model fit. It is computed as shown in Equation (4):

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} = 1 - \frac{\sum (y_i - f_i)^2}{\sum (y_i - \bar{y})^2} \quad (4)$$

where SS_{res} is the overall squares of residuals (errors), and SS_{tot} is the total overall of squares (proportional to the variance of the data), y_i is the real target value, $\bar{y}$ is the mean of the actual values, and f_i is the predicted value for the target. R^2 is considered a relative metric.

5.1.5. Normalized Root-Mean-Square Error (NRMSE)

The NRMSE evaluates the precision of a forecasting model by contrasting predicted values with observed ones. As a normalized version of the RMSE, it provides a comparative analysis of the model’s performance (Equation (5)).

$$NRMSE = \frac{RMSE}{y_i(max) - y_i(min)} \quad (5)$$

where $y_i(max)$ is the maximum observed value in the actual data, and $y_i(min)$ is the minimum observed value in the actual data. A lower NRMSE signifies a more accurate model alignment with the real data. As the NRMSE is considered a relative metric, it is commonly represented as a percentage by multiplying the outcome by 100.

5.1.6. Coefficient of Variation of the Root-Mean-Square Error (CVRMSE)

To account for the mean value of the observations, besides the NRMSE, the RMSE can be normalized to establish a more informative metric with the Coefficient of Variation of the CVRMSE. This transformation allows the total error to be represented as a percentage (Equation (6)). The mathematical formulations of both RMSE and CVRMSE inherently prioritize larger errors, since an individual error’s contribution is proportional to the square of its magnitude rather than the magnitude itself.

$$CVRMSE(\%) = \frac{RMSE}{\bar{y}} = \frac{\sqrt{\sum (y_i - f_i)^2 / N}}{\sum y_i / N} \quad (6)$$

where y_i is the real target value, f_i is the predicted value for the target, and N is the number of values. One significant advantage of the CVRMSE is the fact that it is a dimensionless

indicator, and therefore, it can facilitate cross-study comparisons because it filters out the scale effect. Similar to the NRMSE, the CVRMSE is considered a relative metric.

5.2. Classification

Regarding TCP classification, some of the common and advanced metrics are the following:

5.2.1. Basic Metrics

In general, there are commonly used metrics like accuracy, precision, recall, F1 score, and specificity [136] that provide valuable information regarding the performance of the models. Furthermore, some advanced metrics can be also used depending on the nature of the problem.

5.2.2. ROC Curve and Area Under the Curve (AUC-ROC)

A Receiver Operating Characteristic (ROC) curve is a graphical representation that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. Furthermore, the Area Under the ROC Curve (AUC-ROC) quantifies the overall ability of the model to discriminate between positive and negative classes, with values ranging from 0.5 (no discrimination) to 1 (perfect discrimination). The formula is as follows:

$$AUC = \int_0^1 TPR(FPR), d(FPR) \quad (7)$$

with TPR representing the True Positive Rate, that is, the proportion of actual positives correctly identified, and FPR representing the False Positive Rate, that is, the proportion of actual negatives incorrectly identified as positives.

5.2.3. Matthews Correlation Coefficient (MCC)

The Matthews Correlation Coefficient (MCC) is a measure regarding the quality of binary classifications, considering true and false positives and negatives. It returns a value between -1 and $+1$, where $+1$ indicates perfect prediction, 0 no better than random prediction, and -1 indicates total disagreement between prediction and observation.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (8)$$

where TP (true positive) represents the correctly predicted positive instances, TN (true negative) the correctly predicted negative instances, FP (false positive) the negative instances incorrectly predicted as positive, and FN (false negative) the positive instances incorrectly predicted as negative.

5.2.4. Cohen's Kappa

Cohen's Kappa is a statistic that measures inter-rater agreement for categorical items, adjusting for agreement occurring by chance. It is especially useful in evaluating the reliability of classifications.

$$\kappa = \frac{p_o - p_e}{1 - p_e} \quad (9)$$

where p_o (Observed Agreement) represents the proportion of instances where raters agree, and p_e (Expected Agreement) represents the proportion of instances where agreement is expected by chance.

5.2.5. F2-Score

The F2-score is a variant of the F-measure that places more emphasis on recall than precision. It is particularly useful when the cost of false negatives is higher than that of false positives.

$$F_2 = \frac{5 \times \text{Precision} \times \text{Recall}}{4 \times \text{Precision} + \text{Recall}} \quad (10)$$

5.2.6. Balanced Accuracy

Balanced accuracy is the average of recall or sensitivity (TP rate) and specificity (TN rate). It is particularly useful for datasets with imbalanced classes.

$$\text{Balanced Accuracy} = \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \quad (11)$$

5.2.7. Hamming Loss

The Hamming loss is the fraction of labels that are incorrectly predicted. In multi-label classification, it is the fraction of the wrong labels to the total number of labels.

$$\text{Hamming Loss} = \frac{1}{n} \sum_{i=1}^n \frac{|y_i \Delta \hat{y}_i|}{|L|} \quad (12)$$

where n is the total number of samples, y_i the true label set for the i th sample, $\hat{y}_i$ is the predicted label set for the i th sample, and Δ is the symmetric difference between sets.

6. Discussion

Effective TCP is crucial for effective traffic management and urban planning. In this work, various predictive models and methods were explored, each with distinct methodologies and applications. This section provides a discussion and comparative analysis of the presented TCP models, evaluating their strengths, limitations, and suitability for different traffic scenarios.

Starting with the statistical models, Section 4.1 illustrated that they could be effective for linear and stationary time series data, with interpretable parameters, but they struggled with nonlinear patterns, and they were limited in capturing nonlinear patterns, with degrading performance on complex traffic dynamics. These models are more suitable for STTCP in stable traffic conditions or for LT TCP problems [60–67].

Regarding the linear ML models, Section 4.2.1 indicated that in the case of LR models, one could rely on their simplicity, ease of implementation, and interoperability; however, they were unable to fit nonlinear relationships, and they were sensitive to outliers. Similarly to statistical models, LR models are suitable for scenarios with linear relationships between variables in STTCP cases [74–76]. Other linear ML models, like SVM [78–80], EN [86] or lasso [85] are more robust to high-dimensional data and effective in capturing nonlinear relationships with appropriate kernel functions; however, they are computationally intensive and require careful parameter tuning. These models in general are suitable with complex, but not excessively large, datasets.

In general, TBMs are simple to understand and interpret; they can handle both numerical and categorical data as well as large datasets and perform better around zero values. On the contrary, they are more sensitive to overfitting, unstable with small variations in data and occasionally require more computational resources, depending on the case. In general, TBMs can perform well in all traffic forecasting horizons, but they are more suitable for complex traffic systems with heterogeneous data sources and scenarios requiring interpretability [87–94,96,97].

DL models excel in capturing spatial and temporal dependencies, they are effective with grid-based traffic data, they can handle sequential data effectively (for the case of RNN models), and in general, they can perform better compared to the aforementioned model categories. On the other hand, they usually require large datasets to perform well, are less effective for temporal patterns, sensitive to the vanishing-gradient problem (in the case of RNN models), require increased computational resources and are less interpretable, compared to TBMs, for example. Highly complex DL models can provide accurate prediction but usually perform like “black boxes”, as it is difficult to understand why certain predictions are made. Furthermore, although DL models are suitable for all TCP horizons, RNN perform well for STTCP (1-hour ahead), but not for LT TCP (24 h ahead), since the sequential information becomes less relevant and useful as the TCP horizon is long [17,91,106,115,116].

Finally, there are various ensemble methods and in general, they provide a balanced approach, achieving high accuracy and robustness across diverse traffic scenarios, though they necessitate careful tuning and validation. As most of the ensemble methods are custom (besides stacking, voting, bagging, and boosting), interpretability can be challenging, and they require extensive development period and computational power [126,127,132,134].

In general, although there are many models presented in this review, as strengths and weaknesses are presented, this section identified that models that are both interpretable and perform sequence-to-sequence predictions are generally absent from the literature. As a result, providing such solutions could boost solving TCP problems.

7. Implications and Challenges for TCP

The development of TCP systems offers a valuable opportunity to improve how urban mobility is managed, with benefits that range from enhanced transportation efficiency to reduced environmental impact. However, implementing these systems in practical settings poses various difficulties. This section explores the potential benefits of TCP alongside the barriers that must be addressed to enable its effective application. By considering both its transformative impacts and the challenges it faces, we can better navigate the complexities involved in bringing these innovations to retailiation.

7.1. Transformative Impacts

TCP was developed rapidly and is supported by new sensing technologies, ML models, and integrated data pipelines. These advances have real-world implications for city infrastructure, economic activity, environmental health, and everyday life. Yet, implementing TCP methods in practice comes with a range of difficulties that must be confronted.

From a practical standpoint, accurate and timely congestion predictions can transform how we manage transportation networks. Traffic managers and urban planners could, for example, anticipate bottlenecks during the morning rush hour or before large public events, then adjust signal timing or recommend alternate routes to drivers [137]. Reliable forecasts can help first responders move more efficiently, dispatching emergency vehicles to avoid traffic, and assist logistics companies in scheduling deliveries to cut down travel times and fuel consumption. Ultimately, this can make roads safer, reduce greenhouse gas emissions, and alleviate stress for everyday commuters [138].

Economically, these benefits are substantial. Reduced congestion correlates with improved productivity, as workers spend less time stuck in traffic and goods reach markets more quickly. The potential cost savings for shipment transport are enormous [139]. Furthermore, by smoothing traffic flows, cities can decrease pollution from idling vehicles, improve local air quality, and enhance the overall urban experience. The policy implications are equally significant. Municipal authorities, by relying on advanced prediction models,

can design more forward-looking interventions concerning public transportation [140]. They might consider shifting traffic away from central areas at peak times, investing in additional public transit options, or introducing dynamic pricing schemes to discourage driving during rush hour [141].

7.2. Limitations and Barriers

Despite the positive outcomes, challenges remain. One key obstacle is data quality. Predictions rely heavily on accurate, up-to-date information, such as vehicle counts, speeds, or occupancy data from sensors and cameras. Sensors can malfunction, produce missing values, or degrade over time. Integrating various data sources like sensors, GPS data, and third-party information into one coherent model may require modeling through complex information networks [142] and often involves complex cleaning and calibration procedures. Without careful attention to data integrity, predictions may become unreliable [143].

Another challenge is the inherently dynamic and unpredictable nature of traffic. Conditions shift rapidly due to accidents, severe weather, special events, or unexpected surges in demand. While ML and DL models capture many complex patterns, their effectiveness can weaken when confronted with rare events or sudden disruptions, which are not well represented in historical data. Continuous model retraining or adaptive frameworks may be required, increasing computational costs and complexity [144,145].

Scalability and model transferability also pose problems. A model trained for one city might not work well in another, due to differences in road layouts, driver behavior, climate patterns, or urban design. Transportation departments might need custom models or extensive retraining, which increase both cost and time before practical implementation. This lack of generalizability can limit the widespread adoption of TCP techniques [146].

Interpretability is another area of concern. Highly complex DL models can deliver accurate forecasts but often act like “black boxes”, making it hard to understand why certain predictions are made. Urban planners and policymakers might be hesitant to rely on models they cannot fully interpret, especially when those predictions guide expensive infrastructure investments or are related to accidents [147]. Balancing accuracy with explainability is not easy. It requires domain expertise, thoughtful model development, and possibly the integration of simpler baseline models or Explainable AI (XAI) techniques [148].

Data privacy and security are also front and center. Traffic data, especially those from connected vehicles or mobile devices, have the potential to expose sensitive information about individual travel behaviors. This raises critical ethical and regulatory concerns, including how such data should be securely stored, who should have access to them, and how long they should be retained for. To address these challenges, strict compliance with data protection laws, like the General Data Protection Regulation (GDPR) in the European Union, is essential. This involves carefully anonymizing or aggregating data to protect privacy while ensuring that the data remain useful for generating accurate predictions [149].

To address concerns about security and sensitive traffic data, technologies like blockchain can play a role in enabling secure and decentralized data sharing. By offering tamper-proof mechanisms, blockchain could help traffic systems comply with strict privacy regulations while maintaining data integrity [150]. Furthermore, edge computing plays a significant role in processing this vast quantity of data locally, minimizing latency and enabling swift responses to changing traffic conditions. Edge computing solutions can accurately obtain vehicle count, speed, type, and direction, facilitating effective traffic monitoring [151]. Furthermore, integrating crowdsourced data and IoT devices has significantly enhanced TCP models. By combining data from connected vehicles, sensors, and mobile users, these models achieve real-time monitoring and proactive decision-making, leading

to optimized traffic flow and reduced congestion [152]. However, these technologies, like blockchain, edge computing, and IoT devices, integration into TCP and ITSs require careful consideration, due to potential issues, like computational load, development complexity, and scalability.

Finally, organizational and cultural barriers can slow the adoption of TCP [153]. Implementing predictive modeling tools into everyday traffic management may require new training for staff, adjustments to operational workflows, or upgrades in technology infrastructure. Some decision-makers might be skeptical of these methods until they see consistent, demonstrable improvements in traffic conditions. Gaining stakeholder trust, ensuring proper maintenance of the models, and aligning predictive insights with existing traffic control policies are all crucial steps.

In short, while TCP methods hold great promise for improving urban mobility, the path to widespread and effective deployment is not straightforward. Data integrity, model adaptability, interpretability, privacy concerns, and organizational readiness all play a role. Recognizing these challenges and working to address them will determine how fully the potential benefits of congestion prediction can be realized.

8. Future Directions

The concept of proactive intervention will likely come to the forefront [154]. Instead of merely predicting congestion, future systems could automatically suggest preemptive measures, like adjusting traffic signals before a traffic jam forms, sending targeted route recommendations to drivers, or nudging travelers to shift travel times. Reinforcement learning or agent-based models that continuously refine their strategies based on real-time outcomes may prove invaluable, making traffic systems more self-regulating and responsive [155].

Another promising direction is the use of transfer learning and domain adaptation techniques. Instead of building a new model from scratch for every city, researchers and practitioners could develop frameworks that leverage knowledge gained in one location to jumpstart predictions in another. This would speed up deployment, reduce costs, and make predictive technologies more accessible to cities with fewer resources.

Lessons from other forecasting application domains, where comparative studies of ML models have been used to optimize predictions, could inform similar strategies in TCP. Such techniques can help create adaptable models capable of delivering reliable results across different urban environments [156]. Furthermore, with the increase in electric vehicles and the need for charging demand, using TCP to predict energy charging demand could improve forecasting performance.

Moreover, advances in TCP are likely to focus on more holistic and adaptive approaches. Integrating data from multiple sources beyond the usual sensor or GPS feeds will be key [157]. Mobile phone data, drone-based imagery, and information gleaned from social media or shared mobility services can enrich traffic models. With connected and autonomous vehicles on the horizon, even more granular data on routes, speeds, and pedestrian activity will become accessible. These richer datasets will help models better anticipate unusual conditions, improve long-term planning, and potentially manage entire transportation networks more efficiently.

Efforts to improve interpretability and trust in models will also gain momentum. User-friendly visualizations, simplified model structures, and the incorporation of explainable AI methods can help transportation engineers and policymakers understand why a prediction was made, making it easier to justify actions taken based on that knowledge. Privacy-preserving data analysis will also remain a high priority, ensuring that traffic predictions can be both precise and respectful of individual rights.

TCP research should build on today's achievements while looking to make models more versatile, transparent, and responsive. By embracing new data sources, refining algorithms, ensuring privacy and interpretability, and integrating proactive management strategies, we move closer to a world of truly intelligent and sustainable transportation.

Finally, as mentioned in Section 6, this review has identified that models that are both interpretable and can perform sequence-to-sequence predictions are generally absent from the literature. As a result, providing such a solution could boost solving TCP problems.

9. Conclusions

Recent advances in TCP show considerable promises for shaping more efficient and resilient transportation networks. This review emphasized how blending classic statistical techniques with modern ML, DL, and ensemble methods could yield more accurate and flexible forecasts. By leveraging a range of forecasting horizons, employing appropriate evaluation metrics, and following a clear methodological process, practitioners are better equipped to capture the complex patterns underlying urban traffic flows.

These capabilities are becoming increasingly important as cities face growing populations and evolving mobility demands. Predictive traffic models can help ensure that supply better matches demand, reducing travel delays, improving safety, and curbing environmental impacts. Additionally, the inflow of high-quality data from the IoT and improved sensor technologies will continue refining our ability to anticipate congestion before it becomes problematic.

Future work must focus on making these models more interpretable, scalable, and adaptable, as well as ensuring robust data privacy. By addressing these concerns, researchers, planners, and policymakers can confidently integrate TCP tools into their decision-making processes. Ultimately, these predictive insights will serve as vital components of next-generation ITSs, contributing to more sustainable, responsive, and livable urban environments.

The key finding of this review and some points regarding implications, challenges, and future applications include:

- The main steps of TCP problems, include data collection and engineering, data pre-processing and TCP model selection.
- There are various parameters that can be considered for TCP, like weather data, time/seasonality features, street information, miscellaneous and historical traffic information.
- Statistical models remain effective for capturing linear and stationary patterns but struggle with fluctuations, such as nonlinear or dynamic traffic conditions.
- ML models like TBMs perform well in capturing nonlinear relationships and handling imbalanced datasets, often outperforming linear ML models in complex scenarios; however, they may be sensitive to overfitting.
- DL models, particularly RNNs, such as LSTM and GRU, perform well in capturing temporal dependencies, have the ability to perform seq2seq multi-step predictions utilizing a single model, but require large datasets, computational resources, and have limited interpretability.
- Ensemble methods offer robust solutions by combining the strengths of individual models, enhancing performance and adaptability across different TCP cases, but may have development complexities.
- Regarding practical implications and challenges, accurate TCP can optimize traffic management, reduce environmental impact, and support ITSs. However, ensuring data quality, integrating IoT data sources, improving model interpretability, addressing privacy concerns, and using technologies like blockchain need careful consideration.

- Future opportunities include the usage of technologies such as blockchain, edge computing, and the IoT to enhance TCP systems or new model implementations like sequence-to-sequence TBMs that combine both interpretability and sequential input/output data handling.

Author Contributions: Conceptualization, A.M.; methodology, A.M. and P.K.; Formal analysis, A.M.; Validation, A.M., P.K. and C.T.; writing—original draft preparation, A.M. and P.K.; writing—review and editing, A.M., P.K. and C.T.; supervision, C.T.; project administration/resources C.T.; funding acquisition, C.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AB	AdaBoost
ANN	Artificial Neural Networks
AI	Artificial Intelligence
ARMA	Autoregressive Moving Average
ARIMA	Auto-Regressive Integrated Moving Average
ARX	Auto-Regressive model with eXogenous inputs
AST3DRNet	Attention-Based Spatio-Temporal 3D Residual Neural Network
AUC	Area Under the Curve
BiLSTM	Bidirectional Long Short-Term Memory
BRR	Bayesian Ridge Regression
CB	CatBoost or Categorical Boosting
CDT	Cell Dwell Time
R^2	Coefficient of Determination
CNN	Convolutional Neural Network
CVRMSE	Coefficient of Variation of the Root-Mean-Square Error
DBN	Deep Belief Network
DL	Deep Learning
DT	Decision Tree
EN	Elastic Net
ET	Extra Trees
FNN	Feedforward Neural Network
FSM	Fourier Series Model
GBM	Geometric Brownian Motion
GB	Gradient Boosting
GRU	Gated Recurrent Unit
HMM	Hidden Markov Model
HGB	Histogram-Based Gradient Boosting
IoT	Internet of Things
LASSO	Least Absolute Shrinkage and Selection Operator
LGBM	Light Gradient Boosting Machine

LR	Linear Regression
LSTM	Long Short-Term Memory
LTTCP	Long-Term Traffic Congestion Prediction
MCC	Matthews Correlation Coefficient
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MR	Mean Reversion
ML	Machine Learning
MLP	Multi-Layer Perceptron
MLR	Multiple Linear Regression
MVHS-LSTM	Multiple Variables Heuristic Selection LSTM
MSE	Mean Squared Error
MTTCP	Medium-Term Traffic Congestion Prediction
NNs	Neural Networks
NRMSE	Normalized Root-Mean-Square Error
seq2seq	Sequence-to-sequence
PR	Polynomial Regression
RBFN	Radial Basis Function Network
RF	Random Forest
RR	Ridge Regression
AUC-ROC	ROC Curve and Area Under the Curve
RMSE	Root-Mean-Square Error
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
SARIMA	Seasonal Auto-Regressive Integrated Moving Average
SHAP	Shapley Additive Explanation
STTCP	Short-Term Traffic Congestion Prediction
SVM/R	Support Vector Machine/Regression
SW	Sliding Window
TCP	Traffic Congestion Prediction
TBM	Tree-Based Model
XAI	Explainable Artificial Intelligence
XGB	Extreme Gradient Boosting
ZI	Zero-inflated

References

1. United Nations. *World Urbanization Prospects: The 2018 Revision*; United Nations: New York, NY, USA, 2019.
2. Zhang, K.; Batterman, S. Air pollution and health risks due to vehicle traffic. *Sci. Total Environ.* **2013**, *450–451*, 307–316. [\[CrossRef\]](#)
3. Gakis, E.; Kehagias, D.; Tzovaras, D. Mining Traffic Data for Road Incidents Detection. In Proceedings of the 17th International IEEE Conference on Intelligent Transportation Systems, Qingdao, China, 8–11 October 2014.
4. Shokri, D.; Larouche, C.; Homayouni, S. A Comparative Analysis of Multi-Label Deep Learning Classifiers for Real-Time Vehicle Detection to Support Intelligent Transportation Systems. *Smart Cities* **2023**, *6*, 2982–3004. [\[CrossRef\]](#)
5. European Commission. Impact of Driving Conditions and Driving Behaviour—ULEV. 2023. Available online: <https://wikis.ec.europa.eu/display/ULEV/Impact+of+driving+conditions+and+driving+behaviour> (accessed on 31 October 2023).
6. Regragui, Y.; Moussa, N. A real-time path planning for reducing vehicles traveling time in cooperative-intelligent transportation systems. *Simul. Model. Pract. Theory* **2023**, *123*, 102710. [\[CrossRef\]](#)
7. Oladimeji, D.; Gupta, K.; Kose, N.A.; Gundogan, K.; Ge, L.; Liang, F. Smart Transportation: An Overview of Technologies and Applications. *Sensors* **2023**, *23*, 3880. [\[CrossRef\]](#)
8. Razali, N.A.M.; Shamsaimon, N.; Ishak, K.K.; Ramli, S.; Amran, M.F.M.; Sukardi, S. Gap, techniques and evaluation: Traffic flow prediction using machine learning and deep learning. *J. Big Data* **2021**, *8*, 152. [\[CrossRef\]](#)
9. Khan, A.; Aslam, S.; Aurangzeb, K.; Alhussein, M.; Javaid, N. Multiscale modeling in smart cities: A survey on applications, current trends, and challenges. *Sustain. Cities Soc.* **2022**, *78*, 103517. [\[CrossRef\]](#)
10. Mirindi, D. A Review of the Advances in Artificial Intelligence in Transportation System Development. *J. Civ. Constr. Environ. Eng.* **2024**, *9*, 72–83. [\[CrossRef\]](#)

11. Korkmaz, H.; Erturk, M.A. Prediction of the traffic incident duration using statistical and machine-learning methods: A systematic literature review. *Technol. Forecast. Soc. Change* **2024**, *207*, 123621. [\[CrossRef\]](#)
12. Jiang, W. Cellular traffic prediction with machine learning: A survey. *Expert Syst. Appl.* **2022**, *201*, 117163. [\[CrossRef\]](#)
13. Fan, X.; Xiang, C.; Gong, L.; He, X.; Qu, Y.; Amirgholipour, S.; Xi, Y.; Nanda, P.; He, X. Deep learning for intelligent traffic sensing and prediction: Recent advances and future challenges. *CCF Trans. Pervasive Comput. Interact.* **2020**, *2*, 240–260. [\[CrossRef\]](#)
14. Qi, Y.; Cheng, Z. Research on traffic congestion forecast based on deep learning. *Information* **2023**, *14*, 108. [\[CrossRef\]](#)
15. Rahman, M.M.; Joarder, M.M.A.; Nower, N. A comprehensive systematic literature review on traffic flow prediction (TFP). *Syst. Lit. Rev. Meta-Anal. J.* **2022**, *3*, 86–98. [\[CrossRef\]](#)
16. Mystakidis, A.; Tjortjis, C. Big Data Mining for Smart Cities: Predicting Traffic Congestion using Classification. In Proceedings of the 2020 11th International Conference on Information, Intelligence, Systems and Applications (IISA), Piraeus, Greece, 15–17 July 2020; pp. 1–8. [\[CrossRef\]](#)
17. Mystakidis, A.; Tjortjis, C. Traffic congestion prediction and missing data: A classification approach using weather information. *Int. J. Data Sci. Anal.* **2024**, 1–20. [\[CrossRef\]](#)
18. Xu, H.; Jiang, C. Deep belief network-based support vector regression method for traffic flow forecasting. *Neural Comput. Appl.* **2020**, *32*, 2027–2036. [\[CrossRef\]](#)
19. Lu, S.; Zhang, Q.; Chen, G.; Seng, D. A combined method for short-term traffic flow prediction based on recurrent neural network. *Alex. Eng. J.* **2021**, *60*, 87–94. [\[CrossRef\]](#)
20. Wang, Y.; Szeto, W.Y.; Han, K.; Friesz, T.L. Dynamic traffic assignment: A review of the methodological advances for environmentally sustainable road transportation applications. *Transp. Res. Part B Methodol.* **2018**, *111*, 370–394. [\[CrossRef\]](#)
21. Wu, C.H.; Ho, J.M.; Lee, D.T. Travel-time prediction with support vector regression. *IEEE Trans. Intell. Transp. Syst.* **2004**, *5*, 276–281. [\[CrossRef\]](#)
22. Ma, X.; Tao, Z.; Wang, Y.; Yu, H.; Wang, Y. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transp. Res. Part C Emerg. Technol.* **2015**, *54*, 187–197. [\[CrossRef\]](#)
23. Lv, Y.; Duan, Y.; Kang, W.; Li, Z.; Wang, F.Y. Traffic flow prediction with big data: A deep learning approach. *IEEE Trans. Intell. Transp. Syst.* **2014**, *16*, 865–873. [\[CrossRef\]](#)
24. Zhang, J.; Zheng, Y.; Qi, D. Deep spatio-temporal residual networks for citywide crowd flows prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31.
25. Kamarianakis, Y.; Prastacos, P. Space–time modeling of traffic flow. *Comput. Geosci.* **2005**, *31*, 119–133. [\[CrossRef\]](#)
26. Zhang, C.; Patras, P. Long-term mobile traffic forecasting using deep spatio-temporal neural networks. In Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing, Los Angeles, CA, USA, 26–29 June 2018; pp. 231–240.
27. Tsirigotis, L.; Vlahogianni, E.; Karlaftis, M. Does Information on Weather Affect the Performance of Short-Term Traffic Forecasting Models? *Int. J. Intell. Transp. Syst. Res.* **2012**, *10*, 1–10. [\[CrossRef\]](#)
28. Koesdwiady, A.; Soua, R.; Karray, F. Improving Traffic Flow Prediction with Weather Information in Connected Cars: A Deep Learning Approach. *IEEE Trans. Veh. Technol.* **2016**, *65*, 9508–9517. [\[CrossRef\]](#)
29. Chung, E.; Ohtani, O.; Warita, H.; Kuwahara, M.; Morita, H. Does weather affect highway capacity? In Proceedings of the 5th International Symposium on Highway Capacity and Quality of Service, Yokohama, Japan, 25–29 July 2006; pp. 139–146.
30. Nookala, L. Weather Impact on Traffic Conditions and Travel Time Prediction. Ph.D. Thesis, University of Minnesota, Minneapolis, MN, USA, 2006.
31. Akhtar, M.; Moridpour, S. A review of traffic congestion prediction using artificial intelligence. *J. Adv. Transp.* **2021**, *2021*, 8878011. [\[CrossRef\]](#)
32. Luo, X.; Li, D.; Zhang, S. Traffic flow prediction during the holidays based on DFT and SVR. *J. Sens.* **2019**, *2019*, 6461450. [\[CrossRef\]](#)
33. Li, H.; Lv, Z.; Li, J.; Xu, Z.; Wang, Y.; Sun, H.; Sheng, Z. Traffic flow forecasting in the COVID-19: A deep spatial-temporal model based on discrete wavelet transformation. *ACM Trans. Knowl. Discov. Data* **2023**, *17*, 1–28. [\[CrossRef\]](#)
34. Li, M.; Tong, P.; Li, M.; Jin, Z.; Huang, J.; Hua, X.S. Traffic flow prediction with vehicle trajectories. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 2–9 February 2021; Volume 35, pp. 294–302.
35. Kaya, S.; Kilic, N.; Kocak, T.; Gungor, C. From Asia to Europe: Short-term traffic flow prediction between continents. In Proceedings of the 2014 21st IEEE International Conference on Telecommunications (ICT), Lisbon, Portugal, 4–7 May 2014; pp. 277–282.
36. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
37. John, P.M.; Massaron, L. *Machine Learning for Dummies*, 2nd ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2021; pp. 1–467. ISBN 978-1-119-72401-8.

38. Khan, A.A.; Minai, A.F.; Devi, L.; Alam, Q.; Pachauri, R.K. Energy demand modelling and ANN based forecasting using MATLAB/simulink. In Proceedings of the 2021 IEEE International Conference on Control, Automation, Power and Signal Processing (CAPS), Jabalpur, India, 10–12 October 2021; pp. 1–6. [\[CrossRef\]](#)
39. Sipola, N. *Heat Demand Forecasting Models' Development: Use of Data Mining Tools in SQL Server Analysis Services*; Lappeenranta Teknillinen Yliopisto, Tuotantotalouden Tiedekunta, Tietotekniikka/Lappeenranta University of Technology, School of Industrial Engineering and Management, Computer Science: Lappeenranta, Finland, 2015. Available online: <http://lutpub.lut.fi/handle/10024/117310> (accessed on 2 March 2024).
40. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. Available online: <https://www.tensorflow.org/> (accessed on 12 December 2024).
41. Imambi, S.; Prakash, K.B.; Kanagachidambaresan, G. PyTorch. In *Programming with TensorFlow: Solution for Edge Computing Applications*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 87–104. [\[CrossRef\]](#)
42. Wes McKinney. Data Structures for Statistical Computing in Python. In Proceedings of the 9th Python in Science Conference; van der Walt, S., Millman, J., Eds.; SciPy: Austin, TX, USA, 2010; pp. 56–61. [\[CrossRef\]](#)
43. Harris, C.R.; Millman, K.J.; van der Walt, S.J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N.J.; et al. Array programming with NumPy. *Nature* **2020**, *585*, 357–362. [\[CrossRef\]](#)
44. Hunter, J.D. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* **2007**, *9*, 90–95. [\[CrossRef\]](#)
45. Zhang, Y.; Xin, D. Dynamic optimization long short-term memory model based on data preprocessing for short-term traffic flow prediction. *IEEE Access* **2020**, *8*, 91510–91520. [\[CrossRef\]](#)
46. Nourani, V.; Partoviyan, A. Hybrid denoising-jittering data pre-processing approach to enhance multi-step-ahead rainfall-runoff modeling. *Stoch. Environ. Res. Risk Assess.* **2018**, *32*, 545–562. [\[CrossRef\]](#)
47. Stepchenko, A.; Chizhov, J.; Aleksejeva, L.; Tolujew, J. Nonlinear, non-stationary and seasonal time series forecasting using different methods coupled with data preprocessing. *Procedia Comput. Sci.* **2017**, *104*, 578–585. [\[CrossRef\]](#)
48. Singla, P.; Duhan, M.; Saroha, S. A point and interval forecasting of solar irradiance using different decomposition based hybrid models. *Earth Sci. Inform.* **2023**, *16*, 2223–2240. [\[CrossRef\]](#)
49. Yu, W.; Su, J.; Zhang, W. Research on short-term traffic flow prediction based on wavelet de-noising preprocessing. In Proceedings of the 2013 IEEE Ninth International Conference on Natural Computation (ICNC), Shenyang, China, 23–25 July 2013; pp. 252–256.
50. Ilienescu, A.M.; Iovanovici, A.; Vladutiu, M. Supervised learning data preprocessing for short-term traffic flow prediction. In Proceedings of the 2022 IEEE 16th International Scientific Conference on Informatics (Informatics), Poprad, Slovakia, 23–25 November 2022; pp. 132–136.
51. Kwak, S.K.; Kim, J.H. Statistical data preparation: Management of missing values and outliers. *Korean J. Anesthesiol.* **2017**, *70*, 407–411. [\[CrossRef\]](#)
52. Tian, Y.; Zhang, K.; Li, J.; Lin, X.; Yang, B. LSTM-based traffic flow prediction with missing data. *Neurocomputing* **2018**, *318*, 297–305. [\[CrossRef\]](#)
53. Zheng, L.; Huang, H.; Zhu, C.; Zhang, K. A tensor-based K-nearest neighbors method for traffic speed prediction under data missing. *Transp. B Transp. Dyn.* **2020**, *8*, 182–199. [\[CrossRef\]](#)
54. Liu, J.; Ong, G.P.; Chen, X. GraphSAGE-Based Traffic Speed Forecasting for Segment Network with Sparse Data. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 1755–1766. [\[CrossRef\]](#)
55. Zhang, R.; Ashuri, B.; Deng, Y. A novel method for forecasting time series based on fuzzy logic and visibility graph. *Adv. Data Anal. Classif.* **2017**, *11*, 759–783. [\[CrossRef\]](#)
56. Manthiramoorathi, M.; Mani, M.; Murthy, A.G. Application of Pareto's Principle on Deep Learning Research Output: A Scientometric Analysis. In Proceedings of the International Conference on Machine Learning and Smart Technology-ICMLST, Chennai, India, 22–26 October 2021.
57. Jin, Y.; Gruna, R.; Sendhoff, B. Pareto analysis of evolutionary and learning systems. *Front. Comput. Sci. China* **2009**, *3*, 4–17. [\[CrossRef\]](#)
58. Gholamy, A.; Kreinovich, V.; Kosheleva, O. Why 70/30 or 80/20 Relation Between Training and Testing Sets: A Pedagogical Explanation. 2018. Available online: <https://www.cs.utep.edu/vladik/2018/tr18-09.pdf> (accessed on 3 March 2024).
59. Said, S.E.; Dickey, D.A. Testing for unit roots in autoregressive-moving average models of unknown order. *Biometrika* **1984**, *71*, 599–607. [\[CrossRef\]](#)
60. Williams, B.M.; Durvasula, P.K.; Brown, D.E. Urban freeway traffic flow prediction: Application of seasonal autoregressive integrated moving average and exponential smoothing models. *Transp. Res. Rec.* **1998**, *1644*, 132–141. [\[CrossRef\]](#)
61. Ghosh, B.; Basu, B.; O'Mahony, M. Multivariate short-term traffic flow forecasting using time-series analysis. *IEEE Trans. Intell. Transp. Syst.* **2009**, *10*, 246–254.
62. Ahmed, M.S.; Cook, A.R. Analysis of Freeway Traffic Time-Series Data by Using Box-Jenkins Techniques. *Transp. Res. Rec.* **1979**, *722*, 1–9.

63. Lee, S.; Fambro, D. Application of Subset Autoregressive Integrated Moving Average Model for Short-Term Freeway Traffic Volume Forecasting. *Transp. Res. Rec.* **1999**, *1678*, 179–188.
64. Williams, B.M.; Hoel, L.A. Modeling and Forecasting Vehicular Traffic Flow as a Seasonal ARIMA Process: Theoretical Basis and Empirical Results. *J. Transp. Eng.* **2003**, *129*, 664–672.
65. van der Voort, M.; Dougherty, M.; Watson, S. Combining Kohonen Maps and ARIMA Time Series Models to Forecast Traffic Flow. *Transp. Res. Part C Emerg. Technol.* **1996**, *4*, 307–318.
66. Zhang, N.; Zhang, Y.; Lu, H. Seasonal autoregressive integrated moving average and support vector machine models: Prediction of short-term traffic flow on freeways. *Transp. Res. Rec.* **2011**, *2215*, 85–92.
67. Okutani, I.; Stephanedes, Y.J. Dynamic Prediction of Traffic Volume Through Kalman Filtering Theory. *Transp. Res. Part B Methodol.* **1984**, *18*, 1–11. [\[CrossRef\]](#)
68. Emami, A.; Sarvi, M.; Bagloee, S.A. Short-term traffic flow prediction based on faded memory Kalman Filter fusing data from connected vehicles and Bluetooth sensors. *Simul. Model. Pract. Theory* **2020**, *102*, 102025. [\[CrossRef\]](#)
69. Ying, J.; Dong, X.; Li, B.; Tian, Z. Auto-Regressive Model with Exogenous Input (ARX) Based Traffic Flow Prediction. In Proceedings of the CICTP 2021, Xi'an, China, 16–19 December 2021; pp. 295–304.
70. Andrés Peñaranda, D.; Guevara, J.; Cabrales, S. Fourier Series for Seasonal Traffic Forecasting: An Application to a Real Toll Road Concession in Colombia. In Proceedings of the Construction Research Congress 2024, Des Moines, IA, USA, 20–23 March 2024; pp. 1238–1247.
71. Qi, Y.; Ishak, S. A Hidden Markov Model for short term prediction of traffic conditions on freeways. *Transp. Res. Part C Emerg. Technol.* **2014**, *43*, 95–111. [\[CrossRef\]](#)
72. Pongpaibool, P.; Tangamchit, P.; Noodwong, K. Evaluation of road traffic congestion using fuzzy techniques. In Proceedings of the TENCON 2007-2007 IEEE Region 10 Conference, Taipei, Taiwan, 30 October–2 November 2007; pp. 1–4. [\[CrossRef\]](#)
73. Porikli, F.; Li, X. Traffic congestion estimation using HMM models without vehicle tracking. In Proceedings of the IEEE Intelligent Vehicles Symposium, Parma, Italy, 14–17 June 2004; pp. 188–193. [\[CrossRef\]](#)
74. Pun, L.; Zhao, P.; Liu, X. A multiple regression approach for traffic flow estimation. *IEEE Access* **2019**, *7*, 35998–36009.
75. Li, D. Predicting short-term traffic flow in urban based on multivariate linear regression model. *J. Intell. Fuzzy Syst.* **2020**, *39*, 1417–1427. [\[CrossRef\]](#)
76. Al-Hamadi, H.; Soliman, S. Long-term/mid-term electric load forecasting based on short-term correlation and annual growth. *Electr. Power Syst. Res.* **2005**, *74*, 353–361. [\[CrossRef\]](#)
77. Hoerl, A.E.; Kennard, R.W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **1970**, *12*, 55–67. [\[CrossRef\]](#)
78. Scholkopf, B.; Smola, A.J.; Williamson, R.C.; Bartlett, P.L. New support vector algorithms. *Neural Comput.* **2000**, *12*, 1207–1245. [\[CrossRef\]](#)
79. Asadi, M.; TaghaviGhalesari, A.; Kumar, S. Machine learning techniques for estimation of Los Angeles abrasion value of rock aggregates. *Eur. J. Environ. Civ. Eng.* **2022**, *26*, 964–977. [\[CrossRef\]](#)
80. Nidhi, N.; Lobiyal, D. Traffic flow prediction using support vector regression. *Int. J. Inf. Technol.* **2022**, *14*, 619–626.
81. Efron, B.; Hastie, T.; Johnstone, I.; Tibshirani, R. Least angle regression. *Ann. Stat.* **2004**, *32*, 407–499. [\[CrossRef\]](#)
82. Zou, H.; Hastie, T. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **2005**, *67*, 301–320. [\[CrossRef\]](#)
83. Seber, G.A.F.; Lee, A.J. *Linear Regression Analysis*; Wiley: John & Sons: Hoboken, NJ, USA, 2012. ISBNs 1118274423/9781118274422.
84. MacKay, D.J.C. Bayesian interpolation. *Neural Comput.* **1992**, *4*, 415–447. [\[CrossRef\]](#)
85. Van Schaijik, M. *Threshold Regression Estimation via Lasso, Elastic-Net, and Lad-Lasso: A Simulation Study with Applications to Urban Traffic Data*; Arizona State University: Tempe, AZ, USA, 2015.
86. Shiomi, Y.; Toriumi, A.; Nakamura, H. International analysis on social and personal determinants of traffic violations and accidents employing logistic regression with elastic net regularization. *IATSS Res.* **2022**, *46*, 36–45.
87. Guikema, S.; Quiring, S. Hybrid data mining-regression for infrastructure risk assessment based on zero-inflated data. *Reliab. Eng. Syst. Saf.* **2012**, *99*, 178–182. [\[CrossRef\]](#)
88. Lee, S.K.; Jin, S. Decision tree approaches for zero-inflated count data. *J. Appl. Stat.* **2006**, *33*, 853–865. [\[CrossRef\]](#)
89. Gruber, N.; Jockisch, A. Are GRU Cells More Specific and LSTM Cells More Sensitive in Motive Classification of Text? *Front. Artif. Intell.* **2020**, *3*, 40. [\[CrossRef\]](#) [\[PubMed\]](#)
90. Shih, Y.S.; Liu, K.H. Regression trees for detecting preference patterns from rank data. *Adv. Data Anal. Classif.* **2019**, *13*, 683–702. [\[CrossRef\]](#)
91. Tsalikidis, N.; Mystakidis, A.; Koukaras, P.; Ivaškevičius, M.; Morkūnaitė, L.; Ioannidis, D.; Fokaides, P.A.; Tjortjis, C.; Tzovaras, D. Urban traffic congestion prediction: A multi-step approach utilizing sensor data and weather information. *Smart Cities* **2024**, *7*, 233–253. [\[CrossRef\]](#)

92. Rambabu, M.; Ramakrishna, N.; Polamarasetty, P.K. Prediction and Analysis of Household Energy Consumption by Machine Learning Algorithms in Energy Management. *E3S Web Conf.* **2022**, *350*, 02002. [\[CrossRef\]](#)
93. Myles, A.J.; Feudale, R.N.; Liu, Y.; Woody, N.A.; Brown, S.D. An introduction to decision tree modeling. *J. Chemom. J. Chemom. Soc.* **2004**, *18*, 275–285. [\[CrossRef\]](#)
94. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [\[CrossRef\]](#)
95. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Proceedings of the Advances in Neural Information Processing Systems*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Long Beach, CA, USA, 2017; Volume 30. Available online: https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf (accessed on 2 March 2024).
96. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining KDD '16*, New York, NY, USA, 13–17 August 2016; pp. 785–794. [\[CrossRef\]](#)
97. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [\[CrossRef\]](#)
98. Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Drogush, A.V.; Gulin, A. CatBoost: Unbiased boosting with categorical features. *Adv. Neural Inf. Process. Syst.* **2018**, *31*. [\[CrossRef\]](#)
99. Hastie, T.; Rosset, S.; Zhu, J.; Zou, H. Multi-class adaboost. *Stat. Its Interface* **2009**, *2*, 349–360. [\[CrossRef\]](#)
100. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. *Mach. Learn.* **2006**, *63*, 3–42. [\[CrossRef\]](#)
101. Thianniwet, T.; Phosaard, S.; Pattara-Atikom, W. Classification of Road Traffic Congestion Levels from GPS Data using a Decision Tree Algorithm and Sliding Windows. *Lect. Notes Eng. Comput. Sci.* **2010**, *60*, 261–271.
102. Gheni, A.A. Urban Traffic Congestion Prediction Using GTFS Data and Advanced Machine Learning Models. *Int. Res. J. Innov. Eng. Technol.* **2024**, *8*, 25–31. [\[CrossRef\]](#)
103. Dong, X.; Lei, T.; Jin, S.; Hou, Z. Short-term traffic flow prediction based on XGBoost. In *Proceedings of the 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS)*, Enshi, China, 25–27 May 2018; pp. 854–859.
104. Huang, Y.; Jiang, S.; Jafari, M.; Jin, P.J. Real-Time Driver and Traffic Data Integration for Enhanced Road Safety. *IEEE Trans. Comput. Soc. Syst.* **2024**, *11*, 7711–7722. [\[CrossRef\]](#)
105. Zhao, X.; Qi, H.; Yao, Y.; Guo, M.; Su, Y. Traffic order analysis of intersection entrance based on aggressive driving behavior data using CatBoost and SHAP. *J. Transp. Eng. Part A Syst.* **2023**, *149*, 04023037. [\[CrossRef\]](#)
106. Marblestone, A.H.; Wayne, G.; Kording, K.P. Toward an integration of deep learning and neuroscience. *Front. Comput. Neurosci.* **2016**, *10*, 215943. [\[CrossRef\]](#)
107. Jain, L.C.; Medsker, L.R. Design and Applications. In *Proceedings of the Recurrent Neural Networks*; CRC Press: Boca Raton, FL, USA, 1999. Available online: <https://api.semanticscholar.org/CorpusID:262144264> (accessed on 2 March 2024). [\[CrossRef\]](#)
108. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [\[CrossRef\]](#) [\[PubMed\]](#)
109. Dey, R.; Salem, F.M. Gate-variants of gated recurrent unit (GRU) neural networks. In *Proceedings of the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Boston, MA, USA, 6–9 August 2017; pp. 1597–1600.
110. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 1–74.
111. Khan, A.; Zameer, A.; Jamal, T.; Raza, A. Deep belief networks based feature generation and regression for predicting wind power. *arXiv* **2018**, arXiv:1807.11682. [\[CrossRef\]](#)
112. Khalid, M. Wind power economic dispatch–impact of radial basis functional networks and battery energy storage. *IEEE Access* **2019**, *7*, 36819–36832. [\[CrossRef\]](#)
113. Pattara-Atikom, W.; Peachavanish, R.; Luckana, R. Estimating Road Traffic Congestion using Cell Dwell Time with Simple Threshold and Fuzzy Logic Techniques. In *Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference*, Seattle, WA, USA, 30 September–3 October 2007; pp. 956–961.
114. Zhuang, W.; Cao, Y. Short-Term Traffic Flow Prediction Based on CNN-BILSTM with Multicomponent Information. *Appl. Sci.* **2022**, *12*, 8714. [\[CrossRef\]](#)
115. Shin, D.H.; Chung, K.; Park, R.C. Prediction of traffic congestion based on LSTM through correction of missing temporal and spatial data. *IEEE Access* **2020**, *8*, 150784–150796. [\[CrossRef\]](#)
116. Jeong, M.H.; Lee, T.Y.; Jeon, S.B.; Youm, M. Highway speed prediction using gated recurrent unit neural networks. *Appl. Sci.* **2021**, *11*, 3059. [\[CrossRef\]](#)
117. Chen, G.; Zhang, J. Applying Artificial Intelligence and Deep Belief Network to predict traffic congestion evacuation performance in smart cities. *Appl. Soft Comput.* **2022**, *121*, 108692. [\[CrossRef\]](#)
118. Zhai, Y.; Wan, Y.; Wang, X. Optimization of Traffic Congestion Management in Smart Cities under Bidirectional Long and Short-Term Memory Model. *J. Adv. Transp.* **2022**, *2022*, 3305400. [\[CrossRef\]](#)

119. Ranjan, N.; Bhandari, S.; Zhao, H.P.; Kim, H.; Khan, P. City-wide traffic congestion prediction based on CNN, LSTM and transpose CNN. *IEEE Access* **2020**, *8*, 81606–81620. [\[CrossRef\]](#)
120. Zhang, T.; Liu, Y.; Cui, Z.; Leng, J.; Xie, W.; Zhang, L. Short-term traffic congestion forecasting using attention-based long short-term memory recurrent neural network. In *Proceedings of the International Conference on Computational Science*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 304–314.
121. Li, L.; Dai, F.; Huang, B.; Wang, S.; Dou, W.; Fu, X. AST3DRNet: Attention-Based Spatio-Temporal 3D Residual Neural Networks for Traffic Congestion Prediction. *Sensors* **2024**, *24*, 1261. [\[CrossRef\]](#) [\[PubMed\]](#)
122. Tukey, J.W. Some Thoughts on Clinical Trials, Especially Problems of Multiplicity. *Science* **1977**, *198*, 679–684. [\[CrossRef\]](#) [\[PubMed\]](#)
123. Ahuja, R.; Sharma, S.C. Stacking and voting ensemble methods fusion to evaluate instructor performance in higher education. *Int. J. Inf. Technol.* **2021**, *13*, 1721–1731. [\[CrossRef\]](#)
124. Sarajcev, P.; Kunac, A.; Petrovic, G.; Despalatovic, M. Power System Transient Stability Assessment Using Stacked Autoencoder and Voting Ensemble. *Energies* **2021**, *14*, 3148. [\[CrossRef\]](#)
125. Bühlmann, P. Bagging, Boosting and Ensemble Methods. In *Handbook of Computational Statistics*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 985–1022. [\[CrossRef\]](#)
126. Huang, Y.; Zhang, J.; Bao, H.; Yang, Y.; Yang, J. Complementary Fusion of Deep Network and Tree Model for ETA Prediction. In *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*, Beijing, China, 2–5 November 2021; pp. 638–641.
127. Awan, A.A.; Majid, A.; Riaz, R.; Rizvi, S.S.; Kwon, S.J. A Novel Deep Stacking-Based Ensemble Approach for Short-Term Traffic Speed Prediction. *IEEE Access* **2024**, *12*, 15222–15235. [\[CrossRef\]](#)
128. Huang, H.; Chin, H.C. Modeling road traffic crashes with zero-inflation and site-specific random effects. *Stat. Methods Appl.* **2010**, *19*, 445–462. [\[CrossRef\]](#)
129. Chen, F.; Chen, S.; Ma, X. Crash frequency modeling using real-time environmental and traffic data and unbalanced panel data models. *Int. J. Environ. Res. Public Health* **2016**, *13*, 609. [\[CrossRef\]](#)
130. Shi, R.; Du, L. Multi-Section Traffic Flow Prediction Based on MLR-LSTM Neural Network. *Sensors* **2022**, *22*, 7517. [\[CrossRef\]](#)
131. Rizal, A.A.; Soraya, S.; Tajuddin, M. Sequence to sequence analysis with long short term memory for tourist arrivals prediction. *J. Phys. Conf. Ser.* **2019**, *1211*, 012024. [\[CrossRef\]](#)
132. Feng, B.; Xu, J.; Zhang, Y.; Lin, Y. Multi-step traffic speed prediction based on ensemble learning on an urban road network. *Appl. Sci.* **2021**, *11*, 4423. [\[CrossRef\]](#)
133. Zięba, M.; Przewieźlikowski, M.; Śmieja, M.; Tabor, J.; Trzciński, T.; Spurek, P. Regflow: Probabilistic flow-based regression for future prediction. In *Proceedings of the Asian Conference on Intelligent Information and Database Systems*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 267–279.
134. Cini, N.; Aydin, Z. A Deep Ensemble Approach for Long-Term Traffic Flow Prediction. *Arab. J. Sci. Eng.* **2024**, *49*, 12377–12392.
135. Guo, C.; Zhu, J.; Wang, X. MVHS-LSTM: The Comprehensive Traffic Flow Prediction Based on Improved LSTM via Multiple Variables Heuristic Selection. *Appl. Sci.* **2024**, *14*, 2959. [\[CrossRef\]](#)
136. Hossin, M.; Sulaiman, M.N. A review on evaluation metrics for data classification evaluations. *Int. J. Data Min. Knowl. Manag. Process.* **2015**, *5*, 1.
137. Ma, X.; Yu, H.; Wang, Y.; Wang, Y. Large-scale transportation network congestion evolution prediction using deep learning theory. *PLoS ONE* **2015**, *10*, e0119044. [\[CrossRef\]](#)
138. Falcocchio, J.C.; Levinson, H.S. The Costs and Other Consequences of Traffic Congestion. In *Road Traffic Congestion: A Concise Guide*; Springer International Publishing: Cham, Switzerland, 2015; pp. 159–182. [\[CrossRef\]](#)
139. Hensher, D.A. Tackling road congestion—What might it look like in the future under a collaborative and connected mobility model? *Transp. Policy* **2018**, *66*, A1–A8. [\[CrossRef\]](#)
140. Nguyen-Phuoc, D.Q.; Young, W.; Currie, G.; De Gruyter, C. Traffic congestion relief associated with public transport: State-of-the-art. *Public Transp.* **2020**, *12*, 455–481. [\[CrossRef\]](#)
141. Wang, Y.; Zhong, H. Mitigation strategies for controlling urban particulate pollution from traffic congestion: Road expansion and road public transport. *J. Environ. Manag.* **2023**, *345*, 118795. [\[CrossRef\]](#) [\[PubMed\]](#)
142. Koukaras, P.; Berberidis, C.; Tjortjis, C. A Semi-supervised Learning Approach for Complex Information Networks. In *Proceedings of the Intelligent Data Communication Technologies and Internet of Things*; Hemanth, J., Bestak, R., Chen, J.I.Z., Eds.; Springer: Singapore, 2021; pp. 1–13.
143. Faouzi, N.E.E.; Leung, H.; Kurian, A. Data fusion in intelligent transportation systems: Progress and challenges—A survey. *Inf. Fusion* **2011**, *12*, 4–10. [\[CrossRef\]](#)
144. Hammad, M.A.; Jereb, B.; Rosi, B.; Dragan, D. Methods and Models for Electric Load Forecasting: A Comprehensive Review. *Logist. Supply Chain Sustain. Glob. Chall.* **2020**, *11*, 51–76. [\[CrossRef\]](#)

145. Mystakidis, A.; Ntozi, E.; Afentoulis, K.; Koukaras, P.; Gkaidatzis, P.; Ioannidis, D.; Tjortjis, C.; Tzovaras, D. Energy generation forecasting: Elevating performance with machine and deep learning. *Computing* **2023**, *105*, 1623–1645. [\[CrossRef\]](#)
146. Jin, Y.; Chen, K.; Yang, Q. Transferable Graph Structure Learning for Graph-based Traffic Forecasting Across Cities. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining KDD '23, Long Beach, CA, USA, 6–10 August 2023; pp. 1032–1043. [\[CrossRef\]](#)
147. Li, C.; Zhang, B.; Wang, Z.; Yang, Y.; Zhou, X.; Pan, S.; Yu, X. Interpretable Traffic Accident Prediction: Attention Spatial–Temporal Multi-Graph Traffic Stream Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 15574–15586. [\[CrossRef\]](#)
148. Barredo Arrieta, A.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; Garcia, S.; Gil-Lopez, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115. [\[CrossRef\]](#)
149. Liu, Y.; Yu, J.J.Q.; Kang, J.; Niyato, D.; Zhang, S. Privacy-Preserving Traffic Flow Prediction: A Federated Learning Approach. *IEEE Internet Things J.* **2020**, *7*, 7751–7763. [\[CrossRef\]](#)
150. Koukaras, P.; Afentoulis, K.D.; Gkaidatzis, P.A.; Mystakidis, A.; Ioannidis, D.; Vagropoulos, S.I.; Tjortjis, C. Integrating Blockchain in Smart Grids for Enhanced Demand Response: Challenges, Strategies, and Future Directions. *Energies* **2024**, *17*, 1007. [\[CrossRef\]](#)
151. Khan, A.; Khattak, K.S.; Khan, Z.H.; Gulliver, T.A.; Abdullah. Edge computing for effective and efficient traffic characterization. *Sensors* **2023**, *23*, 9385. [\[CrossRef\]](#) [\[PubMed\]](#)
152. Moumen, I.; Abouchabaka, J.; Rafalia, N. Enhancing urban mobility: Integration of IoT road traffic data and artificial intelligence in smart city environment. *Indones. J. Electr. Eng. Comput. Sci.* **2023**, *32*, 985–993. [\[CrossRef\]](#)
153. Shaygan, M.; Meese, C.; Li, W.; Zhao, X.G.; Nejad, M. Traffic prediction using artificial intelligence: Review of recent advances and emerging opportunities. *Transp. Res. Part C Emerg. Technol.* **2022**, *145*, 103921. [\[CrossRef\]](#)
154. Koukaras, P.; Dimara, A.; Herrera, S.; Zangrando, N.; Krinidis, S.; Ioannidis, D.; Fraternali, P.; Tjortjis, C.; Anagnostopoulos, C.N.; Tzovaras, D. Proactive Buildings: A Prescriptive Maintenance Approach. In *Proceedings of the Artificial Intelligence Applications and Innovations. AIAI 2022 IFIP WG 12.5 International Workshops*; Maglogiannis, I., Iliadis, L., Macintyre, J., Cortez, P., Eds.; Springer: Cham, Switzerland, 2022; pp. 289–300.
155. Gerostathopoulos, I.; Pournaras, E. TRAPPED in Traffic? A Self-Adaptive Framework for Decentralized Traffic Optimization. In Proceedings of the 2019 IEEE/ACM 14th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS), Montreal, QC, Canada, 25 May 2019; pp. 32–38. [\[CrossRef\]](#)
156. Koukaras, P.; Mustapha, A.; Mystakidis, A.; Tjortjis, C. Optimizing Building Short-Term Load Forecasting: A Comparative Analysis of Machine Learning Models. *Energies* **2024**, *17*, 1450. [\[CrossRef\]](#)
157. Xing, J.; Wu, W.; Cheng, Q.; Liu, R. Traffic state estimation of urban road networks by multi-source data fusion: Review and new insights. *Phys. A Stat. Mech. Appl.* **2022**, *595*, 127079. [\[CrossRef\]](#)

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.