The Journal of Supercomputing (2025) 81:511
https://doi.org/10.1007/s11227-025-06979-4

RESEARCH

A novel geo-independent and privacy-preserved traffic speed prediction framework based on deep learning for intelligent transportation systems

Murat Akin[1] · **Yavuz Canbay**[2] · **Seref Sagiroglu**[3]

Accepted: 24 January 2025
© The Author(s) 2025

Abstract

Intelligent transportation systems (ITS) involve collecting, storing, and delivering real-time traffic data to optimize efficiency, enhance transportation safety, and reduce energy consumption by integrating advanced electronics, information systems, and telecommunication technologies into roads, vehicles, and related infrastructure. These systems typically use some indicators such as speed, flow, and density to predict traffic information, which are measured by geographically fixed-position sensors located on the roads. Privacy leakage is a major problem since traffic information derived from users' or vehicles' mobility data contains sensitive information. Consequently, existing approaches are constrained by geographical dependencies, restrictions and privacy vulnerabilities. This study introduces a state-of-the-art, privacy-preserving, feature engineering-based, geo-independent traffic speed prediction framework to address these challenges for the first time. The framework, evaluated using performance metrics such as mean absolute error (MAE), mean absolute percentage error (MAPE), mean square error (MSE), and root-mean-square error (RMSE) on a real-world dataset, was implemented using long short-term memory, bidirectional long short-term memory, gated recurrent unit (GRU), linear regression, support vector regression, multilayer perceptron and random forest regression, and differential privacy with varying epsilon values. According to the results, GRU yielded the most favorable outcome with MAE of 7.7, MAPE of 28.5%, MSE of 132.7, and RMSE of 11.5 for Istanbul, MAE of 9.7, MAPE of 17.2%, MSE of 177.4, and RMSE of 13.2 for Eskisehir, and MAE of 3.1, MAPE of 5.4%, MSE of 12.4, and RMSE of 3.4 for Konya Roads with the epsilon value of 200, respectively. These findings highlight the framework's potential to advance privacy-aware traffic management and urban mobility solutions, offering a transformative approach for ITS applications.

Keywords Intelligent transportation systems · Traffic speed prediction · Geo-independent framework · Privacy-preserving feature engineering

Extended author information available on the last page of the article

Published online: 17 February 2025

1 Introduction

The population in urban centers is experiencing daily growth due to increased industrialization and improved living standards. Simultaneously, there is a notable increase in the number of vehicles participating in traffic, leading to challenges in terms of mobility in both developing residential areas and urban centers. In urban areas, the utilization of a limited transportation network by vehicles is necessitated due to the inability of the current transportation infrastructure to accommodate the rapid growth in the number of vehicles [1]. The enhancement of the urban and intercity road transportation network is undertaken by governmental and public entities, which construct new roads as required. Nevertheless, the network's capacity fails to meet the increasing demand posed by ever-growing number of vehicles, leading to a progressive rise in traffic density within urban areas. As a result, entities engaged in transportation management are actively working to devise strategies to mitigate challenges related to traffic regulation, driver well-being, and environmental impact [2].

Intelligent transportation systems incorporate sensing and communication technology to enhance travel and transportation efficiency, safety, and comfort. The field of traffic research and engineering investigates traffic behavior to build smooth and safe traffic applications while considering economic conditions, thanks to the expansion of traffic resources along with the development of technology [3]. Traffic indicators are expressed as flow, density, and speed which are the three fundamental factors used to understand traffic's behavior and characteristics and to define a traffic network [4, 5]. Traffic systems use these indicators to display road flow information to users in various colors, suggest alternative routes, and calculate journey times so that users can plan their trips effectively.

Traffic prediction systems traditionally use various sensor technologies to observe traffic parameters such as speed, flow, and density. These sensors include loop detection sensors, video cameras, radar-based sensors, ultrasonic sensors, and GPS-based mobile devices. Loop sensors, which are placed on road surfaces, are one of the oldest and still widely used technologies. These sensors can determine flow rates, but due to their static nature, they offer limited opportunities in terms of spatial coverage and are insufficient for determining traffic estimation dynamics in unobserved areas. Radar and ultrasonic sensors are additional sensors used to detect nearby vehicles and determine vehicle speeds. These systems also provide limited opportunities due to their high installed cost and fixed installations. Video-based systems, which use computer algorithms, process incoming data from cameras and provide real-time traffic information. These systems are frequently used thanks to advancements in machine learning and image processing, but video-based systems may not be very effective due to problems such as visibility problems, computational costs, and limitations in data recording. In addition, such sensors can obtain sensitive information that may identify individuals. Therefore, privacy concerns have come to the forefront in recent years, and mobile GPS-based devices have become an important data source and enable real-time traffic prediction over a wide geographic area [6, 7].

Current traffic prediction methods use different types of data by combining heterogeneous sensor technology in order to enhance security and coverage. For example, hybrid systems can use structures such as loop detectors and video cameras to provide prediction models that operate in areas not equipped with sensors, together with both spatial and GPS data. However, these approaches are limited due to static sensor applications. In addition, they face high infrastructure costs and inadequate privacy protection. Hence, there is a need for solutions that address these limitations. Such solutions should perform dynamic and geographically independent traffic prediction by collecting data from moving vehicles and they must also should integrate a strong privacy protection mechanism to protect sensitive user data [8].

Traffic speed prediction is an important component of intelligent transportation systems, but it faces several challenges that affect its effectiveness and scalability [6, 9]. The first challenge is data diversity and accessibility. Many existing systems collect data from fixed-position sensors such as loop detectors, radars, and cameras. These sensors are installed at various points on roads or intersections. Consequently, many areas of the road, especially in rural areas, are not monitored, making it difficult to make accurate predictions for these areas. In addition, if inconsistent or incomplete data are collected due to faulty sensors or communication errors, this becomes a significant problem for data diversity. Sensor limitations are another challenge. Fixed sensors cause a limitation for collecting sufficient data outside the area where they are located. High installation and management costs further impact their use. Mobile sensors, such as sensors embedded in vehicles or phones, offer a wider range of use, but due data collection from different sources, issues like sampling frequency inconsistencies and notification delays may arise, affecting data reliability. Prediction accuracy and complexity is another issue. It can be stated that traffic patterns are highly affected by complex spatial and temporal dependencies, including the topology of the road, weather conditions, and time of day effects. Correctly capturing these dependencies requires advanced modeling techniques. Traditional methods fail to integrate the effectiveness of these dependencies and produce suboptimal predictions. In addition, instantaneous changes in traffic conditions, such as accidents or road closures, can result in unpredictable dynamics that are difficult to model. Another challenge is geographical dependency and scalability. Current traffic prediction methods are often dependent on specific geographical areas where sensors are installed. These geographical dependency constraints do not allow the system to adapt to new locations effectively. Furthermore, as urban centers grow rapidly and the number of vehicles increases, the scalability of prediction systems poses greater challenges, especially in real-time applications. Privacy issues are the final challenge to be addressed. Traffic speed estimation systems usually operate on sensitive data such as GPS locations, vehicle speeds, and user identification information. This data allows detailed extraction of users' mobility tracks, raising significant privacy concerns. Existing privacy-preserving methods such as anonymization or federated learning are often insufficient or computationally expensive, limiting their applicability in the large-scale management of real-time data.

The increase in urbanization and the number of vehicles has led to significant challenges in traffic management and has consequently driven the development of intelligent transportation systems. Current traffic prediction models are primarily

based on geographically fixed-position sensors. This limits their applicability to certain regions and presents a significant constraint. Moreover, they ignore the natural dynamics of traffic. In addition, these methods process a various user-based sensitive information such as vehicle speed and location, raising significant privacy concerns. Despite the developments in intelligent transportation systems, there is no study in the literature that simultaneously addresses both geographical constraints and privacy-based vulnerabilities. To address this gap, a new geographically independent and privacy-preserving traffic speed prediction framework is proposed in this study. Unlike traditional methods, the proposed framework does not depend on fixed sensors, it can make dynamic and location-selectable predictions, and this takes into account various dynamics of the city. More so, the engineering process is enhanced by the integration of differential privacy, and the risk of privacy violations is mitigated by using advanced models. The proposed framework presents high effectiveness, and it provides the best result on a real-world dataset. This study introduces a new method for advanced intelligent transportation systems, addressing both privacy vulnerabilities and geographical constraints. The findings present that the framework can help improve the traffic model, while also indicating that it can be a transition tool for future transportation systems [10–12].

Speed, flow, and density indicators are used to determine traffic information and are measured from some sensors installed at fixed geographical points on roads. Prediction efforts for traffic information by using data provided by these sensors have several disadvantages including the following: The predictions on traffic information are limited to the geographical area where the sensors are located. Additionally, dynamic prediction of traffic information for different required segments of a region is not applicable. Furthermore, dynamic routing of traffic for optional roads, including different segments, is unavailable.

While location-based services are provided via mobile applications, institutions or companies that create these applications collect and analyze the location information obtained from these services on the devices [13]. The data collected from vehicles moving on the roads encompass the users' identities, locations, and speeds. As a result, significant privacy concerns may emerge [14]. For instance, an individual's geographical position and speed play a crucial role in determining their level of mobility. Consequently, any point of interest and driver speed can be exploited by malicious actors seeking to acquire sensitive information about consumers [15]. Hence, processing this kind of data requires robust privacy protection measures.

Current traffic prediction studies mainly collect data from geographically fixed sensors, mobile devices, or vehicle networks. Although these traditional methods provide information about traffic dynamics, they also raise significant privacy concerns. Collecting and analyzing data such as GPS coordinates and vehicle speeds poses risks of privacy violations. For example, mobility patterns extracted from location data can reveal sensitive information about human behavior such as points of interest and travel habits. If this information is disclosed, it can lead to unauthorized profiling and identity theft or surveillance. Moreover, inference

attacks, which allow adversarial inferences to be made by analyzing the collected data, also reveal privacy violations [16, 17]. For instance, Liu et al. [18] reported that Tesla Motors exposed the vehicles' location data by using GPS information for traffic prediction, posing significant security risks to the vehicle owner.

Many studies in the literature have aimed to anonymize data or develop federated learning-based approaches to reduce these concerns. However, these techniques often fail to guarantee privacy protection. Anonymization methods are vulnerable to re-identification attacks, particularly when auxiliary datasets are used. Although federated learning reduces the need for data sharing, it can leak sensitive information through model updates or gradient analysis. In addition, many current methods fail to provide privacy concerns within the scope of a geo-independent context [19–21].

In order to overcome the mentioned difficulties, a privacy-preserving traffic speed estimation framework is proposed in this study. As a new method, the proposed framework provides strong privacy protection by using differential privacy. By preserving sensitive attributes and minimizing privacy violations, the proposed framework is proposed as a new benchmark for intelligent transportation systems by providing a scalable and geo-independent solution. Current studies on traffic information prediction rely on the aforementioned indicators provided by data suppliers. The values of these indicators are measured from location-fixed sensors installed on roads. According to the literature, no study has considered dynamic and geographical location-free predictions. In addition, the privacy issue remains another challenge for traffic information predictions. To eliminate these challenges, we propose a state-of-the-art framework that enables dynamic and geographical location-free measurement for predictions and offers a privacy solution for sensitive data.

Current traffic speed prediction systems face several key challenges, such as:

1. Geographical location dependency: The fundamental indicators, including speed, volume, and flow, supplied by data providers, are typically utilized in traffic prediction studies. However, these indicators are measured from location-fixed sensors installed on the roads. This means that these systems only predict the speed of vehicles for restricted areas.
2. Dynamic traffic prediction: Researchers are unable to select particular road segments for which they aim to generate dynamic traffic predictions in existing studies.
3. Privacy concerns and inference attack issue: The studies on traffic information prediction generally employ sensitive information such as speed and coordinates, which can comprise the privacy of users who provide the data to the system.

To overcome these challenges, the proposed framework presents appropriate solutions:

1. Adoption of processing geo-independent traffic information: To address the geographically limited processing of data, the proposed framework supports the processing of traffic indicators for vehicles moving on the roads.
2. User-defined segment selection: The proposed framework allows users to select specific road segments for predicting traffic information, giving them greater control over the prediction process.
3. Implementation of differential privacy for feature engineering: The proposed framework incorporates differential privacy, ensuring that sensitive user features handled during the feature engineering step are protected from disclosure by attackers.

The main contributions of the study are given as follows:

1. We introduce a state-of-the-art, geographical location-independent traffic speed prediction framework. Our framework supports the processing of real-world and dynamic data retrieved from moving vehicles on the roads. A state-of-the-art traffic speed prediction framework is independent of fixed sensor applications. Unlike traditional systems that work depending on geographical locations, our framework enables predictions for areas that cannot be observed with fixed sensors and supports dynamic processing by collecting data from moving vehicles. This geographical independence significantly enhances the applicability of traffic estimation systems across large areas.
2. We present a dynamic prediction approach that allows for the selection of the road segment where the speed information will be estimated.
3. Motivated by privacy concerns in traffic speed prediction, we present a novel feature engineering approach that minimizes privacy issues by employing differential privacy. In order to address the privacy concerns associated with sensitive vehicle data, the proposed framework takes into account differential privacy in the engineering process. This approach maximizes data utility for accurate traffic predictions while protecting it from inference attacks. Additionally, this represents a new application of differential privacy in traffic speed estimation.
4. Unlike many studies based on publicly accessible datasets, this study evaluated a real-world dataset containing over 17 million GPS records collected from moving vehicles. This situation particularly demonstrates the reliability and authenticity of the findings obtained and reveals the potential of the proposed main framework to be used in real-time intelligent transportation systems.

The rest of this paper is organized as follows: Section II provides a summary of prior studies conducted in the field. Section III outlines the materials and methods employed in the study. Section IV introduces the proposed framework designed to address the identified problems. Section V presents the results obtained from the study. Section VI offers a comprehensive discussion of the findings. Finally, Section VII concludes the research by summarizing the key findings and their implications.

2 Related works

Traffic information prediction is primarily investigated with three indicators: flow, density, and speed. Some prior studies that consider these indicators are presented in the following subsections.

2.1 Flow prediction

Qi et al. [22] addressed privacy disclosure risks in centralized machine learning and proposed a blockchain-based federated learning approach for traffic flow prediction. They claimed that since the traditional federated learning approach utilizes a central model coordinator, it still faces security issues such as information disclosure. To address this problem, they proposed a blockchain-enabled federated learning approach operating in a decentralized manner. Additionally, they utilized differential privacy to protect the privacy of the model developed for traffic prediction. The PeMS traffic flow dataset was utilized, and experiments were conducted to evaluate different delta values and malicious vehicle scenarios. The proposed GRU-based prediction model provided better results than SAE, SVM, and LSTM-based models in terms of MSE, MAE, and RMSE metrics.

Akallouch et al. [8] proposed a federated learning model based on local differential privacy for privacy-aware traffic flow prediction. They underlined that sharing users' raw data with traditional machine learning methods poses significant risks as it may contain personal information. Hence, privacy-preserving measures should be implemented. The authors focused on privacy-aware machine learning-based traffic prediction in their research. Differential privacy was utilized to add noise to the gradients of the models developed on the user sides, and LSTM was used in a federated prediction model. They compared their model with centralized models like GRU, SAE, LSTM, and SVM. The PeMS dataset was utilized in the experiments, and MAE, MSE, RMSE, and MAPE metrics were used to evaluate performance. Finally, they stated that their proposed model provides satisfactory traffic prediction results while ensuring privacy.

Lonare and Bhramaramba [10] used federated learning with differential privacy to provide privacy-aware traffic prediction. The privacy of users' data was ensured in two steps. First, local differential privacy is applied to users' local data, and then, prediction is performed with federated learning. A graph convolutional network is used for traffic prediction, and experiments were conducted using the PeMS dataset. They compared the proposed model against ChebNet and GATNET, reporting MSE metric results for various clients. They found that their model achieved performance comparable to the others.

According to Xia et al. [23], traffic flow prediction is a challenging problem due to the complex spatial and temporal dynamics of dynamic changes. Furthermore, data privacy is a significant concern for participants. As a result, the authors proposed a model for predicting short-term traffic flow utilizing graph convolutional network and federated learning. Their model outperformed four

benchmarking models, including LSTM, CNN, ChebNet, and Graph Attention Network. They used the PeMS04 and PeMS08 datasets to compare the algorithms' MAE, MAPE, RMSE, and MAAPE metrics and obtained better results with their model.

The privacy concerns associated with vehicular ad hoc networks were the focus of Deng and Shen's study [24]. They emphasized that privacy protection measures should be implemented for traffic flow prediction because vehicular data may contain sensitive information about individuals, such as identity, location, and other personal details. As a result, the authors proposed a scheme that uses LSTM to forecast traffic flow and federated learning to preserve privacy. The experiments were conducted using the PeMS dataset, and afterward, the performance was assessed based on the loss function.

Although several studies based on deep learning models have shown remarkable effectiveness in predicting traffic flow, Liu et al. [18] noted that privacy is a crucial problem for preserving personal data. They cited some examples of real-world privacy breaches involving location data that occurred in the past when attempting to estimate traffic flow. Thus, realistic solutions that prioritize data privacy are essential. The authors proposed an algorithm for predicting traffic flow based on federated learning. The GRU algorithm was employed in a federated manner and compared with SAE, LSTM, SVM, and traditional GRU-based models. The PeMS dataset was used for the experiments, and performance measures, including MAE, MSE, RMSE, and MAPE metrics, were used to evaluate the results. Finally, their experiments yielded successful results with MAE of 6.85, MSE of 97.77, RMSE of 9.49, and MAPE of 14.69%.

Despite the fact that some deep learning models provide outstanding results for predicting traffic flows, Liu et al. [25] noted that privacy remains a significant challenge for these models. The authors focused on this issue and introduced the FedGRU model, which uses the GRU algorithm to make predictions. Performance evaluations were conducted using MAE, MSE, RMSE, and MAPE metrics on the PeMS dataset. Algorithms such as GRU, LSTM, SAE, and SVM were utilized to compare the proposed model, and the results of the experiments demonstrated that the proposed model produced superior outcomes compared to the others.

Using federated learning based on the spatial–temporal long- and short-term networks algorithm, Yuan et al. [26] predicted traffic flow. They underlined that the majority of studies focus on a centralized training models and fail to address privacy concerns related to distributed traffic data. To address these concerns, they implemented federated learning. In their experimental studies, they compared their model to HA, ARIMA, VAR, XGBoost, ConvLSTM, ST-ResNet, and DMVST-Net using RMSE, MAE, and MAPE metrics. Experimental studies were conducted using the TaxiNYC and TaxiBJ datasets. According to the results, the proposed model outperformed all the other models.

Liu et al. [27] focused on traffic flow prediction and proposed a novel method using transformer-based graph networks. They used the PeMS dataset and employed MAE, MAPE, and RMSE metrics for performance evaluators. The proposed method was compared with existing methods, and the results showed that it achieved lower error rates, with MAE of 13.90, RMSE of 23.31, and MAPE of 9.06%.

Xie and Jin [28] emphasized the importance of traffic flow in optimizing traffic resource allocation and enhancing efficiency of transportation systems. As a result, they focused on traffic flow prediction and compared several graph neural network algorithms' performances. They employed the PeMS dataset and compared eight different algorithms. MAE, MAPE, and RMSE are used as performance indicators. According to the experimental results, it has been obtained that MTGNN algorithm achieved the lowest error rates with MAE of 15.31, RMSE of 24.25, and MAPE of 10.21%.

Hu et al. [29] presented a novel federated learning-based graph convolution network for traffic flow prediction. They employed the PeMS dataset and used MAE and RMSE performance metrics. The proposed method was compared with ten existing algorithms, and the results showed that their method outperformed the others, achieving the lowest error rates, with MAE of 15.87 and RMSE of 23.78.

Miao et al. [30] proposed a new blockchain-based traffic prediction model that combines ARIMA, empirical mode decomposition (EMD), and singular value decomposition to address the challenges of nonlinear, unstable, and noisy traffic data in real-time applications for smart cities. In their approach, EMD is used for data stabilization, SVD is also used for noise reduction and ARIMA is used for accurate time series prediction. The proposed model achieved significantly lower error rates compared to traditional models. In particular, the proposed model produced MAE of 2.83 and RMSE of 4.62 scores. It has been observed that other models gave higher error rates in comparison. The findings obtained as a result of the experimental studies show that the proposed model is suitable both in terms of accuracy and computational efficiency and can be used in real-time traffic prediction.

Qi et al. [22] proposed a privacy-preserving framework that provides traffic flow prediction by combining federated learning and blockchain technology. The proposed model eliminates the need for a centralized server by using decentralized learning and uses a GRU network for traffic predictions. To ensure privacy, it uses a local differential privacy mechanism to protect the location data of vehicles. Experimental results show that the proposed model achieved superior performance compared to existing methods with MAE of 7.96, MSE of 101.49, and RMSE of 11.04. Additionally, blockchain-based verification enhances the reliability of the framework and even mitigates malicious attacks. The proposed model offers an innovative and practical traffic management system that supports policymakers in decision-making.

2.2 Density prediction

When dealing with traffic density prediction, Ren and Tang [31] emphasized the privacy concerns of car owners. The authors proposed a linear programming framework to protect geo-indistinguishability while detecting traffic density. The experiments utilized the San Francisco and T-Drive datasets. The proposed framework was compared against the S2M and RC algorithms. The outcomes have demonstrated that the proposed system achieved superior results compared to the alternatives with respect to the MAER metric.

Huang et al. [32] presented a privacy-preserving traffic density prediction model. They stated that crowdsourced tracking data can be used to estimate traffic density; however, this raises significant privacy leakage concerns. The authors employed homomorphic encryption to encrypt tracking data, while it was being transmitted, as well as local differential privacy to secure users' location information. A dataset obtained from the Didi-Gaia open data plan was used for the experiments. Finally, it was reported that the proposed model provided privacy and utility for traffic density estimation using the MAPE metric.

2.3 Speed prediction

Xia et al. [23] proposed a model for predicting short-term traffic speed by using graph convolutional network and federated learning. The model presented better results than the others, and they used the PeMS04 and PeMS08 datasets to compare the algorithms' MAE, MAPE, RMSE, and MAAPE results.

Predicting traffic speed was another area of focus for Zhang et al. [33]. According to the authors, the application of federated learning to intelligent transportation systems simply protects data privacy, but it ignores topological knowledge about the transportation system. To address this issue, they proposed a novel differential privacy-based adjacency matrix. For the purpose of predicting traffic speed, they employed attention-based spatial–temporal graph neural networks and federated learning. They used the PeMS dataset in the experiments to compare their approach to HA, ARIMA, LSVR, DCRNN, WaveNet, and STGCN algorithms. It was observed that their model outperformed the others in terms of RMSE, MAE, and MAPE metrics.

2.4 Other traffic studies

Hassija et al. [34] presented a system that uses deep learning and blockchain technologies to predict traffic congestion. This system allows drivers to securely share real-time traffic data using blockchain and rewards them with tokens. The shared data are then used to predict possible traffic congestion using a neural network structure and LSTM method. In doing so, it provides accurate predictions using both live data and historical traffic patterns. The results demonstrate that the proposed method outperforms traditional systems. Additionally, it encourages drivers to participate in the system, thereby improving the quality of traffic predictions.

In the study of Wang et al. [35], a decentralized system was proposed to collect real-time traffic data using blockchain technology. This system uses a two-layer blockchain architecture for efficiency and provides protection against various attack threats while also proposing a deep learning-based approach to predict traffic status. Privacy is ensured through cryptographic methods. Experimental studies show that the model is effective, offering an effective and secure structure as an alternative to traditional centralized traffic prediction systems. It is resistant to cyber-attacks and significantly enhances the accuracy of traffic data.

Corovic et al. [36] demonstrated the use of the YoloV3 algorithm for real-time detection of traffic participants. The study aimed to detect objects such as cars, trucks, passengers, traffic signs, and traffic lights in traffic with the help of cameras. By employing neural networks, the authors achieved high precision, particularly under varying conditions such as day, night, and different weather scenarios. The algorithm was developed on a customized dataset specifically designed for traffic detection, and the authors discussed potential enhancements, including the integration of other sensors and the use of larger datasets to improve accuracy. The results of the experimental studies show that this algorithm can be effectively applied in real time in autonomous driving systems.

Mhalla et al. [37] developed a system for multiple object detection and observation in traffic using deep learning. This system used the fast RCNN model and a customized framework to increase the success of object detection in changing traffic situations. In this study, they integrated transfer learning and customized framework for generic modeling in specific traffic situations. In particular, real-time detection of vehicles such as pedestrians, cars, buses, and motorcycles was achieved with high accuracy.

Chougule et al. [38] developed a novel system to reduce traffic congestion at intersections. This system uses federated learning and vertical data fragmentation methods to minimize excess fuel consumption, reduce idle waiting times, and increase vehicle flows. By combining real-time traffic and privacy-protected algorithms, the system has dynamically adjusted the signal timing to effectively pass vehicles at intersections. Test results demonstrated that the proposed method significantly increases vehicle throughput and is more effective than traditional traffic management systems.

Sri et al. [39] use the graph neural network structure in predicting traffic patterns. The aim of graph neural networks is to reveal complex relationships and dependencies in the data. Road networks are represented in the form of a graph, and traffic data are collected and processed. The model has been tested using various professions, and it has been seen that it gives accurate results in predicting traffic volumes on various road connections. This approach shows that the principled management of potential graph neural network structures will be facilitated, their effectiveness will be increased, and better route planning will be done thanks to signal control and congestion reduction.

To sum up, Table 1 highlights that past studies typically used open-source datasets and concentrated on general flow and density predictions. Only two studies focused on speed prediction. Xia et al. [23] primarily used federated learning making the adjacency matrix private by using differential privacy. Zhang et al. [33] emphasized short-term traffic prediction by employing GCN and federated learning on public datasets.

Our study, however, proposes a framework that employs privacy-preserving feature engineering and short-term traffic speed prediction. We tested our framework on a real-world dataset collected from moving vehicles. Data privacy is ensured through differential privacy, and predictions are made using RNN-based algorithms such as LSTM, BiLSTM, and GRU, as well as baseline algorithms like LR, SVR, MLP, and RF.

Table 1 Comparison of the previous studies (G.I: geo-independent)

Study	Focus	Time	Dataset	Privacy Measure	Algorithms	G.I	Metrics with min values
[22]	Flow	Short term, long term	PeMS	Federated learning, blockchain	GRU	No	MSE:101.49 MAE:7.96 RMSE:11.04
[8]	Flow	Short term	PeMS	Differential privacy	LSTM	No	MAE:8.03 MSE:103.24 RMSE:11.16 MAPE:18.76%
[10]	Flow	Short term	PeMS	Differential privacy	GCN	No	MSE:0.01 (test loss)
[23]	Flow	Short term	PeMS04, PeMS08	Federated learning	GCN	No	MAE:14.66 MAPE:11.44% RMSE:22.30 MAAPE:8.76%
[24]	Flow	Long term	PeMS	Federated learning	LSTM	No	Test loss
[18]	Flow	Short term	PeMS	Federated learning	GRU	No	MAE: 6.85 MSE: 97.77 RMSE: 9.49 MAPE: 14.69%
[25]	Flow	Short term	PeMS	Federated learning	GRU	No	MAE: 7.96 MSE: 101.49 RMSE: 11.04 MAPE: 17.82%
[26]	Flow	Long term	TaxiNYC, TaxiBJ	Federated learning	Spatial–temporal LSTM	Yes	RMSE: 9.32 MAE: 2.58 MAPE: 21.61%
[27]	Flow	Short term	PeMS	n/a	GNN	No	MAE: 13.90 RMSE: 23.31 MAPE: 9.06%
[28]	Flow	Short term	PeMS	n/a	MTGNN	No	MAE:15.31 RMSE: 24.25 MAPE: 10.21%
[29]	Flow	Short term	PeMS	Federated learning	GCN	No	MAE 15.87 RMSE 23.78
[31]	Density	n/a	T-Drive, San Francisco	Differential privacy	Linear programming	Yes	MAER:1.4%
[32]	Density	n/a	Didi-Gaia	Differential privacy	Classical density approach	Yes	MAPE:20.35%

Table 1 (continued)

Study	Focus	Time	Dataset	Privacy Measure	Algorithms	G.I	Metrics with min values
[23]	Speed	Short term	PeMS04, PeMS08	Federated learning	GCN	No	MAE:0.81 MAPE:1.47% RMSE:1.54 MAAPE:1.46%
[33]	Speed	Short term	PeMS	Federated learning, differential privacy	GNN	No	RMSE:6.42 MAE:3.03 MAPE:8.84%
This paper	Speed	Short term	Real-world Dataset	Differential privacy	LSTM, BiLSTM, GRU (Framework)	Yes	MAE: 3.1 MAPE: 5.40% MSE: 12.4 RMSE: 3.4

Table 2 The features and definitions for dataset

Feature	Definition
Vehicle ID	A unique ID that belongs to a vehicle using the navigation system
Time stamp	Time information in date and time format
Y	Longitude coordinate
X	Latitude coordinate
Vehicle Speed (VS)	Instantaneous measured speed of the vehicle
Angular Direction (AD)	The direction of the vehicle in the coordinate plane with 0° pointing north way

Table 3 Sample dataset records

Vehicle ID	Time stamp	Y	X	VS	AD
fce*****	16.06.2023 15:00	39.954361	32.571415	35	40
fcf*****	16.06.2023 15:00	39.902203	32.765202	50	171
84C*****	16.06.2023 15:00	39.920960	32.993813	101	236
129*****	16.06.2023 15:00	39.894039	32.803215	49	270
cb5*****	16.06.2023 15:00	39.802967	32.393925	57	202

3 Material and methods

In this section, the dataset collected and used in the experiments, the privacy-preserving feature engineering technique, and the proposed framework are explained.

3.1 Used dataset

In this study, a daily dataset collected from vehicles moving through Ankara on June 16, 2023, was used. The raw vehicle dataset, which consists of GPS signals, is approximately 1.2 gigabytes and contains 17 million records. The GPS signals are stored at varying time intervals, and the records do not have any frequency. The signals produced by vehicles have six features, as outlined in Table 2, with their definitions provided.

A sample of the dataset is shown in Table 3. The vehicle data used in the study were obtained from validated navigation devices installed in the vehicles of the companies contracted by Başarsoft Information Technologies Inc., and the variety of these vehicles cover all kinds of motor vehicles from private cars to cargo delivery vehicles. When analyzing the vehicle data, micro-level data (driving time, distance traveled, etc.) of each vehicle were not analyzed, but macro-level attributes, such as speed, position, and angular direction, were taken into account as shown in Table 3.

3.2 Privacy-preserving feature engineering

Feature engineering is an important part for a research handling a dataset [40]. This section describes how the features are handled with respect to privacy. To achieve this, we used the OPTICS clustering algorithm to cluster the vehicles moving in the same direction and then applied differential privacy to the average speed of each cluster to preserve privacy. In this section, first, the algorithms are explained, followed by a detailed explanation of the privacy-preserving feature engineering technique.

3.2.1 OPTICS

The OPTICS is a density-based algorithm evolved from the DBSCAN algorithm, and it locates the neighborhood of points based on the cardinality measurement [41]. In the DBSCAN algorithm, the "p" point is randomly selected as the core object, and the proximity radius epsilon (ε) for the distance to the neighbors and MinPts for how many members are around the core points are used as input parameters. In the OPTICS algorithm, "produced distance" reckons with ε_i values, which are lower than ε value. The difference between OPTICS and DBSCAN is that the set members are not determined in the first step. Instead of determining the set members immediately, the order of all objects is created, and the core distance and the appropriate accessible measurement information are stored. Clustering is performed by considering the least number of close neighbors for the objects determined as core objects among the objects whose data order is obtained. In Eq. (1), the formula for determining the core distance can be seen. While the neighborhood of a "p" point at ε distance is expressed as $N_\varepsilon(p)$, it is considered undefined if it is less than the MinPts value. If this distance is equal to and greater than the minimum point, the distances to the neighboring points are considered MinPts-distance, and the nearest neighbor point is chosen as the next seed point (Ester et al. [42]).

$$\text{distance to the neighbor point} \begin{cases} \text{if Cardinality } N_\varepsilon(p) < \text{MinPts, undefined} \\ \text{other conditions, MinPts - distance } (p) \end{cases} \quad (1)$$

The OPTICS algorithm is used to cluster the vehicles moving on the road according to predefined segments. After applying the OPTICS algorithm, vehicle clusters are formed, enabling the extraction of specific traffic information.

3.2.2 Differential privacy

Differential privacy, proposed by Dwork [43], enables privacy-preserving operations on raw datasets. The privacy method ensures that every query on a dataset returns the same results regardless of whether a record is included in the dataset or not. This means altering any record in any way will not affect how a query is answered. This method is used to prevent background information

attacks. Differential privacy is a widely used technique for preserving privacy and addressing such statistical leakages. A formal definition is provided below.

Definition: a randomized mechanism M satisfies ε-differential privacy (ε-DP) if for any adjacent databases D_1 and D_2, and any subsets S of all possible outputs;

$$Pr\left[M\left(D_1\right) \in S\right] \leq \exp(\varepsilon) * \Pr[M(D_2 \in S] \tag{2}$$

A query on a dataset has its sensitivity. Sensitivity is the degree to which the outcome of a query changes depending on whether a record appears or not. For instance, a "COUNT" query has a sensitivity of 1.

Definition: The sensitivity of a query f can be defined as in Eq. (3);

$$\Delta f = \max_{D_1 D_2} \|f\left(D_1\right) - f\left(D_2\right)\| \tag{3}$$

Definition: For a function f, following mechanism M satisfies ϵ-DP;

$$M(D) = f(D) + Laplace\left(\frac{\Delta f}{\epsilon}\right) \tag{4}$$

Differential privacy is categorized into global and local privacy models [44]. In global differential privacy, a trusted curator collects data and then provides perturbed answers to untrusted third parties' queries. However, local differential privacy perturbs data on the data owner side and then releases perturbed data to untrusted third parties or servers. This paper focuses on global differential privacy by employing output perturbation. As stated in [45], output perturbation satisfies global differential privacy.

We used differential privacy to obtain noisy cluster speed results. Specifically, clustering produces the clusters of moving objects. Hence, since our paper proposes predicting traffic speed in a private manner, we apply differential privacy to the average speed values of the clusters.

3.2.3 Privacy-preserving feature engineering approach

As mentioned before, our approach enables us to obtain differentially private features. Since we focus on predicting traffic speed information, our approach handles speed features to ensure their privacy. The problem definition is provided below.

Given the emphasis of this paper on traffic speed prediction, privacy-preserving measures are applied to speed data to ensure privacy. Consider an adversary with prior knowledge who is capable of executing an inference attack on a traffic dataset through the utilization of varying outputs for specific queries. In addition, a trusted curator has a dataset X_1, and an adversary has a dataset X_2, which are called adjacency datasets and include the speed data of the users. It is known that X_1 and X_2 differ only in one row, such as $\left\{x_0\right\} = X_1 - X_2$. Assuming the adversary applies an average query on X_1 and gets the average values. Then, the adversary applies the same query on X_2 to obtain the average values. After all, when the adversary compares the two values, missing record x_0 can be easily inferred [46]. An illustrative representation of the issue addressed in this paper is provided in Fig. 1. Consider that an adversary has the knowledge of

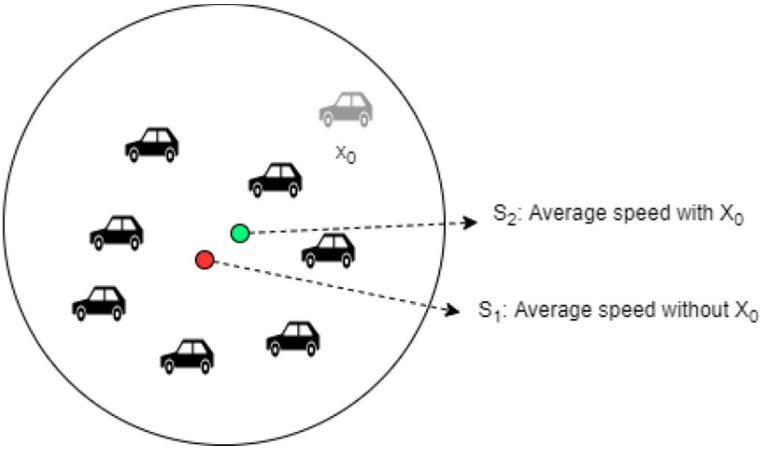


Fig. 1 An example of inference attack for speed prediction

average values S_1 and S_2 and the number of items N_i in the database. The adversary can infer x_0 as presented in Eq. (5),

$$x_0 = N_i S_1 - \left(N_i - 1\right) S_2 \tag{5}$$

To protect the individuals in a dataset from the inference attack illustrated in Fig. 1, the actual value of the average speeds generated by an average query must be protected. Otherwise, any inference attack could reveal the actual values of individuals. In the proposed framework, the speed data are subjected to output perturbation using differential privacy techniques to ensure both user privacy and privacy-aware feature engineering.

3.3 Error metrics

MAE, MAPE, MSE and RMSE are error metrics frequently used to evaluate the performance of applications developed to predict traffic parameters [47]. These error rates are calculated by comparing the difference between each predicted value ($\widetilde{y}_i$) and the actual value (y_i) for an array with n elements. The calculations for MAE, MAPE, MSE, and RMSE are performed according to Eq. (6–9) [48, 49].

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\mathrm{y_i} - \widetilde{y}_i\right| \tag{6}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|\mathrm{y_i} - \widetilde{y}_i\right|}{\mathrm{y_i}} \tag{7}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widetilde{y}_i\right)^2 \tag{8}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \left(y_i - \widetilde{y}_i\right)^2}{n}} \tag{9}$$

3.4 Haversine distance

The Haversine distance measures the greatest circle around the distance between two points on a spherical plane [50]. Since the shape of the earth is a flattened sphere, the distance between the GPS points is calculated using the Haversine distance, as shown in Eq. (10) as indicated in [51]. The formula shows the distance calculation as "d", the earth radius as "r", latitude values as θ, and longitude values as φ.

$$d = 2r \cdot \arcsin\left(\sqrt{\sin^2\left(\frac{\theta_2 - \theta_1}{2}\right) + \cos\left(\theta_1\right) \cdot \cos\left(\theta_2\right) \cdot \sin^2\left(\frac{\varphi_1 - \varphi_2}{2}\right)}\right) \tag{10}$$

4 Proposed framework

In this paper, we proposed a state-of-the-art, privacy-preserving, feature engineering-based traffic speed prediction framework for ITS. The proposed framework, which focuses on speed prediction while ensuring privacy, consists of six stages. In the first stage, data are read and filtered, and GPS points of selected segment are matched and placed on the map. In the second stage, the direction is selected for the vehicle data specified on the map. These vehicles are clustered based on density, and data cleaning is performed for unlabeled vehicle data (labeled as -1) and vehicles with an exceeding standard deviation in the third stage. In the fourth stage, privacy-preserving feature engineering was applied to the speed values of each vehicle cluster. By applying differential privacy to the average speed data of each vehicle cluster, one of the attributes is created for the vehicle clusters in a private manner. In the fifth stage, a flow is created according to the distance calculated by the mean of the coordinates of the vehicle clusters with respect to a specific starting point. Finally, the features of the clusters are given as inputs to the prediction model, and privacy-preserved speed prediction of the road is performed in the last stage.

The proposed framework was implemented using Python 3 on a computer with 16 GB RAM, an Intel Xenon 3.4 GHz processor, and libraries such as NumPy, Pandas, and Matplotlib. The general structure and workflow of the proposed privacy-aware speed prediction framework are shown in Fig. 2, and a general pseudocode of the framework is provided in Algorithm 1, which is explained in detail above. In addition, each stage is explained in detail in the following sub-headings.

Algorithm 1: The general pseudocode of the proposed framework

Input: data $X = \{x_1, x_2, \ldots, x_N\}$, selected road segment rs

Output: predicted speed values $Sp = \{s_1, s_2, \ldots, s_M\}$

for x_i in X:
  if x_i belongs to rs:
    append x_i to X_{rs}
for x_j in X_{rs}:
  if x_j is between time intervals of t_1 and t_2:
    append x_j to X_{rs_t}
for x_k in X_{rs_t}:
  if coordinate of x_k matches with SHP file:
    if angular direction of x_k between 30 and 170:
      append x_k to $X_{istanbul}$
    if angular direction of x_k between 220 and 270:
      append x_k to $X_{eskisehir}$
    if angular direction of x_k between 160 and 220:
      append x_k to X_{konya}
for each data in $\{X_{istanbul}, X_{eskisehir}, X_{konya}\}$:
  clusters = OPTICS (data)
  for each cluster c_i in clusters C = $\{c_1, c_2, \ldots, c_L\}$:
    for each element e_i in c_i
      if angular direction of $e_i > std(angular)$:
        remove noise e_i
  for each cluster c_i in clusters C = $\{c_1, c_2, \ldots, c_L\}$:
    $f = \{c_{speed}^{private}, c_{frontXY}, c_{rearXY}, c_{length}, c_{density}, c_{numberofvehicles}\}$; extract features
  X_{flow} = *sort* (C, f); according to the proximities to a defined starting point
  X_{flow_t} = *transform* ($X_{flow}, window_{size}$); transform time series to supervised learning
  $X_{flow_t}Train$, $X_{flow_t}Test$ = *split_data* (X_{flow_t}); split for train and test datasets
  Model = *train* ($X_{flow_t}Train$); create any regression algorithm and train with train dataset
  $Sp = predict(X_{flow_t}Test)$; test the model with test dataset
  Obtain Sp

4.1 Data input and filtering

The data utilized in this research were acquired from Başarsoft Information Technologies Inc. and consist of one-day Ankara data, with a size of 1.2 gigabytes, encompassing approximately 17 million vehicle records from June 2023. The data

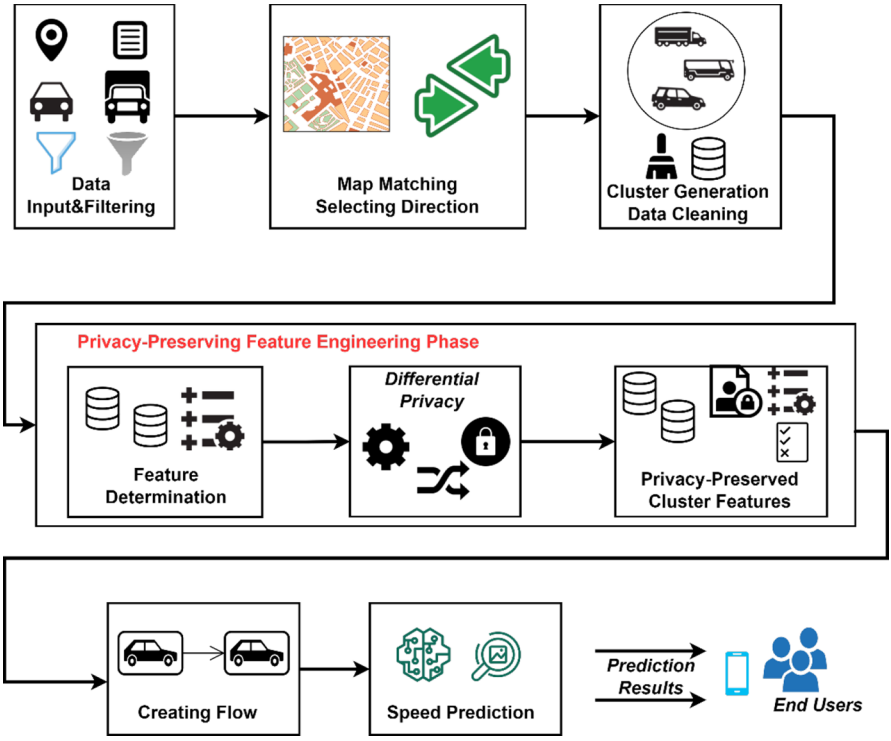


Fig. 2 The proposed privacy-preserving speed prediction framework

include six features: vehicle ID, time stamp, latitude–longitude coordinates, instantaneous speed, and angular direction information. In order to place the read data on the desired road segment, the vehicle data were selected for fifteen minutes to make a short-term prediction. Afterward, these data were filtered regionally on the Istanbul Road, Eskisehir Road, and Konya Road, which are accepted as the main arteries for the province of Ankara. During the filtering process, two specific points, determined separately for each road and whose latitude–longitude information is shown in Table 4, were selected. The vehicle data were filtered within the "rectangular regions" shown in Fig. 3. This filtering process helps to obtain the target region for subsequent analysis.

After preprocessing such as filtering, direction selection, and road buffering, the raw dataset is reduced to 10,216 rows of records for Istanbul Road, 5,776 rows of records for Eskisehir Road, and 4,337 rows of records for Konya Road, respectively, and each file size is approximately 0.5–1 MB.

4.2 Map-matching and selecting direction

After determining the target region and prediction range, a map-matching procedure was performed on the relevant vehicle data. A Shapefile with the SHP extension

Table 4 Coordinate information on selected regions including main arteries

Selected regions including	First point		Second point	
	Longitude	Latitude	Longitude	Latitude
Istanbul Road	32.6500	39.9440	32.8353	39.9635
Eskisehir Road	32.6386	39.8678	32.8120	39.9127
Konya Road	32.8080	39.8500	32.8335	39.9490



a) Istanbul Road b) Eskisehir Road c) Konya Road

Fig. 3 Location points of filtered vehicle data

contains geometric shapes and was used for map-matching. Each GPS point was buffered and matched with the selected SHP file using the location index of the SHP files. A sample of SHP file details of Konya Road, which consists of Mevlana Boulevard, is given in Table 5, and the SHP files of the main arteries are shown in Fig. 4.

Matching the GPS points on the map is a preprocessing step for determining the route in the proposed framework. This process was performed for the Istanbul, Eskisehir, and Konya Roads in Ankara province, all of which have two-way traffic flow. Since the vehicle data mapped to the road include movements in two directions, an additional selection process was conducted, considering the angular direction of vehicles traveling in the same direction. The angular direction of the vehicles moving eastward or southeastward on the Istanbul Road falls between 30 and 170°, those moving westward or southwestward on the Eskisehir Road are between 220 and 270°, and vehicles moving southward on the Konya Road fall between 160 and 220°. Based on these values, angular direction filtering was applied to select vehicles traveling southeastward on Istanbul Road, southwestward on Eskisehir Road, and southward on Konya Road. This filtering ensures that only one-way vehicles are included in the model.

4.3 Cluster generation and data cleaning

After the map-matching process and selection of vehicle directions for the target road, the vehicle data were clustered using the density-based clustering method OPTICS, taking into account the latitude and longitude information. The minimum number of vehicles required for clustering was set to 20, though this parameter can be adjusted as needed. Unlike DBSCAN, OPTICS does not require a predefined minimum distance value for clustering. Instead, it clusters vehicles based

Table 5 Sample demonstration of the SHP file

NETWORK_ID	GEOLOC	NETWORK NAME	LENGTH
6005****	LINESTRING (32.81012 39.863739, 32.81018 39.86…	Mevlana Blvd	1140
6011****	LINESTRING (32.809455 39.85838802, 32.80991796…	Mevlana Blvd	617
6009****	LINESTRING (32.80920984 39.85751403, 32.809296…	Mevlana Blvd	99
6004****	LINESTRING (32.80947192 39.85368597, 32.809259…	Mevlana Blvd	429
6039****	LINESTRING (32.811012 39.850026, 32.810899 39…	Mevlana Blvd	137

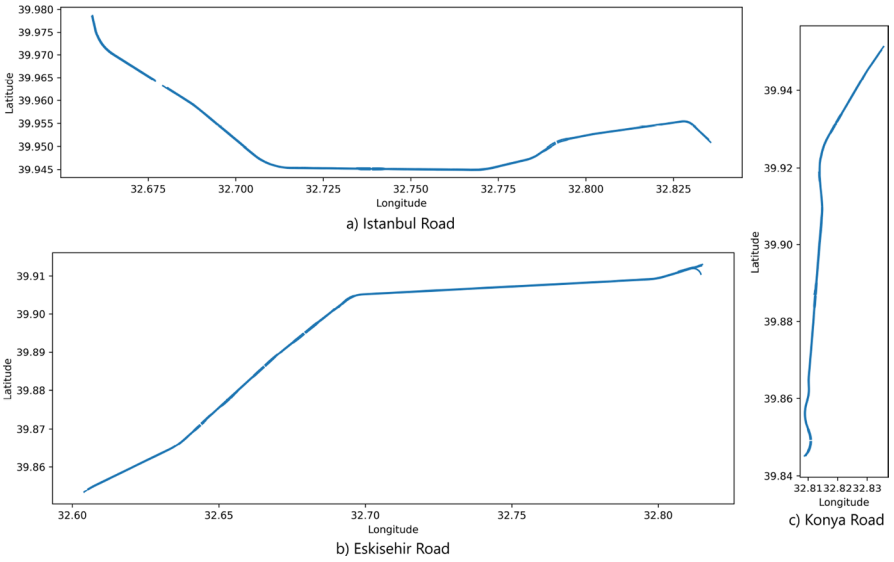


Fig. 4 SHP file of main arteries

on their probability of belonging to a cluster. This approach allows vehicle data to be grouped into clusters without a predefined minimum distance threshold, thereby forming the road features as intended in the framework's design.

As a result of the density-based clustering process, non-clustered GPS records are marked as "-1" and subsequently removed from the dataset. Additionally, it was found that some vehicles providing GPS signals at intersections were traveling in various directions and on different parts of the road. Therefore, vehicles with the highest standard deviation in angular direction (AD) were excluded from the clusters. The standard deviation and mean calculations for the AD value are shown in Eqs. (11) and (12), where "n" represents the number of vehicles in a given cluster.

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(AD_i - \overline{AD}\right)^2} \tag{11}$$

$$\mu = \frac{1}{n}\sum_{i=1}^{n} AD_i \tag{12}$$

4.4 Privacy-preserving feature engineering

In the proposed framework, the features of each cluster were extracted to create the path characteristic. The key features include cluster speed and front-rear vehicle coordinate information. Cluster speed is utilized for speed prediction after applying differential privacy, while front-rear vehicle coordinate information is used for color map illustrations. Additionally, the condition of clusters was examined using additional features.

After clustering, differential privacy noise is applied to the average speed values of each cluster. The noise is calibrated based on a predefined privacy budget to ensure a strong privacy guarantee while preserving the utility of the aggregated data. The noise-added cluster averages are then used as features for training the traffic prediction model, ensuring that sensitive information about individual vehicles is never exposed.

The privacy-aware feature extraction process is detailed in Algorithm 2. For each cluster, the number of clusters, differentially private average cluster speed, geographical location of the front and rear vehicles, the length and density of each cluster are presented in Table 6 along with their explanations.

Algorithm 2: The pseudocode of the privacy-aware feature extraction process

Input: clusters C	
Output: extracted features f	
1	for each c_i in C:
2	$c_{numberofvehicle}$ = number of vehicles in c_i
3	$c_{frontXY}$ = front vehicle coordinates of c_i
4	c_{rearXY} = rear vehicle coordinates of c_i
5	$c_{speed}^{private}$ = perturbed average speed of the vehicles in c_i
6	c_{length} = Haversine ($c_{frontXY}$, c_{rearXY})
7	$c_{density} = \frac{c_{length}}{c_{numberofvehicle}}$
8	$f = \{c_{numberofvehicles}, c_{frontXY}, c_{rearXY}, c_{speed}^{private}, c_{length}, c_{density}\}$
9	obtain f

Table 6 Generated cluster features

Feature Name	Explanations
Number of vehicles in a cluster $(c_{numberofvehicle})$	The sum of the signals produced by each vehicle in a cluster
Differentially private average cluster speed $(c_{speed}^{private})$	The ratio of the total speed to the number of vehicles in a cluster, calculated in a private manner
Front vehicle X (c_{frontX})	Minimum X (Longitude) value from vehicles moving from West to East with Min $(X_{Cluster})$
Front vehicle Y (c_{frontY})	Y (Latitude) value of the vehicle with Min $(X_{Cluster})$
Rear vehicle X (c_{rearX})	Minimum X (Longitude) value from vehicles moving from East to West with Max $(X_{Cluster})$
Rear vehicle Y (c_{rearY})	Y (Latitude) value of the vehicle with Max $(X_{Cluster})$
Length of cluster (c_{length})	Haversine distance between front vehicle and rear vehicle
Density of cluster $(c_{density})$	The ratio of cluster length to the number of vehicles in the cluster

In Fig. 5, clusters with their average speeds are illustrated. Each icon represents a cluster, with the speed values shown in red. The average speed values of each cluster have been subjected to differential privacy. The cluster's average speed values were calculated as the ratio of the total speed of the vehicles in a cluster to the number of vehicles, as detailed in Table 6. Output perturbation was applied to generate noisy average speeds. Figure 6 illustrates the perturbed cluster speed values for the same clusters in Fig. 5. Additionally, the amount of perturbation applied to the speed values of each cluster was adjusted within the predefined epsilon values. Finally, the feature values processed using differential privacy are ready to form a flow from a starting point and establish a traffic flow. The features are now prepared to create traffic flow for speed prediction in a private manner.

4.5 Creating flow

Flow creation is a process based on a predetermined starting point and a final point for privacy-aware prediction. At this stage, the formed clusters are sorted according to their proximity to the starting point. For our experimental work, three intersections were selected as starting points: Asti junction (32.8122E, 39.9123N) for traffic traveling west on the Eskisehir Road, Ankara-Ayas Road junction intersection (32.6786E, 39.9634N) for traffic traveling southeast on the Istanbul Road, and Kepekli junction (32.8109E, 39.8496N) for traffic traveling north on the Konya Road. Haversine distance was used to determine the distance from the starting point of each vehicle cluster moving in the specified direction.

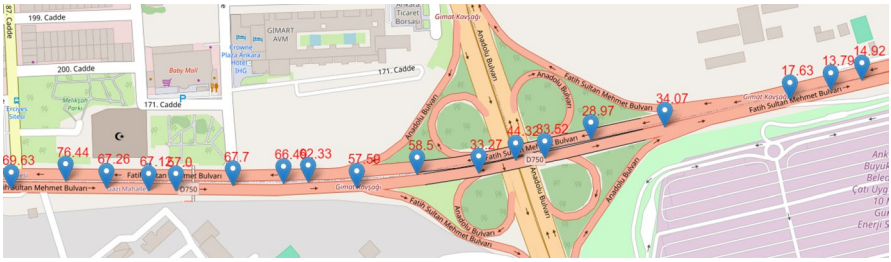


Fig. 5 An example of extracted clusters with average speed values



Fig. 6 Private average speed values of the clusters are presented in Fig. 5

4.6 Speed prediction

The proposed framework supports the use of various regression algorithms. Since deep learning-based regression algorithms provide superior results, we employed RNN variations, including LSTM, BiLSTM, and GRU models in the traffic speed prediction phase. Additionally, we utilized traditional regression algorithms such as LR, SVR, MLP, and RF. The differentially private speed values of the vehicle clusters, sorted according to the created flow, were used as input values for the speed prediction algorithm. MAE, MAPE, MSE and RMSE error metrics were employed to compare the predicted speed values with the original values.

5 Experimental results

Our privacy-aware traffic speed prediction framework was tested on three main arteries in the province of Ankara: Istanbul, Eskisehir, and Konya Roads. The vehicle traffic data, read in the first stage, were filtered to cover the specified road segments and mapped to the relevant roads using shape geometric files. For vehicles traveling on the two-way roads, only those moving in the specified directions, as described in Sect. 4.2, were selected. The mapped and selected data for the main arteries are illustrated in Fig. 7.

After selecting and mapping the filtered data, vehicle clusters were generated using the OPTICS algorithm. Some GPS points were cleaned and labeled as -1

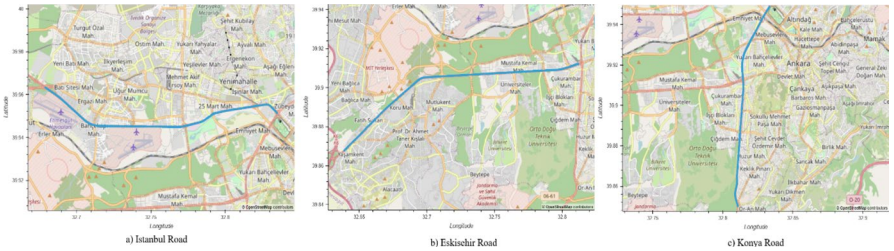


Fig. 7 Mapped data of three main arteries



Fig. 8 Obtained clusters on the roads

(indicating that they do not belong to any cluster) if they were outside the standard deviation of the speed. The distribution of the clusters on the roads is presented in Fig. 8.

Features such as cluster speed and vehicle count are extracted when obtaining pure data clusters. The key step here is the application of differential privacy to ensure that the speed feature is protected. By ensuring the feature's privacy, the dataset can be analyzed without compromising individual privacy. From the cluster characteristics, the number of vehicles in each cluster and the cluster's average speed are shown in Fig. 9 for each road. For instance, when the Istanbul Road graph is examined, there are over 60 clusters with a vehicle count ranging between 20 and 30, while approximately 10 clusters have an average speed below 20 km/h. Similarly, in the Eskisehir Road graph, over 35 clusters have a vehicle count between 30 and 40, but only a few clusters show an average speed below 10 km/h.

After obtaining privacy-preserved features, a flow was created by sorting the clusters according to a starting point to define the road characteristics. The road speed map was then visualized using the private average speed data of clusters as inputs for the prediction algorithm, and a privacy-aware speed prediction was performed. The traffic speed colors created by the "privacy-preserved cluster speed data" on the roads are marked in blue for smooth traffic above 60 km/h, yellow for medium density between 25 and 60 km/h, and red for heavy traffic below 25 km/h. The resulting visualization of the road maps is depicted in Fig. 10.

When the graphs colored with respect to the privacy-preserved speeds on the map are scrutinized, while the traffic flowed well in the east on the Istanbul Road, there was a transition from medium density to heavy traffic, starting

from the middle of the road at a major intersection. From the easternmost point, the departure point, vehicle clusters going in the direction of the west created high traffic on the Eskisehir Road and fluent-medium heavy traffic was obtained toward the end of the road. The traffic on the route was frequently heavy to moderately heavy when moving in the south along the Konya Road, as seen by the speed graph.

The final step of the proposed framework involves using privacy-preserved cluster speed values to predict traffic speeds. To test the framework, we utilized LSTM, BiLSTM, GRU, LR, SVR, MLP, and RF algorithms for prediction. Each RNN-based model was built with a two-layered architecture containing 256 and 128 neurons. The "rmsprop" optimizer was employed, as it performs well in situations with abrupt gradient changes. The model was trained for 100 epochs with a training–test ratio of 80–20%. Hyperparameter tuning was performed via grid search. The error values obtained as a result of the model are provided in Tables 7, 8, 9 and 10 and illustrated in Figs. 11, 12, 13, 14. Additionally, Fig. 15 shows the comparison of the speed prediction values of the test set with the actual speed values of the cluster.

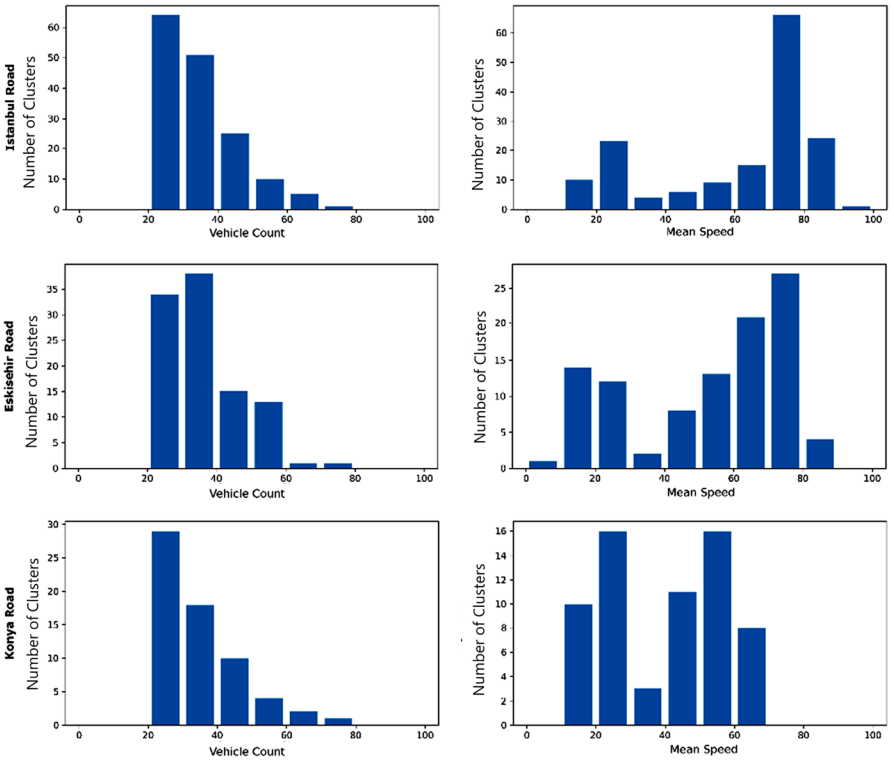


Fig. 9 Cluster average speed and number of vehicles in the clusters

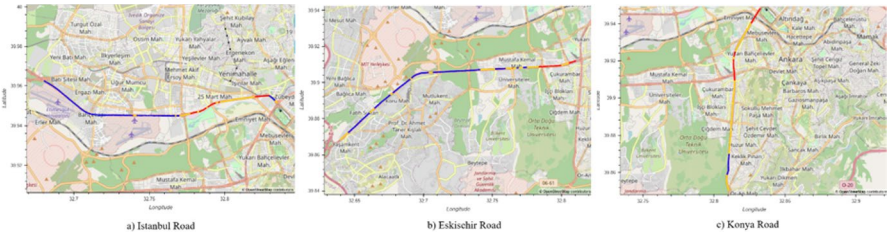


Fig. 10 Visualization of the road speed maps

The MAE, MAPE, MSE, and RMSE error rates for LSTM, BiLSTM, and GRU are comparable when the error values in Tables 7, 8, 9 and 10 and the graphs in Figs. 11, 12, 13 and 14 are considered. In addition, the results of these algorithms are lower than classical algorithms such as LR, MLP, SVR, and RF. In Fig. 15, when the graphs for each road are reviewed individually, it is observed that the slope of the graph for Istanbul Road shows a regular increase, particularly after the 15th cluster. The estimations made by LSTM, BiLSTM, and GRU closely match this slope.

There was a steady transition from the 2nd cluster to the 18th cluster on Eskisehir Road, except for the 8th cluster. However, after this point, traffic suddenly changed, with the average speeds of the clusters increasing sharply, resulting in smooth traffic flow. This sudden change significantly increased the prediction error and introduced instability. A similar situation occurred after the 27th cluster on Istanbul Road. These sudden changes caused a notable increase in the MAPE values, by approximately 20–30%, on both roads.

The most successful measurements were observed on the Konya Road, where the MAE and MSE values for the LSTM, BiLSTM, and GRU methods were approximately 3 km/h. These successful results on the Konya Road can be attributed to the number of clusters formed and the speed changes progressing in a consistent order.

6 Discussion

Traffic flow, density, and speed parameters, which are essential indicators for traffic management systems, have long been the focus of research. These parameters serve as critical data sources for various applications, including routing traffic for different user types ranging from experts to end users and generating optimal routes. Traffic parameters are analyzed at different city scales and over varying time intervals to support these applications.

When the literature is surveyed, various points can be discussed, such as:

- Methods, traffic parameters, and error rates: Liu et al. [18] achieved minimum error rates for flow prediction. The MAE, MSE, RMSE, and MAPE results of the proposed method are obtained as 6.85, 97.77, 9.49, and 14.69%, respec-

Table 7 MAE test results of the framework for different algorithms and epsilon values (95% confidence int.)

Road	EPS	LR	SVR	MLP	RF	LSTM	GRU	BiLSTM
Istanbul	100	13.695 ± 3.151	15.344 ± 2.228	13.731 ± 3.168	15.269 ± 3.415	8.345 ± 1.620	7.731 ± 1.451	8.655 ± 1.821
	200	13.645 ± 3.209	15.186 ± 2.368	13.767 ± 3.378	14.192 ± 3.114	8.853 ± 1.514	7.718 ± 1.265	9.052 ± 1.299
	300	14.065 ± 3.372	15.336 ± 2.473	15.126 ± 4.432	14.894 ± 3.119	8.984 ± 1.699	7.965 ± 1.456	9.166 ± 1.521
Eskisehir	100	10.910 ± 0.093	10.480 ± 0.176	11.376 ± 0.200	14.051 ± 1.460	10.130 ± 0.054	10.084 ± 0.051	10.172 ± 0.055
	200	10.749 ± 0.018	10.863 ± 0.183	10.897 ± 0.180	11.591 ± 0.648	10.008 ± 0.032	9.768 ± 0.031	10.592 ± 0.057
	300	10.798 ± 0.098	10.496 ± 0.109	10.471 ± 0.076	12.186 ± 1.690	10.005 ± 0.042	9.824 ± 0.041	10.295 ± 0.052
Konya	100	7.564 ± 2.128	7.646 ± 2.136	7.494 ± 2.038	8.407 ± 2.255	3.435 ± 0.841	3.227 ± 0.770	3.696 ± 0.926
	200	7.590 ± 2.006	7.806 ± 2.187	7.565 ± 2.166	7.938 ± 2.319	3.118 ± 0.856	3.105 ± 0.751	3.160 ± 0.781
	300	7.413 ± 2.159	7.728 ± 2.212	7.390 ± 2.112	8.349 ± 2.225	3.237 ± 0.947	3.165 ± 0.941	3.379 ± 0.964

Table 8 MAPE test results of the framework for different algorithms and epsilon values (95% confidence int.)

Road	EPS	LR	SVR	MLP	RF	LSTM	GRU	BiLSTM
Istanbul	100	48.464 ± 7.394	61.620 ± 8.499	56.833 ± 12.618	53.376 ± 7.975	30.372 ± 7.373	28.722 ± 7.359	31.391 ± 7.476
	200	48.870 ± 7.786	61.807 ± 8.368	57.187 ± 14.868	50.500 ± 7.400	32.232 ± 8.226	28.584 ± 7.101	32.909 ± 8.118
	300	50.342 ± 8.016	62.804 ± 9.352	63.842 ± 15.773	54.740 ± 8.565	32.375 ± 8.785	28.967 ± 7.119	32.540 ± 7.141
Eskisehir	100	22.941 ± 2.711	22.892 ± 2.311	23.390 ± 2.172	25.641 ± 2.715	18.546 ± 1.488	17.642 ± 1.429	18.465 ± 1.619
	200	22.992 ± 2.566	22.988 ± 2.194	22.900 ± 2.190	23.299 ± 2.300	18.737 ± 1.225	17.220 ± 1.180	18.750 ± 1.282
	300	22.158 ± 2.027	22.933 ± 2.485	23.765 ± 2.495	23.185 ± 2.541	18.558 ± 1.008	17.287 ± 1.002	18.308 ± 1.420
Konya	100	13.406 ± 4.790	13.802 ± 5.073	13.308 ± 4.406	15.156 ± 5.923	5.998 ± 1.656	5.647 ± 1.598	6.333 ± 2.061
	200	13.410 ± 4.958	14.078 ± 5.723	13.398 ± 5.170	14.246 ± 5.901	5.505 ± 1.443	5.409 ± 1.736	5.416 ± 1.747
	300	13.148 ± 4.916	13.966 ± 5.754	13.141 ± 4.254	15.120 ± 5.872	5.600 ± 1.725	5.563 ± 1.651	5.974 ± 2.015

Table 9 MSE test results of the framework for different algorithms and epsilon values (95% confidence int.)

Road	EPS	LR	SVR	MLP	RF	LSTM	GRU	BiLSTM
Istanbul	100	267.672 ± 49.041	275.491 ± 33.713	322.063 ± 81.110	327.245 ± 56.930	158.783 ± 42.879	134.913 ± 33.374	163.931 ± 42.275
	200	269.264 ± 42.209	275.864 ± 36.495	344.215 ± 83.381	279.665 ± 50.794	178.046 ± 33.578	132.735 ± 32.582	180.727 ± 39.763
	300	289.562 ± 58.384	284.552 ± 37.923	401.960 ± 110.278	300.361 ± 49.506	185.262 ± 44.964	148.990 ± 40.267	162.607 ± 42.005
Eskisehir	100	262.492 ± 36.491	246.328 ± 28.244	260.283 ± 35.307	300.981 ± 35.965	208.616 ± 7.060	206.247 ± 3.732	210.160 ± 4.145
	200	272.959 ± 39.200	234.611 ± 29.426	267.947 ± 38.209	243.644 ± 38.156	197.651 ± 4.745	177.455 ± 4.105	200.732 ± 4.130
	300	260.724 ± 36.911	238.643 ± 27.078	266.435 ± 32.188	259.364 ± 28.632	199.721 ± 7.456	183.023 ± 6.506	202.155 ± 8.900
Konya	100	74.152 ± 35.836	81.843 ± 44.012	81.312 ± 40.322	96.138 ± 48.545	14.011 ± 5.307	13.350 ± 5.157	17.620 ± 8.423
	200	77.226 ± 41.336	85.203 ± 47.201	79.425 ± 40.564	89.632 ± 49.982	12.579 ± 5.310	12.408 ± 4.967	12.804 ± 5.010
	300	73.834 ± 37.933	84.431 ± 46.854	77.594 ± 38.857	94.633 ± 48.363	13.903 ± 5.580	13.278 ± 5.495	14.324 ± 7.907

Table 10 RMSE test results of the framework for different algorithms and epsilon values (95% confidence int.)

Road	EPS	LR	SVR	MLP	RF	LSTM	GRU	BiLSTM
Istanbul	100	16.361 ± 1.096	16.598 ± 1.580	17.946 ± 1.517	18.090 ± 1.149	12.601 ± 0.955	11.615 ± 0.861	12.804 ± 1.814
	200	16.409 ± 1.236	16.609 ± 1.212	18.553 ± 1.673	16.723 ± 1.984	13.343 ± 0.193	11.551 ± 0.099	13.443 ± 1.049
	300	17.017 ± 1.420	16.869 ± 1.310	20.049 ± 1.494	17.331 ± 1.900	13.611 ± 1.252	12.206 ± 1.167	12.752 ± 1.784
Eskisehir	100	17.502 ± 1.187	17.695 ± 1.093	17.833 ± 1.084	17.349 ± 1.048	14.444 ± 0.195	14.361 ± 0.092	15.465 ± 0.663
	200	17.521 ± 1.082	17.317 ± 1.002	17.869 ± 1.160	17.870 ± 1.019	13.306 ± 0.589	13.252 ± 0.421	15.386 ± 0.447
	300	17.147 ± 1.376	16.948 ± 1.197	17.323 ± 1.269	17.811 ± 1.836	13.554 ± 0.771	13.384 ± 0.104	15.593 ± 0.121
Konya	100	8.611 ± 2.081	9.047 ± 2.433	9.017 ± 2.236	9.805 ± 2.476	3.743 ± 0.709	3.654 ± 0.629	4.198 ± 1.003
	200	8.788 ± 2.352	9.231 ± 2.557	8.912 ± 2.276	9.467 ± 2.640	3.547 ± 0.749	3.422 ± 0.661	3.450 ± 0.856
	300	8.593 ± 2.207	9.189 ± 2.550	8.809 ± 2.206	9.728 ± 2.486	3.592 ± 0.916	3.444 ± 0.891	3.785 ± 1.045

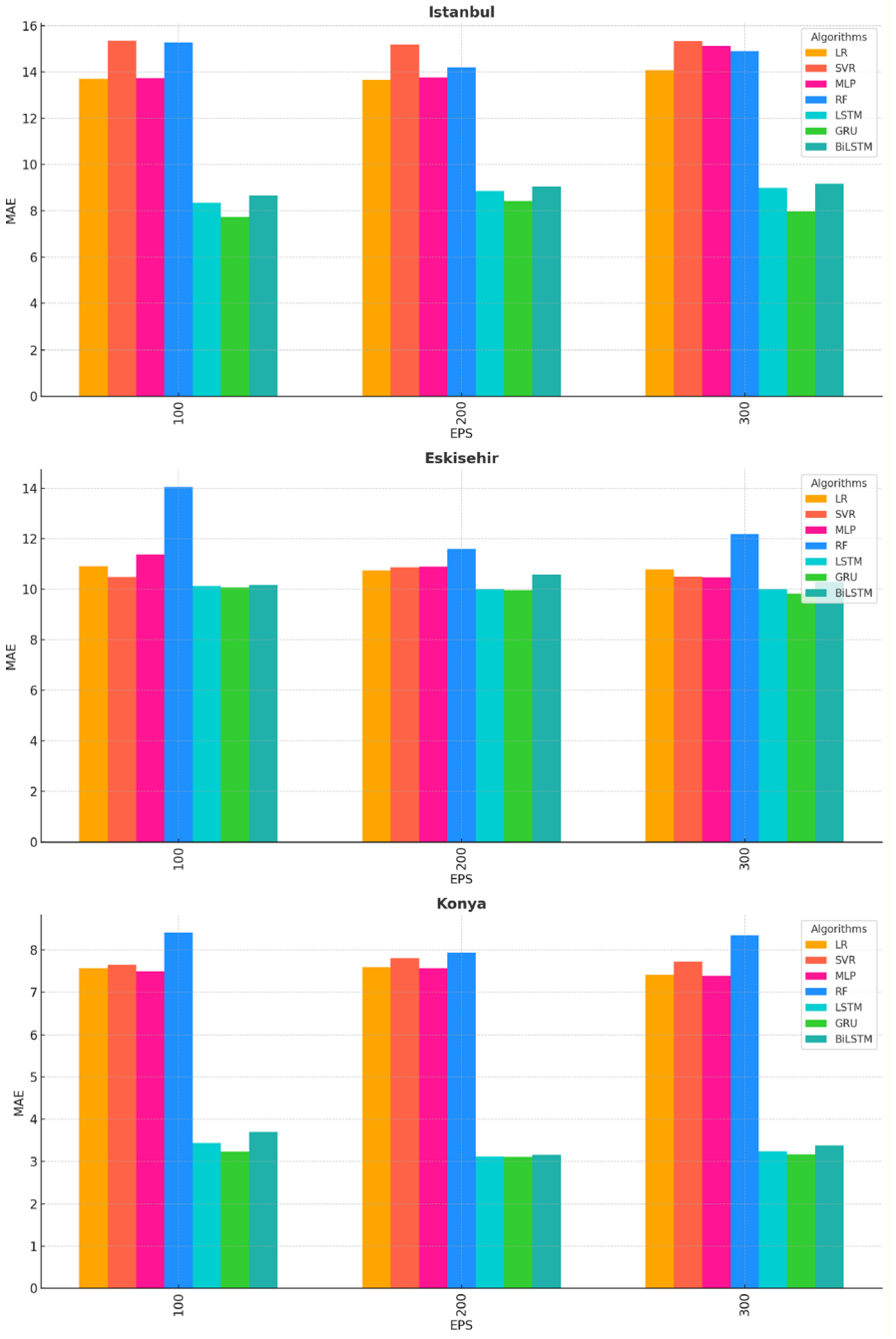


Fig. 11 MAE results of the algorithms

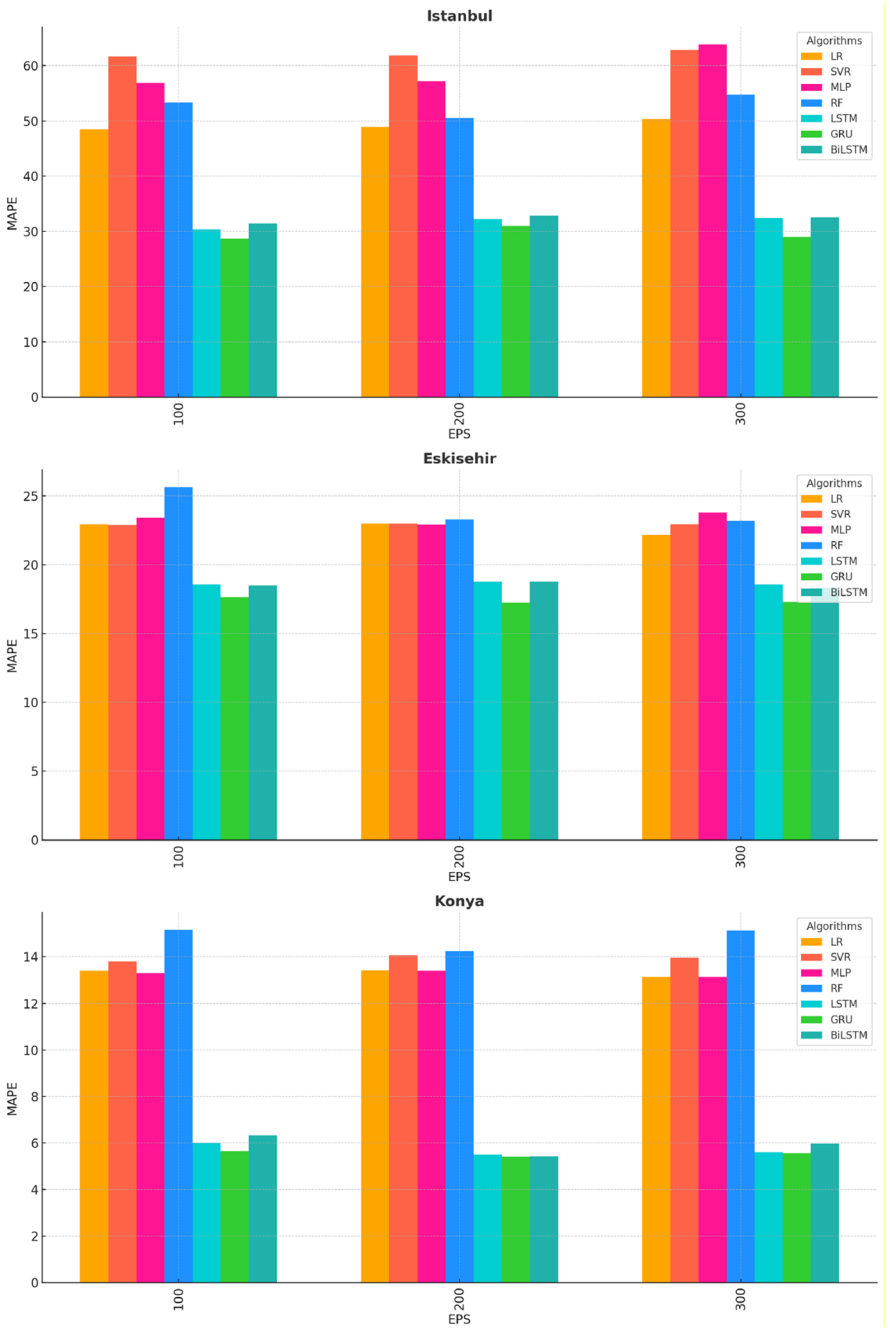


Fig. 12 MAPE results of the algorithms

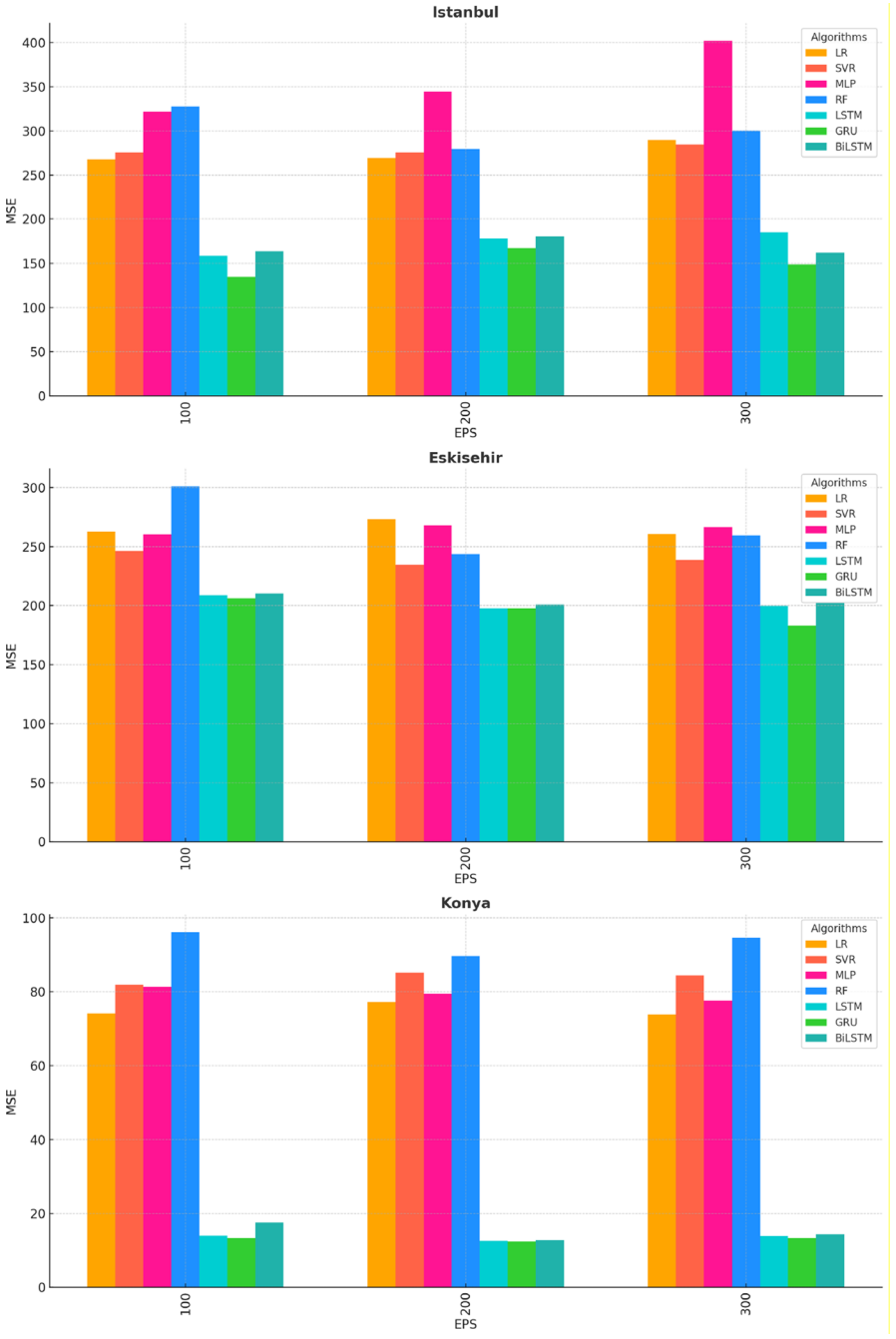


Fig. 13 MSE results of the algorithms

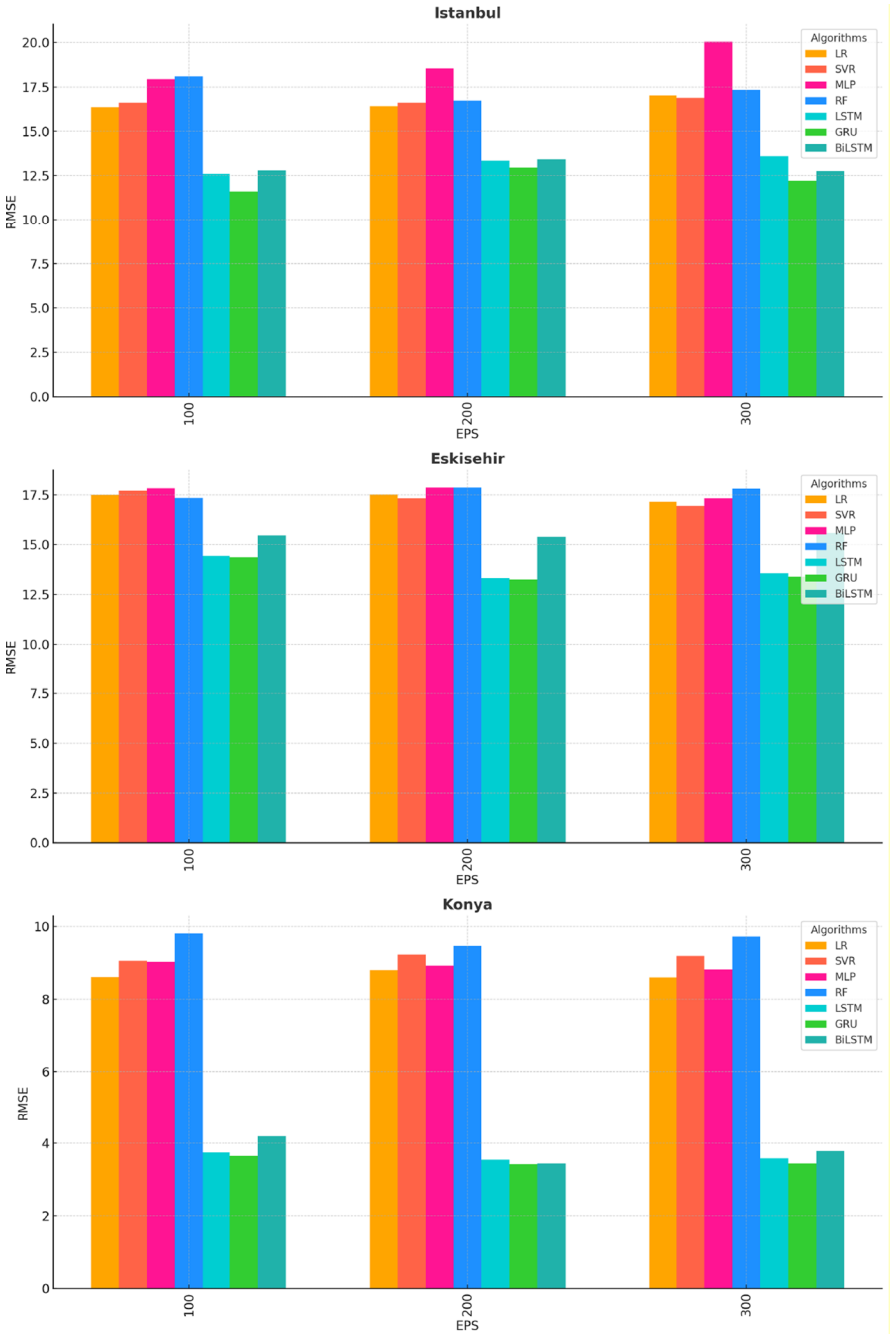


Fig. 14 RMSE results of the algorithms

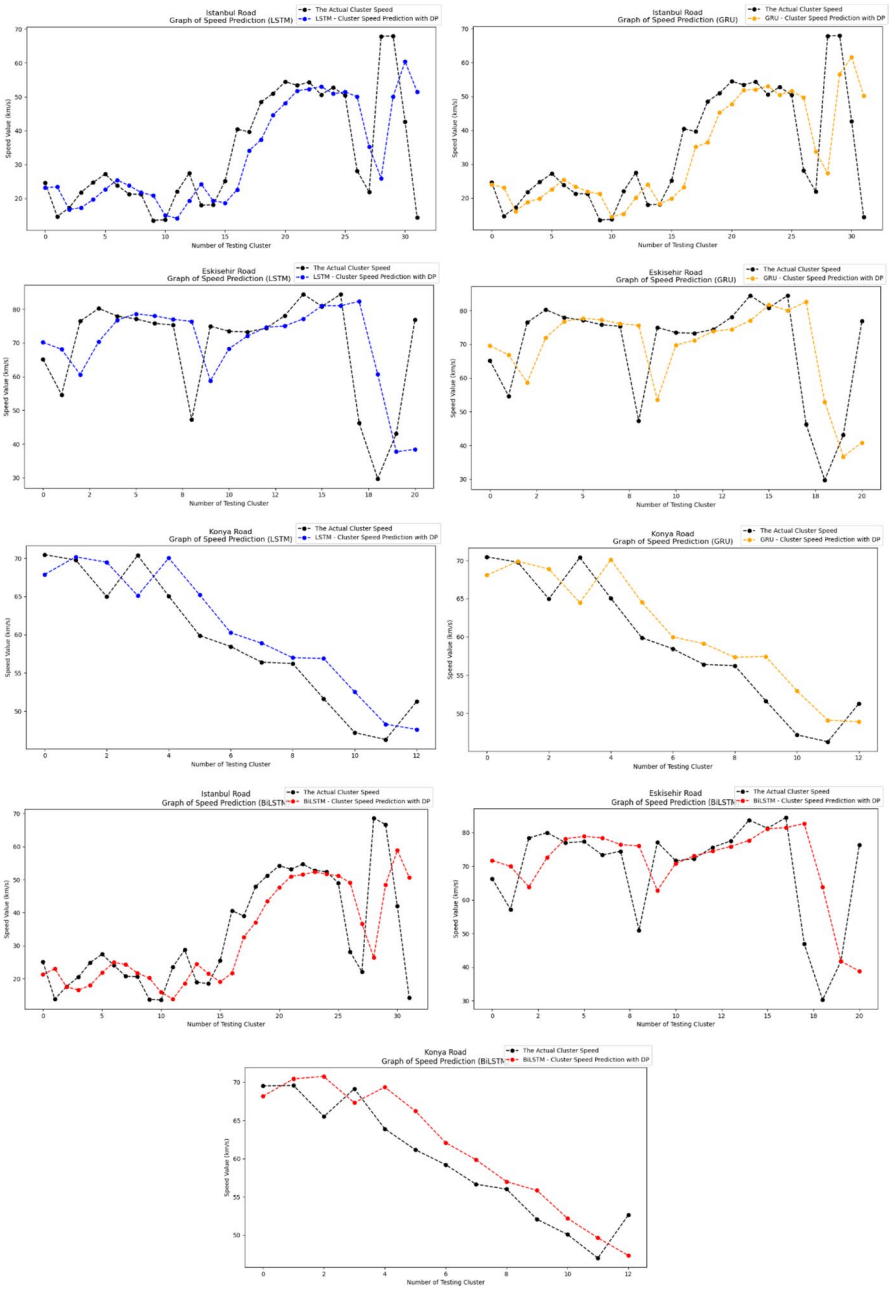


Fig. 15 Speed prediction results

tively. They employed GRU and federated learning and tested the proposed model on the PeMS dataset with short-term analysis. Ren and Tang [31] presented a minimum error rate of MAER 1.4% for density prediction. They utilized the T-Drive and San Francisco datasets and introduced a linear programming-based framework. For speed prediction, Xia et al. [23] obtained the best values with MAE of 0.81, MAPE of 1.47%, RMSE of 1.54, and MAAPE of 1.46%. They employed federated learning and GCN and used the PeMS dataset in their experiments.

- Compared to the studies above, our paper proposes a framework for privacy-aware speed prediction. The superior performance of the proposed framework was achieved with GRU MAE of 7.7, MAPE of 28.5%, MSE of 132.7, and RMSE of 11.5 for Istanbul, MAE of 9.7, MAPE of 17.2%, MSE of 177.4, and RMSE of 13.2 for Eskisehir, and MAE of 3.1, MAPE of 5.4%, MSE of 12.4, and RMSE of 3.4 for Konya Roads with the epsilon value of 200, respectively. The results are promising for future research in privacy-aware traffic speed prediction.
- Datasets: Existing studies use public datasets such as PeMS, Taxi, and T-Drive. The PeMS dataset is collected from some detectors located at fixed coordinates on the highways, in California. Speed, flow, and occupancy signals are obtained from the detectors. In addition, TaxiNYC and TaxiBJ datasets are generated by collecting data such as trip time, trip distance, and pickup and drop-off coordinates from taxi cabs. Also, the T-Drive dataset includes time stamp and GPS location features.
- However, our study utilizes a compiled dataset comprising real-world data, particularly signals obtained from CAN-Bus devices of vehicles moving on the road. Our dataset includes features such as GPS location, speed, and direction. Collecting these features enables a thorough evaluation and analysis of predictions along any desired path, ensuring reliable results. Additionally, our dataset contains a total of 17 million records, posing a significant challenge in managing large volumes of data, a key issue in data analysis. Unlike the previously mentioned public datasets, ours encompasses a larger volume of data. We addressed this challenge by employing feature extraction, a critical component of our framework.
- Applicability: Working with real-world datasets generally provides more reliable solutions to real-world problems. Our framework utilizes a real-world dataset collected from moving vehicles on three major roads in Ankara, the capital city of Turkey. Furthermore, traffic data from any other city can be applied within our framework to generate reliable, privacy-preserved results.
- Privacy and utility aspect: Upon evaluating Fig. 10, it becomes evident that the proposed model demonstrates satisfactory performance compared to non-private results. Thus, our research successfully balances privacy preservation with data utility. The proposed framework is capable of predicting traffic speed while addressing privacy concerns.

When our model is examined in terms of practical implications, its limitations, and real-time scenarios:

- Practical implications: Government agencies and municipalities that provide traffic management applications often obtain vehicle traffic data from diverse sources such as cameras, sensors, and loop detectors installed on roads. These data only pertain to the road of interest and only the traffic dynamics related to the relevant road can be measured and decisions can be made. Literature studies only utilize data with specific geographical regions and temporal limitations, as shown by the examples provided in Table 1. Therefore, the reliance on such restricted data may influence the breadth and applicability of the findings. Additionally, the cost of the equipment placed on the road is high depending on their technology and it is not possible to place them on every road. In order to overcome these constraints, the most prominent data type and available input variables for an optimal model are moving vehicle data (including GPS, speed, angular direction, etc.) based on crowdsourced technology, which can be easily obtained even from mobile phones in today's technologies. In our study, a geographically independent model has been developed that relies on data collected from moving vehicles (cargo vehicles, couriers, taxis, etc.) at the country or city scale, without dependent on a fixed data source. Since the developed model does not use a fixed data source, it is independent of poorly instrumented areas, and in areas where maybe data are insufficient.
- Limitations: The model developed in this study is based solely on moving vehicle data and cannot deal with other different data types. In addition, to place GPS data on the map, road geometry data (i.e., files with SHP extension) must be available from an institution working on geographic information systems. Once the GPS data are correctly matched with the SHP data, prediction can be made using the developed model. Furthermore, the data provider needs to provide not only GPS coordinates and vehicle speeds but also angular directions. If the angular directions are not provided, the direction of vehicles traveling in the same direction cannot be detected and naturally the model cannot produce accurate predictions for the corresponding road. Although the proposed framework provides strong privacy-preserving speed estimation, we acknowledge about making various assumptions such as the data being of high quality and representing the traffic completely, and data sparsity or inconsistency may affect the estimation accuracies. It is known that the implementation of the proposed main framework in real-world scenarios will bring about problems such as real-time responsiveness and high computational cost regarding the use of large-scale datasets. The accessibility of data is another important limitation, as high-resolution and realistic traffic datasets are not always available. In order to overcome these problems, it is thought that in future studies, the proposed framework can be developed in accordance with distributed computing techniques for real-time and scalable implementation. Techniques such as data augmentation and data imputation could also be employed to address issues like data inconsistency or incompleteness.
- Real-time scenarios: The proposed framework is designed for batch processing. However, with the help of some modifications, it can handle large volumes of data in real-time scenarios, which is critical for deployment in intelligent transportation systems. The necessary modifications include incremental clus-

tering, streaming-based predictions, and processing large data volumes. The OPTICS clustering algorithm can be adapted to process streaming data incrementally, dynamically updating clusters as new data arrives. The deep learning algorithms (LSTM, BiLSTM, and GRU) and machine learning algorithms (LR, SVR, MLP, and RF) used for prediction can be deployed in a real-time environment using frameworks like TensorFlow Serving or PyTorch Serve. These platforms are optimized for low-latency inference. Additionally, the framework can be extended to leverage distributed computing platforms such as Apache Spark or cloud-based solutions. These platforms enable parallel processing of the large data volumes generated by moving vehicles, ensuring the system can meet real-time requirements even during peak traffic hours.

7 Conclusion

In this study, a state-of-the-art, privacy-aware framework for short-term traffic speed prediction in ITS was proposed. Experimental results were obtained from three main arteries in Ankara: Istanbul, Eskisehir, and Konya Roads, using LSTM, BiLSTM, GRU, LR, SVR, MLP, and RF algorithms. As shown in Tables 7, 8, 9 and 10 and Figs. 11, 12, 13 and 14, the performance of the algorithms varies with different epsilon values. Generally, the GRU algorithm provided better results compared to the others. Figure 15 demonstrates that the prediction results are acceptable for short-term predictions, while Fig. 10 supports the applicability of the proposed framework.

Compared to other studies on traffic parameters in the literature, this is the first to propose a privacy-aware, feature engineering-based, and geo-independent framework for short-term speed prediction. Vehicle data, including coordinate information, vehicle speed, and vehicle angular direction, are primarily considered in our study. This information serves as fundamental features for understanding vehicular dynamics and behavior. However, it is essential to recognize that the integration of these features with additional data types can significantly improve the quality and depth of analysis. For instance, incorporating vehicle altitude information or vehicle type into the dataset could enhance predictions by accounting for variations in altitude and their effect on different vehicle types. Consequently, traffic flow and speed can be predicted more accurately. Similarly, environmental factors, such as weather conditions, can be included to understand the impact of external influences on traffic flow. Data analysis could reveal trends regarding traffic behavior under extreme weather conditions, such as rain or snow. By incorporating more vehicle features and combining them with environmental data, more effective traffic predictions can be made. Traffic authorities could obtain detailed and comprehensive insights into traffic dynamics. This research is expected to offer valuable contributions to traffic prediction studies that incorporate privacy considerations.

This study is evaluated to consider various issues as future research.

- One of the most important potentials is to test more complex urban traffic networks or larger datasets with the proposed framework in order to evaluate the

generalizability and scalability of the model. Integrating additional variables such as environmental factors, road conditions, and weather conditions can enhance prediction accuracy. Integrating with other data types or using more advanced artificial intelligence algorithms can provide more sophisticated traffic management solutions. In addition, evaluating the integration of the proposed framework into smart city applications and measuring its applicability to traffic policies can reveal important insights.

- Thanks to the ability to capture topological dependencies between road segments, graph neural networks can be used or compared with the proposed framework, which can provide a comparative perspective on traffic dynamics. For example, applying graph neural networks in traffic prediction can reveal the advantages of our proposed model in terms of geo-independent and privacy protection issues.
- The integration of federated learning into the proposed framework considered for future work that will explore to further enhance privacy preservation and scalability.
- Adopting the framework for real-time applications by integrating incremental clustering with streaming data for dynamic updates and applying predictive models (e.g., LSTM, BiLSTM, GRU) on low-latency platforms such as TensorFlow Serving or PyTorch Serve is another future work. Distributed computing platforms, such as Apache Spark or cloud-based solutions, can be leveraged to efficiently manage large-scale, real-time data streams, ensuring scalability and responsiveness during peak traffic periods. These adjustments are crucial for implementation in intelligent transportation systems.

Although the proposed model offers significant improvements, it is necessary to address its various limitations. One limitation relates to moving vehicle data, which may not be continuously available in all areas or road segments, especially in poorly equipped areas or at all hours. To mitigate this, additional data sources could be explored. Another challenge is the dependence on road geometry data, which may not always be accessible or available. Although differential privacy guarantees data privacy, the balance between privacy and utility is one of the important issues. Adding more noise to protect privacy can reduce estimation accuracy. To address this, adaptive privacy mechanisms could be employed to adjust the amount of noise based on density or other factors. Finally, the computational costs of real-time systems are a significant concern, particularly for processing large datasets and applying privacy protection techniques. By addressing these challenges, the framework can be made more robust and applicable to very large areas.

Acknowledgements The authors thank Başarsoft Information Technologies Inc. for providing traffic data and their valuable support.

Author Contribution Murat Akin contributed to the writing of the original draft, as well as serving in roles related to data curation, investigation, methodology, and conceptualization. Yavuz Canbay also participated in writing the original draft and was involved in methodology development, visualization, and conceptualization. Seref Sagiroglu provided critical input through writing review and editing, validation, and supervision of the research process. All authors reviewed the manuscript and collaborated to ensure the integrity and quality of the work presented.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK).

Data availability No datasets were generated or analyzed during the current study.

Declarations

Conflict of interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Qiu H, Zheng Q, Msahli M, Memmi G, Qiu M, Lu J (2020) Topological graph convolutional network-based urban traffic flow and density prediction. IEEE Trans Intell Transp Syst 22(7):4560–4569
2. Hou Y, Chen J, Wen S (2021) The effect of the dataset on evaluating urban traffic prediction. Alex Eng J 60(1):597–613
3. Mathew TV, Rao KK (2007) Fundamental parameters of traffic flow. *Introduction to Transportation Engineering; National Program on Technical Education and Learning (NPTEL): Mumbay, India*, 1–8
4. El Sherief MM, Ramadan IM, Ibrahim AM (2016) Development of traffic stream characteristics models for intercity roads in Egypt. Alex Eng J 55(3):2765–2770
5. Kovacic K, Ivanjko E, Jelusic N (2015) Measurement of road traffic parameters based on multi-vehicle tracking. Proceedings of the Croatian Computer Vision Workshop
6. Chen X, Chen R (2019) A review on traffic prediction methods for intelligent transportation system in smart cities. In: 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), IEEE, pp. 1–5
7. Fan X, Xiang C, Gong L, He X, Qu Y, Amirgholipour S, He X (2020) Deep learning for intelligent traffic sensing and prediction: recent advances and future challenges. CCF Transactions on Pervasive Computing and Interaction 2:240–260
8. Akallouch M, Akallouch O, Fardousse K, Bouhoute A, Berrada I (2022) Prediction and privacy scheme for traffic flow estimation on the highway road network. Information 13(8):381
9. Hossain MN, Ahmed N, Ullah SW (2022) Traffic flow forecasting in intelligent transportation systems prediction using machine learning. 2022 International Conference on Futuristic Technologies (INCOFT)
10. Lonare S, Bhramaramba R (2024) Federated approach for privacy-preserving traffic prediction using graph convolutional network. J Shanghai Jiaotong University (Science) 29(3):509–517
11. Meese C, Chen H, Asif SA, Li W, Shen CC, Nejad M (2022) Bfrt: Blockchained federated learning for real-time traffic flow prediction. 2022 22nd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid)
12. Zhou K, Liu Y (2024) Addressing the privacy and complexity of urban traffic flow prediction with federated learning and spatiotemporal graph convolutional networks. In: Wand M, Malinovská K, Schmidhuber J, Tetko IV (eds) Artificial Neural Networks and Machine Learning – ICANN 2024: 33rd International Conference on Artificial Neural Networks, Lugano, Switzerland, September 17–20, 2024, Proceedings, Part VI. Springer Nature Switzerland, Cham, pp 129–142. https://doi.org/10.1007/978-3-031-72347-6_9

13. Do LN, Vu HL, Vo BQ, Liu Z, Phung D (2019) An effective spatial-temporal attention based neural network for traffic flow prediction. Transp Res Part C: Emerging Technol 108:12–28
14. Gao J, Sun L, Cai M (2019) Quantifying privacy vulnerability of individual mobility traces: a case study of license plate recognition data. Transport Res Part C: Emerg Technol 104:78–94
15. Ciftler BS, Albaseer A, Lasla N, Abdallah M (2020) Federated learning for RSS fingerprint-based localization: A privacy-preserving crowdsourcing method. In: 2020 International Wireless Communications and Mobile Computing (IWCMC), IEEE, pp. 2112–2117
16. Fieschi A, Li Y, Hirmer P, Stach C, Mitschang B (2023) Privacy in Connected Vehicles: Perspectives of Drivers and Car Manufacturers. In: Aiello M, Barzen J, Dustdar S, Leymann F (eds) Service-Oriented Computing: 17th Symposium and Summer School, SummerSOC 2023, Heraklion, Crete, Greece, June 25 – July 1, 2023, Revised Selected Papers. Springer Nature Switzerland, Cham, pp 59–68. https://doi.org/10.1007/978-3-031-45728-9_4
17. Zhang W, Jiang B, Li M, Lin X (2022) Privacy-preserving aggregate mobility data release: an information-theoretic deep reinforcement learning approach. IEEE Trans Inf Forensics Secur 17:849–864
18. Liu Y, James J, Kang J, Niyato D, Zhang S (2020) Privacy-preserving traffic flow prediction: a federated learning approach. IEEE Internet Things J 7(8):7751–7763
19. Kaur G, Grewal SK, Jain A (2024) Federated Learning based Spatio-Temporal framework for real-time traffic prediction. Wirel Pers Commun 136(2):849–865
20. Lee S, Sung J, Shin M-K (2024) Layer-wise personalized federated learning for mobile traffic prediction. IEEE Access 12:53126–53140. https://doi.org/10.1109/ACCESS.2024.3384566
21. Zhang R, Mao J, Wang H, Li B, Cheng X, Yang L (2024) A survey on federated learning in intelligent transportation systems. IEEE Trans Intell Vehicles. https://doi.org/10.1109/TIV.2024.3446319
22. Qi Y, Hossain MS, Nie J, Li X (2021) Privacy-preserving blockchain-based federated learning for traffic flow prediction. Futur Gener Comput Syst 117:328–337
23. Xia M, Jin D, Chen J (2022) Short-term traffic flow prediction based on graph convolutional networks and federated learning. IEEE Trans Intell Transp Syst 24(1):1191–1203
24. Deng J, Shen G (2022) Federated learning-based privacy-preserving traffic flow prediction scheme for VANETs. In: 2022 4th International Conference on Communications, Information System and Computer Engineering (CISCE), IEEE, pp. 374–378
25. Liu Y, Zhang S, Zhang C, James J (2020) Fedgru: Privacy-preserving traffic flow prediction via federated learning. In: 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)
26. Yuan X, Chen J, Yang J, Zhang N, Yang T, Han T, Taherkordi A (2022) Fedstn: graph representation driven federated learning for edge computing enabled urban traffic flow prediction. IEEE Trans Intell Transport Syst 24(8):8738–48
27. Liu Y, Rasouli S, Wong M, Feng T, Huang T (2024) RT-GCN: Gaussian-based spatiotemporal graph convolutional network for robust traffic prediction. Information Fusion 102:102078
28. Xie Y, Jin C (2024) Evaluations of multi-step traffic flow prediction models based on graph neural networks. In: 2024 6th International Conference on Communications, Information System and Computer Engineering (CISCE)
29. Hu N, Liang W, Zhang D, Xie K, Li K, Zomaya AY (2024) FedGCN: A federated graph convolutional network for privacy-preserving traffic prediction. IEEE Trans Sustain Comput 9(6):925–935
30. Miao Y, Bai X, Cao Y, Liu Y, Dai F, Wang F, Dou W (2023) A novel short-term traffic prediction model based on SVD and ARIMA with blockchain in industrial internet of Things. IEEE Internet of Things Journal 10(24):21217–21226
31. Ren W, Tang S (2020) EGeoIndis: an effective and efficient location privacy protection framework in traffic density detection. Vehicular Communications 21:100187
32. Huang Y, Tian Y, Liu Z, Jin X, Liu Y, Zhao S, Tian D (2020) A traffic density estimation model based on crowdsourcing privacy protection. ACM Transact Intell Syst Technol (TIST) 11(4):1–18
33. Zhang C, Zhang S, James J, Yu S (2021) FASTGNN: a topological information protected federated learning approach for traffic speed forecasting. IEEE Trans Industr Inf 17(12):8464–8474
34. Hassija V, Gupta V, Garg S, Chamola V (2020) Traffic jam probability estimation based on blockchain and deep neural networks. IEEE Trans Intell Transp Syst 22(7):3919–3928
35. Wang Q, Ji T, Guo Y, Yu L, Chen X, Li P (2020) TrafficChain: a blockchain-based secure and privacy-preserving traffic map. IEEE Access 8:60598–60612
36. Ćorović A, Ilić V, Đurić S, Marijan M, Pavković B (2018) The real-time detection of traffic participants using YOLO algorithm. In: 2018 26th telecommunications Forum (TELFOR), IEEE, pp. 1–4

37. Mhalla A, Chateau T, Gazzah S, Amara NEB (2018) An embedded computer-vision system for multi-object detection in traffic surveillance. IEEE Trans Intell Transp Syst 20(11):4006–4018
38. Chougule A, Chamola V, Hassija V, Gupta P, Yu FR (2023) A novel framework for traffic congestion management at intersections using federated learning and vertical partitioning. IEEE Trans Consum Electron 70(1):1725–1735
39. Sri LS, Karthick A, Akash S, Anuradha R (2023) Traffic Prediction Using Graph Neural Network. In: 2023 IEEE Guwahati Subsection Conference (GCON)
40. Alizadeh M, Mousavi SE, Beheshti MT, Ostadi A (2021) Combination of feature selection and hybrid classifier as to network intrusion detection system adopting FA, GWO, and BAT optimizers. In: 2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS), IEEE, pp. 1–7
41. Ankerst M, Breunig MM, Kriegel H-P, Sander J (1999) OPTICS: Ordering points to identify the clustering structure. ACM SIGMOD Rec 28(2):49–60
42. Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining 96(34):226–231
43. Dwork C (2006) Differential Privacy. In: Bugliesi M, Preneel B, Sassone V, Wegener I (eds) Automata, Languages and Programming. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 1–12. https://doi.org/10.1007/11787006_1
44. Arachchige PC, Bertok P, Khalil I, Liu D, Camtepe S, Atiquzzaman M (2019) Local differential privacy for deep learning. IEEE Internet Things J 7(7):5827–42
45. Fukuchi K, Tran QK, Sakuma J (2017) Differentially private empirical risk minimization with input perturbation. In: Yamamoto A, Kida T, Uno T, Kuboyama T (eds) Discovery Science: 20th International Conference, DS 2017, Kyoto, Japan, October 15–17, 2017, Proceedings. Springer International Publishing, Cham, pp 82–90. https://doi.org/10.1007/978-3-319-67786-6_6
46. Lu Z, Shen H (2020) Differentially private $ k $ k-means clustering with convergence guarantee. IEEE Trans Dependable Secure Comput 18(4):1541–1552
47. Fusco G, Colombaroni C, Isaenko N (2016) Short-term speed predictions exploiting big data on large urban road networks. Transport Res Part C: Emerg Tech 73:183–201
48. Canbay P, Tas H (2022) Prediction of heating and cooling loads of buildings by artificial intelligence. Int J Pure Appl Sci 8(2):478–489
49. Utku A (2023) Deep learning based hybrid prediction model for predicting the spread of COVID-19 in the world's most populous countries. Expert Syst Appl 231:120769. https://doi.org/10.1016/j.eswa.2023.120769
50. Winarno E, Hadikurniawati W, Rosso RN (2017) Location based service for presence system using haversine method. In: 2017 international conference on innovative and creative information technology (ICITech)
51. Chopde NR, Nichat M (2013) Landmark based shortest path detection by using A* and Haversine formula. Int J Innovative Res Comput Eng 1(2):298–302

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Murat Akin[1] · Yavuz Canbay[2] · Seref Sagiroglu[3]

✉ Murat Akin
muratakin@gazi.edu.tr

Yavuz Canbay
yavuzcanbay@ksu.edu.tr

Seref Sagiroglu
ss@gazi.edu.tr

[1] Faculty of Engineering, Gazi University, Ankara, Turkey

[2] Department of Computer Engineering, Faculty of Engineering, Kahramanmaras Sutcu Imam University, Kahramanmaras, Turkey

[3] Department of Computer Engineering, Faculty of Engineering, Gazi University, Ankara, Turkey