Short-Term Traffic Flow Prediction Using the Modified Elman Recurrent Neural Network Optimized Through a Genetic Algorithm

ABOLGHASEM SADEGHI-NIARAKI\textsuperscript{1,2}, PARIMA MIRSHAFIEI\textsuperscript{3}, MARYAM SHAKERI\textsuperscript{1}, and SOO-MI CHOI\textsuperscript{2}

\textsuperscript{1}Geoinformation Technology Center of Excellence, Faculty of Geomatics Engineering, K. N. Toosi University of Technology, Tehran 19697, Iran
\textsuperscript{2}Department of Computer Science and Engineering, and Convergence Engineering for Intelligent Drone, Sejong University, Seoul 05006, South Korea
\textsuperscript{3}Department of Mathematics and Computer Science, Eindhoven University of Technology, 5612 AZ Eindhoven, The Netherlands

Corresponding author: Soo-Mi Choi (smchoi@sejong.ac.kr)

This work was supported by the Ministry of Science and ICT (MSIT), South Korea, through the Information Technology Research Center (ITRC) Support Program, supervised by the Institute for Information and communications Technology Planning and Evaluation (IITP), under Grant IITP-2020-2016-0-00312.

\textbf{ABSTRACT} Traffic stream determining is an essential part of the intelligent transportation management system. Precise prediction of traffic flow provides a basis for other tasks, like forecasting travel time. While traditional methods have some merits for improving traffic prediction precision in some ways, high precision, considering different circumstances, is still difficult to achieve. This article presents a short-term traffic flow prediction model based on the Modified Elman Recurrent Neural Network model (GA-MENN) to deal with this practical problem. In GA-MENN, the algorithm of Elman Recurrent Neural Network is modified, optimized through the Genetic Algorithm (GA) and considered weather conditions, weekday, hour and day’s classification to forecast the vehicle velocity in Tehran streets and highways. The traffic data were collected from the online Google Map API service for 139 routes in 7 districts in Tehran. The method improves prediction precision and also lowers the prediction error rate, according to experimental results. Exploratory outcomes verify the superior performance of the proposed traffic condition prediction model over Regression Multi-layer Perceptron, Linear Regression, Logistic Regression, Probabilistic Neural Network, Regression Generalized Feedforward, Time-lag Recurrent Network, Support Vector Machine model, Elman neural network, K-NN model, ARIMA, Kalman filter model, Convolutional Neural Networks (CNNs), SARIMA, and Long Short-Term Memory (LSTM) model. To the best of our knowledge, this is the first occasion when that traffic stream is gauged in urban roads and avenues in this specific way.

\textbf{INDEX TERMS} Intelligent transportation, short-term traffic flow prediction, modified Elman recurrent neural network, genetic algorithms.

\section{I. INTRODUCTION}

With the growth of city-dwelling populations, cities are in need of increasingly sophisticated transportation systems to address the transport needs of its citizens [1], [2]. In many cities, the failure to address this issue has led to increased use of private motor vehicles on the streets, leading to exacerbated traffic congestion problems [3]. Given the end effects of traffic congestion such as air pollution, prolonged travel time, and dissatisfaction of citizens, the prediction of traffic flows on the city streets has become a popular subject of research.

Traffic flow forecasting methods can be categorized based on whether they are model-based or data-driven, direct or indirect, and parametric or nonparametric. In the data-driven approach, predictions are based on historical data, or more specifically based on a comparison between current traffic conditions and the most similar historical instances in terms of average speed, traffic volume, the hour of the day, etc. Models and algorithms that operate based on this approach include ARIMA [4], [5], linear models, neural networks, support vector regression model [6], and non-linear time-series models.

The model-based methodology is to make an expectation dependent on demand estimates and capacity estimates for...
road segments. Given the reliance of this method on estimates, it is more complex than the other approaches and requires more careful supervision. Notable examples of this approach include DyanMIT-R, SBOTTP, TOPL (CTM) and OLSM [7].

In the indirect approach, traffic flow predictions are based on the differences in inputs (traffic flow, velocity, volume, etc.), but in the direct approach, traffic flow is predicted based on the past data. The parametric approach involves formulating a mathematical model for the traffic flows. The methods following this approach include Markov chain algorithms, Kalman filtering [8], autoregressive integrated moving average (ARIMA) model [9], exponential smoothing algorithm [10], chaos theory, wavelet algorithm [11] and Bayesian network. The nonparametric approach involves analyzing the historical data to extract some rules for the relationship between them and then predicting the future traffic conditions accordingly. This approach is used in many methods including artificial neural networks (ANNs) [12], support vector regression (SVR) [13], k-nearest neighbor (k-NN) [14], and reinforcement learning [15]. Reference [16] optimized the traffic flow congestion and predict it. They used Markov Decision Process and Q-Learning to learn policies. Reference [17] proposed a computational intelligent model.

Over the years, short-term flow prediction has been addressed by many studies, which have had many interesting results, but there is no a reliable research study that predicts traffic flow using weather conditions, weekday, hour and day’s classification at the same time, which is affecting the traffic. This is the first attempt to produce a model for traffic flow predictions based on traffic information, spatial-temporal information, weather data, the day type in terms of being a working/weekend day, and the day type in terms of holiday status, all together. Considering the substantial impact of weather conditions and day type (working days/weekends, holidays) on traffic flows, incorporation of these variables into the prediction process can yield far more accurate short-term traffic flow projections.

Therefore, this article introduces a GA-MENN model for traffic flow prediction by considering various factors together. This model makes use of the modified Elman recurrent neural network with delay effects incorporated into the formulation to achieve higher degrees of realism. Also, the genetic algorithm (GA) is utilized for process optimization. Not only does this model prevent falling into local minima, it is also very agile to find solutions thanks to GA. Likewise, GA-MENN is a dynamic NN that has the capacity of dynamic mapping function by saving inner history state information to adjust to time-changing attributes. The model uses road data, weather information, working days/non-working status (holiday status), day of the week, and hour of the day for a more accurate short-term traffic flow prediction. To consider holidays in the traffic prediction, days are classification in terms of the holidays into four types of working and non-working days. Traffic data of the roads is obtained from the Google Map API service, which is chosen because of its high accuracy and broad coverage.

The model performance is assessed on the basis of the modified mean absolute percent error (MMAPE) [75], Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean-Squared Error (MSE), and in comparison with over-regression, multi-layer perceptron, linear regression, logistic regression, probabilistic neural network, generalized feedforward regression, time-lag recurrent network, support vector machine models, Elman neural network, K-NN model, ARIMA, Kalman filter model, Convolutional Neural Networks (CNNs), SARIMA, and Long-Short-Term Memory (LSTM) model.

The rest of this document is organized as follows. Section 2 examines the history of research on short-term traffic flow prediction. Section 3 describes the data collection method and the developed model. In Section 4, the model is applied to the existing data and the results are compared with seven other models. Finally, conclusions and suggestion for future works are provided in Section 5.

II. LITERATURE REVIEW

The history of research on short-term traffic flow prediction dates back to the 1980s. The most important part of predicting traffic flow is to construct the traffic prediction model, which can be done in many ways. In research by [18], variable-order Markov (VOM) model and probability suffix tree were used to develop an association rules based method with enhanced prediction performance. They calculated the overall transport transition of a region instead of calculating the velocity of a specific region. AR algorithm is used to extract road relations. Reference [19] proposed a novel model called pattern sensitive network that uses adversarial training to accurately predict the flow of traffic in typical and atypical conditions. In another work, [20] introduced a new method called the online learning weighted support-vector regression (OLWSVR) for the prediction of short-term traffic. For long, neural network models have also been extensively used in traffic flow prediction, as they allow researchers to use non-linear equations and universal approximability of unknown functions. Contrasted with traditional factual forecast models, neural networks possess some inherent properties that make them more suitable for a variety of modeling applications ranging from multivariate modeling to flow modeling. First of all, they are self-adaptive methods driven by data and can identify the relationship with minimal a priori information [21]. Second, given their capacity to gain from information (notwithstanding while fundamental connections between factors are not known), their nonlinear nature, and their capacity for generalization (subsequent to training with test information, they can give expectations to the piece of the information that isn’t utilized), these models are perfects tools for working with noisy databases, which is not uncommon in time-series modeling for traffic prediction in real-time.
The above characteristics show how Artificial Neural Networks can theoretically model temporal as well as spatial traffic. In this area, the literature contains many models of neural networks consisting of structures that are fully static to highly dynamic. A large number of these models have been utilized in traffic congestion modeling. For example, [22] used a combination of neural networks and Bayesian inference for traffic prediction. They coupled the back propagation and the radial basis function into a Bayesian mixed neural network model. In another work, [23] introduced another technique, based on three traffic microwave sensors collected in Beijing, which included a fuzzy neural network structure for traffic prediction. In addition, they used a weighted recursive least squares estimator to optimize the parameters.

Recently, a number of improved algorithms such as Long Short-Term Memory Neural Network (LSTM) [24], [25], state-space neural network model [26], ensemble learning algorithms [27]–[30] and hybrid algorithms [31] & [32] have shown superior effectiveness and computational performance over traditional methods. Reference [33] proposed a LSTM method with feature enhancement in order to predict the traffic flow. The algorithm complements LSTM deficiencies at excessively long distances. In [38], they suggest a recurrent neural network based on Attention-based long-term memory (LSTM). They assess the forecast architecture of the Gray-Chicago-Milwaukee Transport Corridor (GCM) information from Chicagoland in real-time.

In another work, a novel fuzzy-based convolutional neural network (F-CNN) approach for predicting more accurate traffic flow is suggested by applying a fuzzy approach which reflects the features of traffic accidents when first introduced uncertain information for road accidents in the CNN [36].

Besides the mentioned NN models, other supervised learning methods are used in different studies as well. Reference [34] combined the swarm optimization with genetic algorithm to optimize the least square support vector machine and forecast the traffic flow. The model was compared with non-heuristic and heuristic algorithms and proved its ability in predicting the traffic flow. Reference [35] proposed a prediction method by combining support vector machine model and five different denoising algorithms to improve traffic flow forecast precision. A Date Pattern and Likelihood Ratio Trials based for urban short traffic speed forecasting from large scale taxi GPS data is proposed in [37].

Reference [39] extends the applicability of the KNN technique to short-term traffic volume projections and offers a viable strategy to asymmetrical loss projections. Reference [40] implemented an enhanced K-nearest neighbors (K-NN) algorithm to propose a data-driven and non-parametric methodology in order to predict traffic flow congestion by identification of comparable patterns of traffic. While these mentioned studies have proposed different types of methods that have their own advantages and drawbacks to predict short-term traffic flow, they have not evaluated the Elman type of the recurrent neural network, which is a powerful network for dynamic and real-time applications [76].

Several researchers have worked to replace traditional regressions and time-series [41] & [42] with the use of the neural networks. All these works have pointed out the potential superiority of neural networks over the aforementioned approaches and recommended further research using larger and more realistic databases in different fields. Elman proposed in 1990 a model for neural network speech processing, which today is known as the Elman Recurrent Neural Network (ERNN) [43]. The outputs of the hidden layer in ERNN are passed on to the buffer layer, the recurrent layer, and then returned as feedback to the hidden layer. This feedback enables ERNN to track spatial and temporal patterns and learn them. Each hidden neuron is linked to one recurrent neuron with a steady weight of 1 and the number of recurrent neurons is equivalent to the number of hidden neurons. Hence, the recurrent layer is basically a copy of the hidden layer from a moment before [44] & [45]. Although [44], [45] proposed an ERNN based model for short-term traffic prediction, they haven’t addressed the delays of the traffic management systems. Given the presence of delays in real-world systems, this study uses a modified version of ERNN, which is described in later sections.

The search for the optimal network structure often involves a process of trial and error. Research on network structure optimization has suggested that genetic algorithms could serve a highly effective tool for finding near-optimal network structures [46] & [47]. The most important factors of interest in learning optimization process are the speed of convergence and the convergence to local minima. Genetic algorithm, as an archetype of evolutionary computing, has shown superior performance in solving complex, non-linear, and parallel problems without a priori information [48].

In recent years, many researchers have used genetic algorithms to optimize neural network structures in different applications including real-time applications: water temperature prediction [77], parking space prediction [78], cognitive map [79], spatial accuracy [80], and short-term traffic prediction [45], [50], [81]. Reference [49] used the genetic algorithm in specific neural network structures. They proved the genetic algorithm’s superiority over the backpropagation. Reference [50] developed a new method for short-term traffic prediction using advanced time delay neural network and then used a genetic algorithm to optimize it. Zhan Lijun combined the genetic algorithm and Elman recurrent neural network for use in the prediction of stock prices [51]. In another work, to optimize the weight and structure of Elman’s recurrent neural network and to get good predictions on Dongfeng Motor stock prices, Wang Tianee used genetic algorithms [52]. Zhang Xiuling employed a genetic algorithm to optimize the initialization and thresholds of Elman recurrent neural network and employed the developed model to predict the capacity of MH-Ni batteries [53]. In another work, the genetic algorithm was combined with multilayer perceptron for prediction of short-term traffic [54]. The research of [50], [54] focused on optimizing the neural networks to predict the short term traffic flow using spatial and temporal information, but
they have not considered other factors affecting on traffic, e.g., weather, weekday, and special events. Combining the genetic and neural network algorithms has been evaluated in other research fields. Kim et al. combined the genetic algorithm with the back-propagation network (BPN) model for use in cost prediction [55].

Considering different conditions influencing short-term traffic forecasting, such as work days and weekends [23], [36], weather [36], and special days (holidays) [13], [20], [39], [57], are crucial for accurate prediction of short-term traffic flow. Reference [31] proposed a hybrid algorithm using support vector regression, random forest, and genetic algorithms, while [33] proposed a LSTM method with feature enhancement to complements LSTM deficiencies at excessively long distances, but they only used weekday data and removed weekend data. In another work, a novel fuzzy-based convolutional neural network (F-CNN) approach for predicting more accurate traffic flow is suggested by applying a fuzzy approach which reflects the features of traffic accidents when first introduced uncertain information for road accidents in the CNN [36]. In addition, external factors were considered, including weather, wind speed, temperature, and weekday/weekend information, while the holiday condition was ignored in the study. Although the holiday conditions were addressed in some research, they have not considered holidays as an input. Reference [39] extends the applicability of the KNN technique to short-term traffic volume projections and offers a viable strategy to asymmetrical loss projections, evaluating the proposed technique in holidays and normal days. Reference [57] applied convolution neural network (CNN) deep learning to predict short-term traffic flow advocating weather and holiday periods as future works. Other researches are listed in Table 1.

In conclusion, several research studies have been conducted to predict short-term traffic flow using different factors and methods for prediction. Although the literature evaluated different NN models by considering some factors affecting traffic, this research will consider several factors together, GA-MENN model, and google maps data to predict traffic velocity. Most of the research considered temporal [61], [62] and spatial-temporal [18], [56]–[58], [60], [63] features of the vehicle velocity in the short term traffic prediction, but they lacked consideration of atypical conditions including the weather conditions, weekdays, hour and holidays. Some research addressed these factors; however, they have not been considered all together to predict the vehicle velocity. Among the four factors, the days’ classification is defined in this research with regards to the impact of weekends and holidays on the traffic. These parameters along with the vehicle velocities are models using the ERNN model modified with delay effects and the GA optimization algorithm to improve the accuracy and degrees of realism in the field of short-term traffic flow prediction, which has not be applied in this field based on the previous research. Finally, the traffic data of the google map API will be used to train and test the model to forecast short-term traffic follow that the previous research mostly has used GPS, sensor, or simulated traffic data. Google map data is a good source of traffic especially for countries that do not have suitable infrastructure to collect traffic data.

III. MATERIALS AND METHODS
A. DATA DESCRIPTION

Accurate prediction requires reliable traffic information about the area of interest, which was not always easily available. Nowadays, this information can be collected in several ways, for example by sensors and using online data transfer services. In recent years, a growing number of researchers have started to use Google Maps to collect traffic information. For example, [64] introduced a system based on Google Maps and Charts Application Programming Interfaces (API) to help traffic engineers identify traffic nodes without data mining. In another work, the Distance Matrix API of Google Maps was used to predict the urban traffic in Austin [65].

In this work, traffic data were collected from the Google Map API. This API provides various services in the form of map diagrams and route information (between 2 points). In this study, Direction API was used to collect route information, including travel time and route length. Using this API, fairly accurate information about the routes in different days of the week and different hours of the day were collected.

In total, data related to 139 routes in 7 districts in Tehran were collected. The routes were chosen to include 46 highways and 61 roads that had the most traffic throughout the day. In these routes, such as Hemmat highway, Imam Ali highway, and Bagheri expressway, the average speed during the day is between 30-50 km/h which is below the allowed speed range, 70-120 km/h. To achieve comprehensive results, 32 low traffic routes were also included, among them Golbarg street, Janbazan street, and Farjam street. According to the collected data, the average speed of these routes is 20-40 km/h which is below the maximum speed allowed of streets, 50 km/h. Figure 1 shows the network traffic and selected routs. A web-based service written in C# was used to automatically collect data every 30 minutes between 06:00 and 23:00 for 11 days and store them in a SqlServer Database. In the end, 56,741 records were obtained.

In addition to route information, traffic information, time, the weather data and the day’s classification were also added to the model. The half-hourly weather data collected from an online source (www.timeanddate.com) in four categories of Rainy, Clear, Cloudy, and Foggy were added to their corresponding records. Given the impact of weekends and holidays on the traffic in the preceding and following days, all days were divided into four classes (for easier comprehension, all weekend days and holidays, regardless of their nature, are simply referred to as non-working days). Figure 2 demonstrates the classes more clearly.

I- a working day that is followed by another working day, 
II- a working day that is followed by a non-working day, 
III-a non-working day that is followed by a working day,
TABLE 1. A summary of traffic flow prediction.

| Ref  | Temporal feature | Spatial feature | weather | Working days and weekend | Holiday | Parameter Prediction | Traffic data source       | Model                          | Optimization algorithm |
|------|------------------|-----------------|---------|--------------------------|---------|----------------------|---------------------------|-------------------------------|------------------------|
| [10] | ✓                | -               | -       | -                        | -       | Short-Term Freeway Traffic Volume | Data of TransGuide system | ARIMA                         | -                      |
| [13] | -                | -               | -       | ✓                        | -       | Short-Term Traffic Flow | Loop detectors | Online Support Vector Machine | -                      |
| [16] | -                | -               | -       | -                        | -       | Traffic Flow          | Simulation                  | A reinforcement learning approach | -                      |
| [17] | ✓                | -               | -       | ✓                        | -       | Short-Term Traffic Flow | Real data                  | Improved neurofuzzy models via emotional temporal difference learning algorithm | -                      |
| [18] | ✓                | ✓               | -       | -                        | -       | Overall traffic volume of a region | GPS                        | VOM model + probability suffix tree | -                      |
| [20] | -                | -               | -       | ✓                        | -       | Short-Term Traffic Flow | Loop detectors | Online learning weighted support vector regression | -                      |
| [23] | ✓                | -               | -       | ✓                        | -       | Travel speed          | Microwave sensors | K-means + Gaussian fuzzy membership function | -                      |
| [27] | ✓                | -               | -       | -                        | -       | Short-Term Traffic Flow | Loop detectors | Least squares support vector regression (LSSVR) + Gaussian kernel function | -                      |
| [31] | ✓                | -               | Weekdays | -                        | -       | Traffic Flow          | Sensor                     | SVR + Modified GA and a Random Forest | Genetic algorithm |
| [36] | ✓                | ✓               | ✓       | ✓                        | -       | Traffic Flow + Unknown traffic accident | Taxi cab trajectory | Fuzzy CNN                                           | -                      |
| [39] | -                | -               | -       | ✓                        | -       | Short-Term Traffic Flow | Underground electromagnetic soil device | Asymmetric Loss Based on Enhanced KNN | -                      |
| [44] | ✓                | ✓               | -       | -                        | -       | Short-Term Traffic Flow | Loop detector | Jordan-Elman networks, partially recurrent networks, and time-lagged feedforward networks. | -                      |
| [45] | ✓                | -               | -       | ✓                        | -       | Short-Term Traffic Flow | Loop detectors | A hybrid model-based and memory-based methodology | -                      |
| [50] | ✓                | ✓               | -       | -                        | -       | Short-Term Traffic Flow | Simulation                  | Neuro-Geneic Algorithms | Genetic algorithm |
| [54] | ✓                | ✓               | -       | -                        | -       | Short-Term Traffic Flow | Loop detectors | Meta-optimized neural networks | Genetic algorithm |
| [56] | ✓                | ✓               | -       | -                        | -       | Short-Term Traffic Flow | Simulation                  | Linear conditional Gaussian Bayesian network | -                      |
| [57] | ✓                | ✓               | -       | -                        | -       | Short-Term Traffic Flow | Loop detectors | Deep CNN                                           | -                      |
| [58] | ✓                | ✓               | -       | -                        | -       | Long-Term Traffic Flow | PeMS                       | Deep NN                          | -                      |
| [59] | ✓                | ✓               | ✓       | -                        | -       | Short-Term Traffic Flow | Loop detectors | deep learning architecture | -                      |
| [60] | ✓                | ✓               | -       | -                        | -       | Lane-level traffic speed | Microwave sensors | LSTM + GRUENN | -                      |
| [61] | ✓                | ✓               | ✓       | ✓                        | -       | Long-term travel time | Electronic toll collection (ETC) | Modified gradient boosting | -                      |
| [62] | -                | ✓               | -       | -                        | -       | Short-Term Traffic Flow | Simulation                  | Wavelet transformation + ANN | -                      |
| [63] | ✓                | ✓               | -       | ✓                        | -       | Short-Term Traffic Flow | PeMS                       | Adaptive multi-kernel SVM | PSO algorithm |
| Current research | ✓ | ✓ | ✓ | ✓ | ✓ | Travel Speed | Google Map API | GA-MERNN | Genetic algorithm |

IV-a non-working day that is followed by another non-working day.

The day’s class in terms of working/non-working status was also added to the records. For the sake of simplicity, the records were adjusted to have speed instead of time wherever possible. This was done by using the formula $v = \frac{x}{t}$, where $v$ is the velocity, $x$ is the rout distance, and $t$ is the time to compute the velocity in each route. So, we calculated the speed for each route. Figure 3 displays the average speed in different weather conditions. The highest traffic volume is related to rainy weather and the lowest is related to foggy weather. Figure 4 shows the average speed at different days of the week. As can be seen, the highest traffic volume (lowest speed = 31.45 km/h) has been recorded in Tuesdays (traffic versus time of the day is plotted for 20 random routes). Figure 5 illustrates the average speed versus the day’s working/non-working status. As this figure shows, days of class II have had the most traffic and days of class III have had the least. Figure 6 shows the average speed at different hours of the day. It can be seen that the hours between 17:00 and 20:30 have had the highest amount of traffic and the hours between 06:00 and 08:00 have had the lowest traffic [45]. Figure 7 demonstrates the average speed in different routes. As depicted in the figure, 8% of routes have the highest traffic...
39% of them have the average traffic flow and 53% of all have the lowest traffic congestion.

Figure 8 shows scatterplots of sample data to represent patterns of data. Figure 8a plots sample velocities versus routes and weekdays. It shows the velocity was changing from around 15 to 80 at different days of the week. Figure 8b displays the patterns of velocities in different routes and weather conditions. As weather seldom is foggy in Tehran, the sample data in foggy condition is too small compared to other weather conditions. Figure 8c demonstrates the velocity patterns versus routes and hours of a day. Its shows that the hours between 6:00 to 7:00 haven’t recorded the velocity below 20 km/h except some road ID such as 38, 68, 80, 101, and 138.

Figure 9 shows scatterplots of predicted velocities by GA-MENN. Comparing the scatter plots of the predicted
velocities and the sample data (Fig. 8) reveals that the patterns the predicted velocities and input velocities are pretty similar, as expected. Therefore, GA-MENN is suitable model to predict the traffic based on the velocity changes at different routes, different weather conditions and different days of the week.

B. GA-MENN METHOD
This article introduces an algorithm called GA-MENN, which is developed by combining the Elman recurrent neural network with the genetic algorithm. The work attempts to address the fact that in many dynamic systems, the effects of influential phenomena may appear with a delay, and this delay can have a significant impact on the accuracy of analysis and results. The best way to deal with this issue is to consider the delays in the Elman recurrent neural network [66]. In this work, a modified Elman recurrent neural network model is used in combination with the genetic algorithm for short-term traffic flow prediction. GA is used to optimize the parameters of MERNN. GA improves the generalization feature of NN and then the prediction accuracy [82], [83]. Although GA does not guarantee finding a global optimum solution [84], this algorithm compensates for the limitations of static property given by NN models and inclines the training process to fall into a near-optimal solution [82]. In addition, GA makes it possible to model complex nonlinear functions using NN models [85].

Figure 10 illustrates the flowchart of the GA-MENN model. The steps of the proposed procedure are summarized as follows:
1. Feature selection. The original data is supposed to have features. The main part analysis is used to obtain the parts with a total commitment rate of over 90%, thus giving the dataset $S = \{(x_k, y_k) | k = 1, 2, \ldots, n\}$, in which $x_k \in \mathbb{R}^r$ and $y_k \in \mathbb{R}^s$ and $n$ is the number of data.
2. Dataset $S$ is partitioned into training subset $S_1 = \{(x_k, y_k) | k = 1, 2, \ldots, q\}$ and test subset $S_2 = \{(x_k, y_k) | k = q+1, q+2, \ldots, n\}$ and normalized.
3. The modified ERNN model is used to predict travel time. The structure of this modified Elman recurrent neural network is illustrated in Figure 11. In the changed ERNN model, the signal for input $(u(k - r))$ specifically incorporates the pure delay time (while in the old version of structure,
the neural model itself has to approach the delay in the dynamic model). The weight of the network is the same as in the classical ERNN structure. The ERNN modified model output is as follows [66]:

\[ y_{\text{mod}}(k) = \sum_{o=1}^{r} b_o + \sum_{i=0}^{k} W^{(2)}_i x_i(k) \]  

where \( x_i(k) \) is the following hidden nodes output signal, \( i = 1, \ldots, K \), \( x_0(k) = 1 \), and \( b_h \) and \( b_o \) are the biases of the hidden units and output units. Also,

\[ X_i(k) = \varphi(Z_i(k)) \]

where the sum of the hidden nodes \( i \) is as follows:

\[ z_i(k) = b_h + \sum_{i=0}^{m} w_{i,1} u(k - \tau) + \sum_{j=1}^{n} w^{(1)}_{i,j+1} x_j(k - 1) \]

After combining the above equations, the final equation of the modified ERNN model will be

\[ y_{\text{mod}}(k) = \sum_{o=1}^{r} b_o + \sum_{h=1}^{n} w^{(2)}_h \varphi(b_h) + \sum_{i=0}^{m} w^{(2)}_{i,1} u(k - \tau) + \sum_{j=1}^{n} w^{(1)}_{i,j+1} x_j(k - 1) \]  

4. The genetic algorithm is then used to optimize the modified ERNN model through the following process.

1. Population initialization: The weights and thresholds in the definition of chromosomes should be inserted in order to optimize ERNN using GA. The N chromosome population is produced randomly here. The initial weights and thresholds of ERNN are composed of every gene of the chromosome. Below is the chromosome’s floated-point coding length:

\[ K = P \ast m + P \ast n + n + P \ast P \]
2. Fitness function: Each chromosome in the population is evaluated for its fitness. The formula used for this purpose is

\[ f = \frac{1}{N} \sum_{i=0}^{N} \sum_{j} (y'_{j,i} - y_{j,i})^2 \]  

where \( N \) is the number of chromosomes in the population; \( y'_{j,i} \) and \( y_{j,i} \) are the predicted and observed speed in the current route; and \( T \) is the amount of neurons output, that is set to 1 as speed is only variable being predicted. At this point, the minimum fitness value is determined. If it is lower than the overall fitness minimum value, the global fitness minimum value is updated.

3. Selection, crossover, and mutation of the parent population to create the next-generation population.

a) Selection operation: After each chromosome has calculated its fitness, the new population is selected with several chromosomes. The chance of selecting a chromosome depends on its fitness. Given the formulation used in this study, the low fitness value chromosomes are chosen and the one with high fitness values are discarded. The roulette wheel selection operation is applied as follows:

\[ p_i = \frac{F_i}{\sum_{j=1}^{N} F_j} \]  

where \( F_j \) is the fitness value and \( p_i \) is the chance of chromosome \( i \) being chosen.

b) Crossover operation: For the crossover process due to the actual number coding, the real number crossover method is used. The chromosome \( c_k \) and chromosome \( c_i \) crossover operation in \( j \) position are as follows:

\[
C_{kj} = C_{kj}(1 - r) + C_{ij}r, \\
C_{ij} = C_{ij}(1 - r) + C_{kj}r
\]  

Here, \( c_{kj} \) and \( c_{ij} \) are the \( k \)th and \( i \)th values in the chromosome \( j \), representing the ERNN weight or threshold, and \( r \) is a 0 to 1 number.

C) Mutation operation: The mutation operation role is to diversify the population. The \( j \)th gene mutation in the chromosome \( i \)th (\( m_{ij} \)) is defined as follows:

\[
m_{ij} = \begin{cases} 
m_{ij} + (m_{ij} - m_{\text{max}}) \cdot f(g), & r \geq 0.5, \\
m_{ij} + (m_{\text{min}} - m_{ij}) \cdot f(g), & r < 0.5 \end{cases}
\]  

where \( m_{\text{min}} \) and \( m_{\text{max}} \) are the minimum and maximum value of weight or threshold in the \( j \)th gene of the \( i \)th chromosome, and \( r \) is a random number between 0 and 1. \( f(g) \) is given by

\[ f(g) = r_1(1 - \frac{g}{G}) \]
A. Sadeghi-Niaraki et al.: Short-Term Traffic Flow Prediction Using the Modified ERNN Optimized Through a GA

where $r_1$ is an accidental number, $g$ represents the current iteration number, and $G$ presents the evolutionary generation value.

5. When the evolutionary generation value reaches a maximum, the process will stop and the latest weight and threshold values will be extracted. Otherwise, steps (b) and (c) will be repeated.

6. The obtained weight and threshold are applied to ERNN, and after training, it will be able to reach the desired accuracy or condition.

**TABLE 2. The results of models’ comparison.**

| Road Segment ID | Week Day | Time | Weather Status | Holiday Status | Velocity |
|-----------------|----------|------|----------------|---------------|----------|
| 33              | 2        | 18   | 1              | 1             | 62.557   |
| 3               | 1        | 14.5 | 2              | 1             | 35.540   |
| 17              | 3        | 19.5 | 3              | 1             | 37.657   |
| 124             | 7        | 23.5 | 2              | 0             | 19.627   |
| 51              | 1        | 9    | 2              | 2             | 63.2     |
| 51              | 2        | 11   | 2              | 1             | 69.945   |
| 46              | 2        | 20   | 2              | 1             | 58.521   |
| 96              | 1        | 14   | 2              | 3             | 29.332   |
| 81              | 1        | 17.5 | 2              | 1             | 44.530   |
| 159             | 2        | 7    | 2              | 1             | 26.64    |

**IV. RESULTS AND DISCUSSION**

**A. ANALYSIS OF THE PREDICTION RESULTS**

The data obtained from the Google Map API was used to measure the performance of the proposed model alone and in comparison with several other prediction models. Table 2 shows the sample of data that was used for training, evaluating and testing the prediction models. In the table, the integer number from 1 to 7 were used to represent weekdays, Saturday to Friday. For weather status, the numbers 1, 2, 3, and 4 refers the rainy, clear, cloudy, and foggy weather, respectively. The time was calculated in the hour format. The holiday status was represented with 0 (class I), 1 (class II), 2 (class III), and 3 (class VI).

In this evaluation, 60% of the records were used for training, 15% of them were used for validation, and 25% were used for testing. Figure 12 displays the proposed model outputs in comparison with the actual records for the training set. Figure 13 makes a similar comparison for the records of the validation set. In Figure 14, the outputs of the model are compared with the observed values in the testing set.

As is apparent, the proposed model has a high degree of accuracy, as its predictions are very close to real values. This will be discussed in detail in the next section. In addition, the optimized weights and thresholds allow the model to overcome the problem of falling into local minima and enhance its ability to trace optimal solutions, which significantly improves the reliability of predictions.

**B. ERROR ANALYSIS**

The metrics used in the evaluation of the proposed model are the modified mean absolute percent error (MMAPE) [75], Root Mean Square Error (RMSE) [68], and Mean Absolute Error (MAE) [68] with the following formulas:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} |E_i - M_i|^2}{n}}$$

$$\text{MAPE} = \frac{\sum_{i=1}^{n} |E_i - M_i|}{\sum_{i=1}^{n} |E_i + M_i|}$$

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |E_i - M_i|$$

where $E_i$ denotes the predicted values, $M_i$ denotes the observed values, $\bar{E}$ and $\bar{M}$ are the means of $E_i$ and $M_i$, and $n$ is the total number of records. RMSE is used to evaluate the performance of the traffic flow prediction models. Indisputably, the smaller the RMSE, the more accurate the forecasting [27]. MMAPE represents the error between expected values and real values and is typically appropriate for the measurement of non-linear large-scale data sets [31]. The MAE measures
the resemblance between the anticipated vehicle velocity and the real vehicle velocity [40].

For a better evaluation of the model, its performance was compared against regression multi-layer perceptron, linear regression, logistic regression, probabilistic neural network, regression, generalized feedforward, time-lag recurrent network, support vector machine model, Elman neural network, K-NN model, ARIMA, SARIMA, Kalman filter model, Convolutional Neural Networks (CNNs), and Long Short-Term Memory (LSTM) model. KNN is one of the easiest pattern identification techniques. The predictive stage is slow since the distance to each training data point needs to be calculated. Reducing the amount of features or using several prototypes could improve kNN’s speed [69]. As a predictive model, the ARIMA model was widely utilized. For data with a brief range recurring pattern, this model is great [70]. Kalman Theory is used widely in traffic monitoring systems. This technique do not ensure complete precision, but is the most used technique for estimating traffic systems [71]. LSTM neural networks learns the time series for long periods and determines the optimal time lag for prevision automatically [72]. The CNNs have demonstrated their ability to extract spatial characteristics in a local or urban region [73]. The results of this comparison are presented in Table 3.

In addition, the model also evaluated using the time series models including Seasonal ARIMA (SARIMA). SARIMA is a time-series prediction model for periodic time series and the seasonal difference data especially in the traffic speed time series [74]. As the SARIMA formula has one time-dependent input, this model was implemented using the time and velocity parameters.

As shown in Table 3, the GA-MENN model has the best performance among the compared models. The second best performance belongs to generalized feedforward regression model with 2.04939 units higher RMSE and 2.32369 units higher MAE than the best model. Since the CNN and LSTM
models are more appropriate for big data and our records are not enough, the worst model, with the worst validation, training, and testing RMSE, MMAPE, and MAE, is CNN. Besides all the models MMAPE are more than 50 except the GA-MENN which means the GA-MENN has the minimum error between the expected value and the real value. For a closer look, we used Elman neural network in our survey as one the compared model. The RMSE, MMAPE, and MAE of GA-MENN in training, validating, and testing part are less than those of the Elman neural network model. The training RMSE, MMAPE, and MAE of Elman neural network are respectively 2.891, 6.41, and 3.176 higher than those of GA-MENN. Moreover, the validation RMSE, MMAPE, and MAE of Elman neural network are 17.355, 54.24, and 14.448 which are greater than the validation RMSE, MMAPE, and MAE of GA-MENN. The testing RMSE, MMAPE, and MAE of Elman neural network are 17.482, 53.48, and 14.568, whilst the testing RMSE, MMAPE, and MAE of GA-MENN are 14.633, 47.65, and 11.434. Last of all, the RMSE, MMAPE, and MAE of time series model (SARIMA) for the training, validation, and testing are higher than those of GA-MENN. Based on [59], the RNN and deep learning methods including LSTM are two types of the networks that have higher performance in time series prediction with ability of motorizing previous data. The previous studies that have proposed deep learning methods ([57], [59], [60]) have successfully confirmed the performance of deep learning networks to predict short-term traffic flow using spatial-information features of the vehicle velocity. The results of this research confirmed the good performance of the GA-MENN model that is based on RNN using the various input variable (including spatial-temporal information, weather, working/weekend days, and holiday status), especially when the volume of available data are not large, due to better training, evaluating and testing precision of the GA-MENN model compared to those of the CNN and LSTM methods.

For a closer examination, the following formula was used to calculate the mean square error (MSE) over different epochs for the training and validation datasets. The results are plotted in Figure 13.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (E_i - M_i)^2$$  \hspace{1cm} (14)

In Figure 15, it can be seen that from the Epoch30 onwards, MSE has decreased at a slower pace and has approached its final value. Overall, it can be claimed the proposed model is
certainly among the best models for short-term traffic prediction and exhibits an excellent performance under variable conditions.

V. CONCLUSION AND FUTURE WORK
In our paper, we propose the GA-MENN model to enhance the accuracy of short-term traffic flow prediction with consideration of climate conditions, day’s status and weekday and hour. Our paper centers around the forecasting of traffic flow, however, we incorporate the data of actual weather features, weekday, and day’s classes in our models. First, we compare our GA-MENN models with classical machine learning techniques both parametric and non-parametric, including Regression Multi-layer Perceptron, Linear Regression, Logistic Regression, Probabilistic Neural Network, Regression Generalized Feedforward, Time-lag Recurrent Network, Support Vector Machine model, Elman neural network, K-NN model, ARIMA, Kalman filter model, Convolutional Neural Networks (CNNs), and Long Short-Term Memory (LSTM) model. Our model reaches the lowest error rate based on three evaluation criteria: MMAPE, MAE and RMSE. The prediction precision improved up to 12.3% in comparison to the best model among them (Kalman Filter) with the same hyper parameters. In addition, the lowest RMSE error among all models is achieved. Accordingly, we suppose that our paper could be valuable for the modeling, analysis and mining of large-scale traffic data and short-term traffic prediction.

Comparing the results showed improvement of GA-MENN algorithm in traffic prediction compared to the ENN algorithm, about 16.7% on average in the prediction precision. This result indicates the merit of the genetic based modified algorithm for improving the ENN model in the traffic flow prediction.

Future works should address the need for more advanced architectures for prediction of short-term traffic flow. Also, more sophisticated machine learning models can be used in other sub-branches of transportation research, including prediction and identification of accidents. Moreover, researchers can take advantage of other deep learning methods such as Deep Reinforcement Learning model. Considering the low cost of Google services, it is possible to spend more on purchasing more extensive data for developing more accurate models. Researchers can also try combinations with other machine learning models in order to reach more sophisticated and more widely applicable models.

REFERENCES
[1] T. G. Rad, A. Sadeghi-Niaraki, A. Abbasi, and S.-M. Choi, “A methodological framework for assessment of ubiquitous cities using ANP and DEMATEL methods,” Sustain. Cities Soc., vol. 37, pp. 608–618, Feb. 2018.
[2] M. Sharif and A. Sadeghi-Niaraki, “Ubiquitous sensor network simulation and emulation environments: A survey,” J. Netw. Comput. Appl., vol. 93, pp. 150–181, Sep. 2017.
[3] W.-H. Lee, S.-S. Tseng, J.-L. Shieh, and H.-H. Chen, “Discovering traffic bottlenecks in an urban network by spatiotemporal data mining on location-based services,” IEEE Trans. Intell. Transp. Syst., vol. 12, no. 4, pp. 1047–1056, Dec. 2011.
[4] D. Billings and J.-S. Yang, “Application of the ARIMA models to urban roadway travel time prediction—A case study,” in Proc. IEEE Int. Conf. Syst., Man Cybern., Oct. 2006, pp. 2529–2534.
[5] B. M. Williams. “Multivariate vehicular traffic flow prediction: Evaluation of ARIMAX modeling,” Transp. Res. Rec., J. Transp. Res. Board, vol. 1776, no. 1, pp. 194–200, Jan. 2001.
[6] P. Gao, J. Hu, H. Zhou, and Y. Zhang, “Travel time prediction with immune genetic algorithm and support vector regression,” in Proc. 12th World Congr. Intell. Control Automat. (WCICA), Jan. 2016, pp. 987–992.
[7] S. Oh, Y.-J. Byon, R. H. Park, R. H. Park, “Short-term traffic prediction on highway: A review on model-based approach,” KSCE J. Civ. Eng., vol. 22, no. 1, pp. 298–310, Jan. 2018.
[8] A. Stathopoulos, L. Dimitriou, and T. Tsekeris, “Fuzzy modeling approach for combined forecasting of urban traffic flow,” Comput.-Aided Civil Infrastruct. Eng., vol. 23, no. 7, pp. 521–535, Oct. 2008.
[9] S. A. Ahmed and A. R. Cook, Analysis of Freeway Traffic Time-Series Data by Using Box-Jenkins Techniques, vol. 20, no. 6, pp. 2395–2400, Jun. 2019.
[10] B. M. Williams, P. K. Durvasula, and D. E. Brown, “Urban freeway traffic flow prediction: Application of seasonal autoregressive integrated moving average and exponential smoothing models,” Transp. Res. Rec., J. Transp. Res. Board, vol. 1644, no. 1, pp. 132–141, Jan. 1998.
[11] X. Li, C. Wang, and H. Shi, “A travel time prediction method: Bayesian reasoning state-space neural network,” in Proc. 2nd Int. Conf. Inf. Sci. Eng., Dec. 2010, pp. 936–940.
[12] M. Jun and M. Ying, “Research of traffic flow forecasting based on neural network,” in Proc. 2nd Int. Symp. Intell. Inf. Technol. Appl., Dec. 2008, pp. 104–108.
[13] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, “Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions,” Expert Syst. Appl., vol. 36, no. 3, pp. 6164–6173, Apr. 2009.
[14] I. M. Keller, M. R. Gray, and J. A. Givens, “A fuzzy K-nearest neighbor algorithm,” IEEE Trans. Syst., Man, Cybern., vol. 25, no. 4, pp. 580–585, Aug. 1985.
[15] C. G. Atkeson, “Nonparametric model-based reinforcement learning,” in Proc. Adv. Neural Inf. Process. Syst. (NIPS) Conf., Denver, CO, USA, Nov./Dec. 1998.
[16] E. Walraven, M. T. J. Spaan, and B. Bakker, “Traffic flow optimization: A reinforcement learning approach,” Eng. Appl. Artif. Intell., vol. 52, pp. 203–212, Jun. 2016.
[17] J. Abdil, B. Moshiri, B. Abdullahi, and A. K. Sedigh, “Forecasting of short-term traffic-flow based on improved neurofuzzy models via emotional temporal difference learning algorithm,” Eng. Appl. Artif. Intell., vol. 25, no. 5, pp. 1022–1042, Aug. 2012.
[18] C. Yuan, X. Yu, D. Li, and Y. Xi, “Overall traffic mode prediction by VOOM approach and AR mining method,” IEEE Trans. Intell. Transp. Syst., vol. 20, no. 4, pp. 1508–1516, Apr. 2018.
[19] Y. Lin, X. Dai, L. Li, and F.-Y. Wang, “Pattern sensitive prediction of traffic flow based on generative adversarial framework,” IEEE Trans. Intell. Transp. Syst., vol. 20, no. 6, pp. 2395–2400, Jun. 2019.
[20] Y.-S. Jeong, Y.-J. Byon, M. M. Castro-Neto, and S. M. Easa, “Supervised weighting-online learning algorithm for short-term traffic flow prediction,” IEEE Trans. Intell. Transp. Syst., vol. 14, no. 4, pp. 1700–1707, Dec. 2013.
[21] G. Zhang, B. E. Patuwo, and M. Y. Hu, “Forecasting with artificial neural networks:: The state of the art,” Int. J. Forecasting, vol. 14, no. 1, pp. 35–62, 1998.

[22] W. Zheng, D.-H. Lee, and Q. Shi, “Short-term freeway traffic flow prediction: Bayesian combined neural network approach,” J. Transp. Eng., vol. 132, no. 2, pp. 61–62, Feb. 2006.

[23] J. Tang, F. Liu, Y. Zou, W. Zhang, and Y. Wang, “An improved fuzzy neural network for traffic speed prediction considering periodic characteristic,” IEEE Trans. Intel. Transp. Syst., vol. 18, no. 9, pp. 2340–2350, Sep. 2017.

[24] J. W. C. van Lint, S. P. Hoogendoorn, and H. J. van Zuylen, “Accurate free-way travel time prediction with state-space neural networks under missing data,” Transp. Res. C, Emerg. Technol., vol. 13, nos. 5–6, pp. 347–369, Oct. 2005.

[25] D. Lien Minh, A. Sadeghi-Niaraki, H. D. Huy, K. Min, and H. Moon, “Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network,” IEEE Access, vol. 6, pp. 55392–55404, 2018.

[26] X. Ma, H. Yu, Y. Wang, and Y. Wang, “Large-scale transportation network congestion evolution prediction using deep learning theory,” Pla ONE, vol. 10, no. 3, Mar. 2015, Art. no. e0119044.

[27] X. Chen, X. Li, F. Sang, F. Liu, and S. Liu, “Ensemble learning multiple LSSVR with improved harmony search algorithm for short-term traffic flow forecasting,” IEEE Access, vol. 6, pp. 9347–9357, 2018.

[28] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, “Ensemble learning for data stream analysis: A survey,” Inf. Fusion, vol. 37, pp. 132–156, Sep. 2017.

[29] L. Yu, H. Xu, and L. Tang, “LSSVR ensemble learning with uncertain parameters for crude oil price forecasting,” Appl. Soft Comput., vol. 56, pp. 692–701, Jul. 2017.

[30] M. Effati and A. Sadeghi-Niaraki, “A semantic-based classification and regression tree approach for modelling complex spatial rules in motor vehicle crashes domain,” Wiley Interdiscip. Rev., Data Mining Knowl. Discovery, vol. 5, no. 4, pp. 181–194, Jul. 2015.

[31] L. Zhang, N. R. Alharbe, G. Luo, Z. Yao, and Y. Li, “A hybrid forecasting framework based on support vector machine with a modified genetic algorithm and a random forest for traffic flow prediction,” Tsinghua Sci. Technol., vol. 23, no. 4, pp. 479–492, Aug. 2018.

[32] W.-C. Hong, Y. Dong, F. Zheng, and S. Y. Wei, “Hybrid evolutionary algorithms in a SVR traffic flow forecasting model,” Appl. Math. Comput., vol. 217, no. 15, pp. 6733–6747, Apr. 2011.

[33] B. Yang, S. Sun, J. Li, X. Lin, and Y. Tian, “Traffic flow prediction using LSTM with feature enhancement,” Neurocomputing, vol. 332, pp. 33–37, Mar. 2019.

[34] C. Luo, C. Huang, J. Cao, J. Lu, W. Huang, J. Guo, and Y. Wei, “Short-term traffic flow prediction based on least square support vector machine with hybrid optimization algorithm,” Neural Process. Lett., vol. 50, pp. 2305–2322, Mar. 2019.

[35] J. Tang, X. Chen, Z. Hu, F. Zong, C. Han, and L. Li, “Traffic flow prediction based on combination of support vector machine and data denoising schemes,” Phys. A, Stat. Mech. Appl., vol. 534, Nov. 2019, Art. no. 120642.

[36] J. An, L. Fu, M. Hu, W. Chen, and J. Zhan, “A novel fuzzy-based convolutional neural network method to traffic flow prediction with uncertain traffic accident information,” IEEE Access, vol. 7, pp. 20708–20722, 2019.

[37] Y.-L. Hu and L. Luo, “DP-LRT: An urban short-term traffic speed forecasting method based on data driven,” in Proc. Int. Conf. Comput. Sci., Commun. Big Data (ICCSD’19), Beijinge, China, Mar. 2019 [Online]. Available: http://www.dpi-proceedings.com/index.php/dtce/article/view/29998, doi: 10.12783/dtce/ccecd19/29998.

[38] T. Zhang, Y. Liu, Z. Cui, J. Leng, W. Xie, and L. Zhang, “Short-term traffic congestion forecasting using attention-based long short-term memory recurrent neural network,” in Proc. 19th Int. Conf. Comput. Sci. (ICCS), Faro, Portugal, Jun. 2019, pp. 304–314.

[39] Z. Wang, S. Ji, and B. Yu, “Short-term traffic volume forecasting with asymmetric loss based on enhanced KNN method,” Math. Problems Eng., vol. 2019, pp. 1–11, Apr. 2019.

[40] F. G. Habtemichael and M. Cetin, “Short-term traffic flow rate forecasting based on identifying similar traffic patterns,” Transp. Res. C, Emerg. Technol., vol. 66, pp. 61–78, May 2016.

[41] B. L. Smith and M. J. Demetseky, “Short-term traffic flow prediction modeling—a comparison of neural network and nonparametric regression approaches,” in Proc. IEEE Int. Conf. Symp. Man Cybern., Oct. 1994, pp. 1706–1709.
[65] T. Kodani and S. Saito, “Exploring Google maps API as a tool for measuring regional and overall traffic,” Google, New York, NY, USA, Tech. Rep., 2017.

[66] A. Wysocki and M. Lawrytczuk, “Elman neural network for modeling and predictive control of delayed dynamic systems,” Arch. Control Sci., vol. 26, no. 1, pp. 117–142, Mar. 2016.

[67] P. Lingras and P. Mountford. “Time delay neural networks designed using genetic algorithms for short-term inter-city traffic forecasting,” in Proc. IEA/AIE, LNCS, vol. 2070, 2001, pp. 290–299.

[68] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, “Extreme learning machine: Theory and applications,” Neurocomputing, vol. 70, nos. 1–3, pp. 489–501, Dec. 2006.

[69] E. A. Aydinin, Introduction to Machine Learning (Adaptive Computation and Machine Learning Series). Cambridge, MA, USA: MIT Press, 2004.

[70] V. B. Gavirangaswamy, C. Gupta, A. Gupta, and R. Agrawal, “Assessment of ARIMA-based prediction techniques for road-traffic volume,” in Proc. 5th Int. Conf. Manage. Emergent Digit. EcoSyst. (MEDES), New York, NY, USA: ACM, Oct. 2103, pp. 246–251.

[71] D. McHugh. “Traffic prediction and analysis using a big data and visualisation approach,” Dept. Comput. Sci., Inst. Technol. Blanchardstown, Dublin, Ireland, Tech. Rep., 2015. [Online]. Available: http://www.dataminingsmasters.com/uploads/studentProjects/TrafficAnalysis.pdf

[72] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, “Long-term short-term memory neural network for traffic speed prediction using remote microwave sensor data,” Transp. Res. C, Emerg. Technol., vol. 54, pp. 187–197, May 2015.

[73] H. Yu, Z. Wu, S. Wang, Y. Wang, and X. Ma, “Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks,” Sensors, vol. 17, no. 7, pp. 1501, Jun. 2017.

[74] Z. Song, Y. Guo, Y. Wu, and J. Ma, “Short-term traffic speed prediction under different data collection time intervals using a SARIMA-SDGM hybrid prediction model,” PLoS ONE, vol. 14, no. 6, Jun. 2019, Art. no. e0218626.

[75] F. Xiao, D. Zhang, G. Kou, and L. Li, “Learning spatiotemporal features of ride-sourcing services with fusion convolutional network,” 2019, arXiv:1904.06823. [Online]. Available: http://arxiv.org/abs/1904.06823

[76] S. Şeker, E. Ayaz, and E. Türkcan, “Elman’s recurrent neural network applications to condition monitoring in nuclear power plant and rotating machinery,” Eng. Appl. Artif. Intell., vol. 16, nos. 7–8, pp. 647–656, Oct. 2003.

[77] S. Stajkowski, D. Kumar, P. Samui, H. Bonakdari, and B. Gharagabhi, “Genetic-Algorithm-Optimized sequential model for water temperature prediction,” Sustainability, vol. 12, no. 13, p. 5374, Jul. 2020.

[78] J. Qiu, J. Tian, H. Chen, and X. Lu, “Prediction method of parking space based on genetic algorithm and RNN,” in Proc. Pacific Rim Conf. Multimedia, Cham, Switzerland: Springer, Sep. 2018, pp. 865–876.

[79] A. Dunea, T. Puyalnithi, M. V. Vankadara, and N. Chilamkurti, “Analysis of inter-concept dependencies in disease diagnostic cognitive maps using recurrent neural network and genetic algorithms in time series clinical data for targeted treatment,” J. Ambient Intell. Humanized Comput., vol. 10, no. 10, pp. 3915–3923, Oct. 2019.

[80] N. Mohammadi and M. Malek, “Artificial intelligence-based solution to estimate the spatial accuracy of volunteered geographic data,” J. Spatial Sci., vol. 60, no. 1, pp. 119–135, Jan. 2015.

[81] J. Chen, H. Xing, H. Yang, and L. Xu, “Network traffic prediction based on LSTM networks with genetic algorithm,” in Proc. Int. Conf. Signal Inf. Process., Netw. Comput., Singapore: Springer, Nov. 2018, pp. 411–419.

[82] S. Chong, S. Rui, L. Jie, Z. Xiaoming, T. Jun, S. Yunbo, L. Jun, and C. Huiliang, “Temperature drift modeling of MEMS gyroscope based on genetic-ELM neural network,” Mech. Syst. Signal Process., vols. 72–73, pp. 897–905, May 2016.

[83] W. Min, W. Jiawei, G. Jinhui, S. Lihua, and A. Bogong, “Multi-point prediction of aircraft motion trajectory based on GA-Elman-Regularization neural network,” Integr. Ferroelectr., vol. 210, no. 1, pp. 116–127, Sep. 2020.

[84] S. Bouktif, A. Fiaz, A. Ouni, and M. Serhani, “Optimal deep learning LSTM model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches,” Energies, vol. 11, no. 7, p. 1636, Jun. 2018.

[85] H. Chung and K.-S. Shin, “Genetic algorithm-optimized multi-channel convolutional neural network for stock market prediction,” Neural Comput. Appl., vol. 32, no. 12, pp. 7897–7914, Jun. 2020.

ABOLGHASEM SADEGHI-NIARAKI received the B.Sc. degree in geomatics engineering and the M.Sc. degree in geospatial information system (GIS) engineering from the Department of GIS Engineering, Faculty of Geomatics Engineering, K. N. Toosi University of Technology (KNTU), and the Ph.D. degree in geoinformatics engineering from Inha University, South Korea, in 2008. He was the GIS Project Manager of IT Office, Road Maintenance Transportation Organization, Road Ministry, Tehran. He has a variety of teaching experiences at several universities (KNTU, Inha, and Sejong University). He is currently an Assistant Professor. He has been teaching and supervising undergraduate and graduate students. He has developed a strong background in leadership, sustainability, and collaborative research in science and technology using mixed research methodology between Iran and South Korea. He has a strong publication record and has published various peer-reviewed research articles in reputed international journals. His research interests include the AI, XR (MR, AR, and CR), ubiquitous GIS, the IoT, blockchain, smart cities, culture technology (CT), ontology, indoor localization, and drone.

MARYAM SHAKEHI received the bachelor's degree in computer engineering from the Sharif University of Technology, Iran, in December 2012, and the M.B.A. degree from Alzahra University, Tehran, Iran, in 2015. She is currently pursuing the Professional Doctorate in Engineering degree with the Eindhoven University of Technology. Her research interests include geoinformatics and machine learning.

SOO-MI CHOI received the B.S., M.S., and Ph.D. degrees in computer science and engineering from Ewha University, South Korea, in 1993, 1995, and 2001, respectively. She was a Visiting Researcher with the Fraunhofer Institute for Computer Graphics Research IGD in Germany, in 1998, and a Research Professor with the Center for Computer Graphics and Virtual Reality, Ewha University, in 2001. She joined the Faculty of Computer Science and Engineering, Sejong University, South Korea, in 2002. From 2008 to 2009, she was a Visiting Professor with the CG Laboratory, ETH Zurich, Switzerland. She is currently a Full Professor of computer science and engineering with Sejong University and also the Director of the Mobile Virtual Reality Research Center (ITRC), designated by the Korean Ministry of Science and ICT. Her current research interests include virtual and augmented reality, computer graphics, and human–computer interaction. She was the Vice-President of the HCI Society of Korea. She is also the Vice-President of the Korean Institute of Information Scientists and Engineers and the Korea Computer Graphics Society.