A Multitask Learning Model for Traffic Flow and Speed Forecasting

KUNPENG ZHANG1,2, LAN WU1,2, ZHAOJU ZHU3, AND JIANG DENG4
1Key Laboratory of Grain Information Processing and Control, Ministry of Education, Henan University of Technology, Zhengzhou 450001, China
2College of Electrical Engineering, Henan University of Technology, Zhengzhou 450001, China
3School of Mechanical Engineering and Automation, Fuzhou University, Fuzhou 350108, China
4FinSight AI Lab, Qingdao Fantaike Bearing Company, Ltd., Qingdao 266000, China
Corresponding authors: Lan Wu (wulan@haut.edu.cn) and Zhaoju Zhu (zhuzhaoju0216@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61973103.

ABSTRACT Intelligent Transportation Systems (ITS) research and applications benefit from accurate short-term traffic state forecasting. To improve the forecasting accuracy, this paper proposes a deep learning based multitask learning Gated Recurrent Units (MTL-GRU) with residual mappings. To enhance the performance of the MTL-GRU, feature engineering is introduced to select the most informative features for the forecasting. Then, based on real-world datasets, numerical results show that the MTL-GRU can well estimate traffic flow and speed simultaneously, and performs better than other counterparts. Experiments also show that the deep learning based MTL-GRU model can overpower the bottleneck caused by enlarging training datasets and continue to gain benefits. The results suggest the proposed MTL-GRU model with residual mappings is promising to forecast short-term traffic state.

INDEX TERMS Short-term traffic forecasting, deep learning, multitask learning, feature engineering.

I. INTRODUCTION
In Intelligent Transportation Systems (ITS), short-term traffic state forecasting aims at anticipating traffic conditions based on historical observations. Timely traffic state forecasting is crucial for the planning and development of the traffic management and control system [1]–[3]. In this context, improving forecasting accuracy would be of extreme importance.

In recent years, as public infrastructure and data transmission technology advance, more traffic data including traffic flow, traffic speed, traffic incidents, and social events are available, which is beneficial to promote forecasting accuracy. Although a number of models have been developed, many of them leverage conventional methods that may be unsatisfying to penetrate the deep correlation hidden in large datasets. Consequently, forecasting accuracy cannot profit from sharply increasing traffic data. Therefore, new techniques are eagerly demanded to handle the abundant traffic data at a deep level.

As a subset of machine learning, deep learning has drawn tremendous interest from academic and industrial fields. Based on the techniques to learn the multiple levels of representations, deep learning is capable of exploiting complex and nonlinear features from big data. It has been applied with success in video classification [4], natural language processing [5], object detection [6], and many other domains. In the transportation research area, deep learning is increasingly presented in traffic state forecasting and achieves attractive performances [7], [8]. However, to the best knowledge of the authors, the following questions have not been well addressed: (1) Given that multitask learning (MTL) is gaining remarkable performance in other domains [9], [10], how can traffic state forecasting stand to benefit from MTL? (2) Is it feasible to incorporate deep learning and statistical tools to achieve a better performance than that of developing architecture engineering for a deep learning model alone? (3) Given that deep learning is a data-driven approach, how much impact will the training data size have on traffic forecasting?

To address the abovementioned problems, this paper proposes a deep learning based multitask learning framework with Gated Recurrent Units [11] (MTL-GRU) to forecast traffic flow and traffic speed simultaneously. The rest of this paper is organized as follows. Section II reviews the current studies. Section III introduces the MTL-GRU model and feature engineering. Numerical experiments are conducted in Section IV. Finally, Section V draws some interesting conclusions.
II. LITERATURE REVIEW

Over the past few decades, many traffic state forecasting models have been advanced to facilitate traffic management. The existing methods can be classified into three categories: parametric approaches, non-parametric approaches, and hybrid approaches.

Extended from auto-regressive moving average (ARMA), auto-regressive integrated moving average (ARIMA) was first introduced to forecast traffic states by Ahmed and Cook [12]. Over the years, the ARIMA model had been used as the basis for several variants (e.g., seasonal ARIMA (SARIMA) and Kohonen-ARIMA (KARIMA)). SARIMA was capable of extracting the seasonality of variable time series processes and had been successfully applied in traffic flow prediction [13]. KARIMA model adopted the first level to perform clustering between time series samples and leveraged a Kohonen map to aggregate and update clusters [14]. Parametric approaches are suitable to deal with regular variations, but the performance is undesirable when the traffic data show significant stochastic and nonlinear characteristics.

To address this problem, non-parametric regression has shown remarkable advantages in various traffic state forecasting practices. Zheng et al. proposed three classic non-parametric models to improve traffic prediction performance with reduced data dimensionality [15]. A Bayesian network approach was presented for traffic flow prediction [16]. An online learning weighted support vector regression (SVR) was proposed to forecast short-term traffic flow [17]. ANN models have also been developed for traffic forecasting in [18], [19]. Being capable to handle stochastic and nonlinear features, non-parametric approaches show a promising alternative to achieve better performance. However, using shallow structures, traditional non-parametric approaches are insufficient to model complex relationships, consequently, fail to further improve the forecasting accuracy.

To maximize the strengths and whilst minimize the defects of different types of approaches, hybrid methods are explored to achieve attractive results. Cetin and Comert combined the ARIMA model with expectation-maximization and cumulative sum algorithms to carry out short-term traffic flow prediction [20]. An adaptive hybrid fuzzy rule-based system approach was proposed for modeling and forecasting urban traffic flow [21]. Concerning forecasting accuracy as well as computational efficiency, Lippi et al. devised two new support vector regression models by combining the SARIMA model with a Kalman filter [22]. Hybrid approaches deliver plausible results in various practices. But based on conventional theories, the performance is still limited in many scenarios. As Lippi et al. state when enlarging the training data size, the performance quickly reaches its bottleneck, failing to take advantage of larger datasets [22].

In recent years, deep learning has obtained increasing attention in traffic state forecasting to tackle the puzzles abovementioned. Lv et al. proposed a novel deep learning based traffic flow prediction architecture with big data, in which a stacked autoencoder model was used to extract traffic flow features [7]. Polson et al. combined a linear model and a sequence of tanh layers to develop a deep learning model to forecast traffic flow [23]. Wu et al. proposed a deep learning based traffic flow prediction model with convolutional neural networks (CNN) layer to learn spatial features and GRU to capture temporal features [24]. Zhang et al. introduced Generative Adversarial Networks (GAN) to estimate trip travel times the consideration of network-wide spatiotemporal correlations [25]. Kim and Jeong proposed Deep Q-Network (DQN) to predict traffic flow in multiple intersections [26]. Those studies show a promising performance to predict traffic states. However, most existing models are single task learning (STL), which cannot take the advantages of information sharing among related tasks.

Fortunately, numerous attempts were made to forecast traffic states by introducing multitask learning. Jin et al. presented an MTL backpropagation network to carry out traffic flow modeling and forecasting [27]. Deng et al. proposed an MTL framework to identify traffic stations with clustering algorithms and utilize a fast iterative shrinkage-thresholding algorithm (FISTA) to carry out the short and long term traffic speed predictions [28]. Huang et al. introduced a multitask regression to predict traffic flow with a deep belief network (DBN) consisted of several neural network layers [29]. Zhang et al. proposed a deep learning based MTL model with limited neural network layers to predict network-wide traffic speed [30]. However, these models are based on either conventional methods [27], [28] or deep learning structures with only several stacked neural network layers [29], [30], which may restrict their prediction capacities.

In summary, to meet the increasing needs of accurate traffic information in ITS, a wide variety of algorithms have been explored and interdisciplinary skills have been involved. Although these approaches have significantly changed our perception of traffic operations and management, the questions mentioned in Section I are still without convincing answers. To cast light on those problems, this paper probes in the following strategies: (1) With residual mappings [31], the deep learning based MTL-GRU model is proposed to forecast traffic flow and traffic speed simultaneously with a deeper network structure; (2) With the help of statistical tools, this paper conducts feature engineering to extract the most informative features for the proposed MTL-GRU model; (3) This study investigates the impact of the size of training data on model performance.

III. METHODOLOGY

This section first describes MTL and GRU, and then builds an MTL-GRU model with residual mappings to forecast traffic flow and traffic speed simultaneously. Finally, Spearman’s rank correlation coefficient is introduced to select the most informative features with the help of statistical tools. The flowchart of the proposed approach is shown in Figure 1.
A. MULTITASK LEARNING

Approaches for traffic forecasting are usually STL, which fails to take advantage of the information shared by related tasks. Fortunately, MTL is capable to enhance the forecasting performance by learning several tasks at the same time [32]. Multitask approaches are widely adopted in the areas of face detection, natural language processing, semantic classification and information retrieval, and speech synthesis. In the transportation research area, traffic state forecasting tasks are generally approached as single and independent problems. In fact, given that the strong correlation between multiple traffic variables, traffic forecasting is not a stand-alone problem, but it can be influenced by heterogeneous and underlying correlated factors. Given that the rich set of related tasks, more accurate forecasting could be achieved by optimizing different tasks of traffic state forecasting jointly.

MTL can improve learning ability for one task by utilizing the relatedness contained in other tasks. These tasks are learned in parallel while using a shared representation. In this way, the information from one task can help the related tasks to be learned more effectively. Formally, the MTL model is designed to address a sequential forecasting problem: $\hat{Y}(X) = F(X_N)$. Let $X_N = [x_m^n | m = 1, 2, \ldots, M; n = 1, 2, \ldots, N]$ denotes the high dimensional observations for the $M$ tasks, where the $m^{th}$ task has $N$ sequential observations.

The general structures of deep learning based STL and MTL models are illustrated in Figure 2. In an STL approach, the forecasting of traffic flow or traffic speed is considered as a single task, in which the forecasting model must be built and trained separately for each task. In an MTL architecture, the forecasting of traffic flow and speed are considered as related tasks. In the MTL model, deep representations will be extracted from the neural networks by fine-tuning the related tasks jointly and each task is expected to obtain a better result.

The loss functions of an STL model and an MTL model are defined in Equation (1) and (2), respectively.

$$J(W, b) = \frac{1}{2 \cdot N} \sum_{i=1}^{N} (h_{W,b}(x^{(i)}) - y^{(i)})^2 \quad (1)$$

$$J(W, b) = \frac{1}{2 \cdot N} \sum_{m=1}^{M} \left( W_m \sum_{i=1}^{N} (h_{W,b}(x^{(i)}) - y^{(i)})^2 \right) \quad (2)$$

where $M$ and $N$ are the number of tasks and the size of training data, respectively. $W$ and $b$ are weights and bias, respectively. $h_{W,b}(x)$ denotes hypothesis.

B. GATED RECURRENT UNITS

As a class of deep learning neural networks, the recurrent neural network (RNN) is specialized for processing sequential data by introducing memory to retain information in each timestep. This procedure makes the RNN extremely deep, which results in difficulties to train the model due to exploding and vanishing gradient problems [33]. To address the difficulties of training RNN, sophisticated recurrent units, such as long short-term memory (LSTM) [34] and GRU are proposed.

![FIGURE 3. Structure of GRU [11].](image)

In this paper, GRU is employed to forecast short-term traffic flow and speed. A typical structure of the GRU is demonstrated in Figure 3, in which GRU has two gates, a reset gate $r$ decided whether the previous hidden state is ignored, and an update gate $z$ selected whether the hidden state is to be updated with a hidden state $\tilde{h}$. The computational procedure is defined as follows:

$$r_t = \sigma(W_{xr} x_t + W_{hr} h_{t-1} + b_r) \quad (3)$$

$$z_t = \sigma(W_{xz} x_t + W_{hz} h_{t-1} + b_z) \quad (4)$$

$$\tilde{h}_t = \tanh(W_{xh} x_t + W_{hh}(r_t \odot h_{t-1}) + b_h) \quad (5)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t \quad (6)$$

where $W_s$ and $b_s$ are weights bias, respectively. $\odot$ denotes the element-wise vector product, $\sigma$ is the sigmoid function and $x_t$ is the input of this layer at time $t$. The output of this layer is the hidden state at each timestep.

C. MULTITASK LEARNING GATED RECURRENT UNITS

The structure of the proposed MTL-GRU model is shown in Figure 4. The learning process of this model contains two levels divided by the Merge Layer. In the first level, the traffic
flow and speed forecasting task are trained separately to extract the shallow representations of each task individually. In the second level, the learning results of Level 1 are merged and then fed into the GRU layers of Level 2 to learn the deeper representations hidden between the two tasks. Moreover, the residual mapping indicated by the black curve in Figure 4 creates a shortcut connection between these two levels. With the residual mapping, the network of the MTL model can be very deep with the improved performance [31].

D. FEATURE ENGINEERING WITH STATISTICAL INTERPRETATION

The correlations among different variables play a key role to successfully implement the MTL-GRU model. This paper introduces Spearman’s rank correlation coefficient to calculate nonlinear correlation between different traffic variables. The rank of the variables is then correlated using the Pearson correlation coefficient.

For variables $X$ and $Y$ with a size $n$, the $n$ raw scores $X_i, Y_i$ are converted to ranks $rgX_i, rgY_i$, and $r_i$ is calculated as

$$ r_i = \rho_{rgX, rgY} = \frac{cov(rgX, rgY)}{\sigma_{rgX} \sigma_{rgY}} \quad (7) $$

where $\rho$ denotes the usual Pearson correlation coefficient but applied to the rank variables, $cov(rgX, rgY)$ is the covariance of the rank variables, $\sigma_{rgX}$ and $\sigma_{rgY}$ are the standard deviations of the rank variables.

Furthermore, to meet the demands of accurate forecasting and explainable phenomena, the autocorrelation function is utilized to investigate the seasonality of traffic data. In addition, the Box Plot is employed to dig into the interaction among the sampling timestamps of each day and different types of days.

IV. EXPERIMENTS

This section carries out extensive experiments with real-world datasets to validate the performance of the proposed approaches.

A. DATA DESCRIPTION

The traffic datasets including traffic flow and speed are collected from the Caltrans Performance Measurement System (PeMS). The datasets are measured every 30s from over 15000 individual detectors deployed statewide in the freeway systems across California. In this paper, the collected data are aggregated by 15-min intervals from six stations (Table 1) in nine months (January-September 2009) and 15-min is taken as the prediction horizon. Before being fed into the MTL-GRU model, the traffic data is normalized into the range of $[-1, 1]$.

| Station | Highway | Lanes |
|---------|---------|-------|
| 715916  | 5 S     | 4     |
| 716312  | 91 E    | 6     |
| 716841  | 605 N   | 4     |
| 716933  | 5 N     | 5     |
| 717087  | 10 W    | 4     |
| 717123  | 10 E    | 4     |

B. FEATURE ENGINEERING

This paper employs statistical tools to expose relationships among traffic data and transform those relationships into features. Take station 716312 as an example, the latent features hidden in traffic data will be investigated and extracted step by step.

1) CORRELATION COEFFICIENT

As a multitask approach, two tasks (traffic flow and speed) are learned simultaneously. The correlation among tasks plays a fundamental role to deploy MTL effectively. For this purpose, the correlation coefficients of different variables are accessed by Spearman’s rank correlation coefficient, which is capable to test the nonlinear relationship between two variables. In Figure 5, it can be observed that the correlation coefficient indicates a significant nonlinear correlation between the traffic flow variable and the speed variable. This implies one variable could be an informative feature when forecasting another variable.

2) “TIMESTAMP * DAY” FEATURE

Given that the traffic datasets are sampled every 15 minutes, there are 96 data points daily with 96 sampling timestamps, which implies 96 maybe a seasonality. However, for a longer time horizon, other seasonal periods maybe remain unknown. Hence, autocorrelation is introduced to expose veiled significant seasonality. Figure 6 shows the autocorrelation results of both traffic flow and speed data. For both figures, each lag corresponds to a data point or 15-min timestep. The height of the red or green lines represents the correlation between traffic data at $t$ and $t$-time lag. As can be observed in Figure 6, there are two peak points at lag 96 and lag 672, suggesting double seasonality is involved in traffic flow and speed data: daily (96) and weekly (7*96). To convert the
double seasonality to features, each data point regarding a unique timestamp of each day is labeled from 0 to 95, and different day of each week is labeled as 0, 1, 2, ... 6 from Monday to Sunday.

As an essential component of data analysis, intuitive data visualization enables human interpretation and judgment to gain insight among various data. In this section, a nonnegligible correlation between “Timestamp” and “Day” is exposed via data visualization. After grouping traffic data by different types of days from January to September 2009, Figure 7 displays distinct differences from Monday to Sunday in traffic flow and traffic speed data, respectively.

3) FORECASTING FUNCTION OF MTL-GRU
Given that a significant nonlinear correlation exists between traffic flow and traffic speed, the MTL-GRU model is applied to learn the two tasks. In general, as for the task $m$ in an MTL model, the previous observations $x_{m}^{t-1}, x_{m}^{t-2}, \ldots, x_{m}^{t-k}$ ($k$ stands for the past time interval) are utilized to predict the traffic state at timestamp $t$. Therefore, in the MTL-GRU model, the previous traffic flow data points $x_{f}^{t-1}, x_{f}^{t-2}, \ldots, x_{f}^{t-k}$ and traffic speed data points $x_{s}^{t-1}, x_{s}^{t-2}, \ldots, x_{s}^{t-k}$ will be leveraged to forecast $x_{f}^{t}, x_{s}^{t}$. Equation (8) is the forecasting function of the MTL-GRU for traffic flow and speed.

$$\hat{Y}(X) = [x_{f}^{t}, x_{s}^{t}] = F \begin{bmatrix} x_{f}^{t-k} & x_{s}^{t-k} \\ \vdots & \vdots \\ x_{f}^{t-2} & x_{s}^{t-2} \\ x_{f}^{t-1} & x_{s}^{t-1} \end{bmatrix}$$ (8)

To introduce the Timestamp * Day feature in Equation (8), Timestamp labels (0-95) and Day labels (0-6) have been combined and transformed into a categorical through one hot encoding (shown in Table 2). Therefore, three types of features are fed into the MTL-GRU model as input: traffic flow, traffic speed, and Timestamp * Day. Then, the forecasting function of the MTL-GRU is illuminated in Equation (9).

$$\hat{Y}(X) = [x_{f}^{t}, x_{s}^{t}] = F \begin{bmatrix} x_{f}^{t-k} & x_{s}^{t-k} & [1 0 \ldots, 0 0 0]_{k} \\ x_{f}^{t-2} & x_{s}^{t-2} & [0 1 \ldots, 1 0 0]_{2} \\ x_{f}^{t-1} & x_{s}^{t-1} & [0 1 \ldots, 0 0 1]_{1} \end{bmatrix}$$ (9)

C. MODEL COMPARISONS
In order to validate the outperformance of the proposed MTL-GRU, various STL methods are presented and compared. In the experiments, traffic data from six months...
TABLE 3. Traffic flow forecasting. MAPE (%) is reported.

| Station | SARIMA$_{Kal}$ | $k$-NN | $v$-SVM | XGBoost | EFNN | LSTM | Conv-GRU | GRU | TCN | MTL-GRU$_{orig}$ | MTL-GRU |
|---------|----------------|--------|--------|---------|------|------|----------|-----|-----|-----------------|---------|
| 715916  | 5.71           | 6.37   | 6.31   | 6.30    | 6.33 | 5.68 | 5.70     | 5.64| 5.59| 5.2 | 5.42          |
| 716312  | 4.89           | 6.89   | 5.99   | 5.89    | 5.75 | 4.79 | 4.67     | 4.80| 4.70| 4.61          | 4.53    |
| 716841  | 4.99           | 5.50   | 5.85   | 5.80    | 5.76 | 4.89 | 4.81     | 4.80| 4.75| 4.74          | 4.70    |
| 716933  | 5.02           | 6.32   | 5.69   | 5.70    | 5.89 | 5.03 | 4.95     | 4.89| 4.90| 4.83          | 4.71    |
| 717087  | 4.80           | 6.23   | 5.51   | 5.54    | 5.48 | 4.76 | 4.73     | 4.75| 4.71| 4.66          | 4.59    |
| 717123  | 4.50           | 5.76   | 5.25   | 5.37    | 5.30 | 4.42 | 4.45     | 4.40| 4.41| 4.39          | 4.31    |

TABLE 4. Traffic speed forecasting. MAPE (%) is reported.

| Station | SARIMA$_{Kal}$ | $k$-NN | $v$-SVM | XGBoost | EFNN | LSTM | Conv-GRU | GRU | TCN | MTL-GRU$_{orig}$ | MTL-GRU |
|---------|----------------|--------|--------|---------|------|------|----------|-----|-----|-----------------|---------|
| 715916  | 7.93           | 8.23   | 8.20   | 8.17    | 7.90 | 7.92 | 7.93     | 7.90| 7.86| 7.80          | 7.75    |
| 716312  | 3.89           | 4.37   | 4.30   | 4.35    | 3.99 | 3.77 | 3.75     | 3.75| 3.80| 3.72          | 3.66    |
| 716841  | 3.12           | 3.98   | 3.95   | 3.24    | 3.41 | 3.10 | 3.02     | 2.98| 2.97| 2.83          | 2.43    |
| 716933  | 7.67           | 8.24   | 8.30   | 8.31    | 8.34 | 7.59 | 7.49     | 7.45| 7.40| 7.37          | 7.30    |
| 717087  | 4.18           | 5.39   | 5.25   | 5.30    | 5.24 | 4.21 | 4.13     | 4.15| 4.16| 4.11          | 4.02    |
| 717123  | 4.89           | 5.82   | 5.79   | 5.76    | 5.59 | 5.00 | 4.90     | 4.92| 4.90| 4.82          | 4.66    |

(January-July 2009) are employed as the training data and three months (July-September 2009) as the testing period. In STL scenarios, nine types of counterpart models are described as follows.

1) SARIMA$_{Kal}$: SARIMA$_{Kal}$ is a SARIMA model with the Kalman filter, which is utilized in Ref. [22] to predict short-term traffic flow. In this study, two SARIMA$_{Kal}$ models are introduced to carry out traffic flow prediction and traffic speed prediction with its historical data as input.

2) $k$-NN: $k$-Nearest Neighbors ($k$-NN) is a nonparametric method used for classification and regression, which is proposed in Ref. [35] to predict traffic speed. In this study, the inputs of SARIMA$_{Kal}$ models are fed into the $k$-NN models.

3) $v$-SVM: SVM is usually applied to classification, regression, signal processing, etc. Inspired by Dong et al. [36], $v$-SVM is employed here. Two $v$-SVM models are developed to predict traffic flow and traffic speed, respectively. A set of past values of variables (i.e., traffic flow or traffic speed) are selected and fed into the corresponding $v$-SVM model.

4) XGBoost: XGBoost is a scalable machine learning system for tree boosting. It is widely used in many machine learning challenges with excellent performance. The inputs of $v$-SVM models are fed into the corresponding XGBoost model for training and forecasting.

5) EFNN: An EFNN is a neural network that realizes a set of fuzzy rules, and it can evolve its structure and functionality in an adaptive, life-long, modular way. In this paper, the inputs of XGBoost models are fed into the EFNN models to predict traffic flow and traffic speed, individually.

6) LSTM: With past traffic flow data or traffic speed data as input, the LSTM model carries out the prediction of traffic flow or traffic speed without joint training the two tasks.

7) Conv-LSTM: A Conv-LSTM layer is similar to an LSTM layer, but the input transformations and recurrent transformations are both convolutional. The Conv-LSTM model adopts the same input as the LSTM model to forecast traffic flow or traffic speed.

8) GRU: It also utilizes GRU as the kernel neural unit and has a structure of single input-single output, which will be built for two times to forecast traffic flow and traffic speed individually.

9) TCN: Based on CNN, Bai et al. proposed a novel Temporal Convolutional Network (TCN) architecture to handle time series modeling with the help of dilated causal convolutions [37]. Here, two TCN models forecast the traffic flow and traffic speed, respectively.

The abovementioned models (i.e., SARIMA$_{Kal}$, $k$-NN, $v$-SVM, XGBoost, EFNN, LSTM, Conv-GRU, GRU, and TCN) are trained in an STL way. Moreover, to verify the efficiency of residual mappings, an original MTL-GRU (MTL-GRU$_{orig}$) model without residual mappings is introduced to forecast traffic flow and traffic speed simultaneously. The mean absolute percentage error (MAPE) is used to evaluate the performance of the proposed method.

$$MAPE(Y, \hat{Y}) = \frac{1}{N} \sum_{i=1}^{N} \frac{|Y_i - \hat{Y}_i|}{Y_i}$$  \hspace{1cm} (10)

where $Y_i$ and $\hat{Y}_i$ are the $i^{th}$ ground truth and forecasted values, respectively. To avoid the problem that few examples with low traffic flow might affect error measurement, this paper measures MAPE$_{100}$ as in Ref. [23]. In MAPE$_{100}$, the MAPE is calculated where $Y_i > 100$ vehicles/hour.

The forecasting results of traffic flow and traffic speed are detailed in Table 3 and 4, respectively. It can be found that the MTL-GRU model achieves the best performance...
when forecasting traffic flow and traffic speed for six stations. Specifically, as for the traffic flow forecasting, the MAPE is 5.42%, 4.53%, 4.70%, 4.71%, 4.59%, and 4.31%. As for the traffic speed forecasting, the MAPE is 7.75%, 3.66%, 2.43%, 7.30%, 4.02%, and 4.66%. Moreover, in the STL scenarios, compared with the conventional methods (i.e., SARIMA_Kal, k-NN, v-SVM, XGBoost, and EFNN), the deep learning based methods (i.e., LSTM, Conv-GRU, GRU, and TCN) achieve better performance. It is also interesting to notice that the MTL-GRU_{Orig} model performs worse than the MTL-GRU model with residual mappings. In summary, the MTL-GRU model can benefit from learning relevant tasks simultaneously and residual mappings are capable of improving the performance of the multitask learning model.

D. SENSITIVITY ANALYSIS
This section carries out the sensitivity analyses of the MTL-GRU model on the “Timestamp * Day” feature and the size of training data.

To validate the significance of the Timestamp * Day feature, this section applies the MTL-GRU model on six stations (c.f., Table 1). With $k = 4$ as the look-back window, the proposed model is fixed with two GRU layers and two dropout layers. For the GRU layer, the number of units is 128. 0.2 is set as the dropout rate to the dropout layer. With a batch size 700, the training epoch of each model is set as 300. The mean MAPEs of traffic flow and traffic speed are drawn in Figure 8. It can be observed that the MTL-GRU model with the Timestamp * Day feature yields better performance, which indicates that the Timestamp * Day feature plays a positive role in improving the forecasting accuracy.

To explore the impact of enlarging training data size, this section aggregates training periods from one to twelve months (July 2008-July 2009) and employs the same three months (July-September 2009) as the testing period. The mean MAPEs of traffic flow and traffic speed on six stations are shown in Figure 9. This experiment suggests that the performance of deep learning based MTL-GRU approaches benefits from enlarging training data size and receive better performance as more training data available. The result in Figure 9 is different from Ref. [22], which claims the performance does not improve when including in the training data set for more than two months. In summary, compared with conventional methods, the deep learning based MTL-GRU can overpower the bottleneck caused by scaling up training datasets.

V. CONCLUSION
This paper proposes a deep learning based multitask model (i.e., MTL-GRU) for traffic flow and speed forecasting. Combined with feature engineering, the MTL-GRU model with residual mappings achieves the best results compared with other approaches. Then, traffic data from PeMS are used to perform numerical experiments. Meanwhile, both classic methods (i.e., SARIMA_{Kal}, k-NN, v-SVM, XGBoost, and EFNN) and state-of-the-art deep learning approaches (i.e., LSTM, Conv-GRU, GRU, TCN, and MTL-GRU_{Orig}) are introduced as the comparison counterparts. Based on the numerical results, some useful findings are concluded as follows: (1) This study proposes the MTL-GRU model with residual mappings to improve the forecasting accuracy of traffic flow and traffic speed by using multitask learning;
(2) To select the most informative features, this paper employs Spearman's rank correlation coefficient to calculate the nonlinear correlation between different traffic variables; 
(3) The deep learning based MTL-GRU model could overpower the bottleneck caused by scaling up training datasets. In summary, the MTL-GRU model with residual mappings and feature engineering is promising to forecast short-term traffic state.

REFERENCES

[1] E. J. Vlahogianni, M. G. Karlaftis, and J. C. Golas, “Short-term traffic forecasting: Where we are and where we’re going,” Transp. Res. C, Emerg. Technol., vol. 43, pp. 3–19, Jun. 2014.
[2] Z. He, G. Qi, L. Lu, and Y. Chen, “Network-wide identification of turn-level intersection congestion using only low-frequency probe vehicle data,” Transp. Res. C, Emerg. Technol., vol. 108, pp. 320–339, Nov. 2019.
[3] L. Zheng, H. Huang, C. Zhu, and K. Zhang, “A sensor-based K-nearest neighbors method for traffic speed prediction under data missing,” Transp. Performance B, Transp. Dyn., vol. 8, no. 1, pp. 182–199, Jan. 2020.
[4] Y.-C. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang, “Exploiting feature and class relationships in video categorization with regularized deep neural networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 2, pp. 352–364, Feb. 2018.
[5] T. Young, D. Harazika, S. Poria, and E. Cambria, “Recent trends in deep learning based natural language processing,” IEEE Comput. Intell. Mag., vol. 13, no. 3, pp. 55–75, Aug. 2018.
[6] J. Han, D. Zhang, G. Cheng, N. Liu, and D. Xu, “Advanced deep-learning techniques for salient and category-specific object detection: A survey,” IEEE Signal Process. Mag., vol. 35, no. 1, pp. 84–100, Jan. 2018.
[7] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, “Traffic flow prediction with big data: A deep learning approach,” IEEE Trans. Intell. Transp. Syst., vol. 16, no. 2, pp. 865–873, Sep. 2014.
[8] K. Zhang, Z. Liu, and L. Zheng, “Short-term prediction of passenger demand in multi-zone level: Temporal convolutional neural network with multi-task learning,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 4, pp. 1480–1490, Apr. 2020.
[9] R. Ranjan, V. M. Patel, and R. Chellappa, “HyperFace: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 1, pp. 121–135, Jan. 2019.
[10] M. Hessel, H. Soyer, L. Espeholt, W. Czarnecki, S. Schmitt, and H. Van Hasselt, “Multi-task deep reinforcement learning with PopArt,” in Proc. AAAI Conf. Artif. Intell., vol. 33, Jul. 2019, pp. 3796–3803.
[11] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” 2014, arXiv:1412.3555. [Online]. Available: http://arxiv.org/abs/1412.3555
[12] M. S. Ahmed and A. R. Cook, “Analysis of freeway traffic time series data by using Box-Jenkins techniques,” Transp. Res. Board, vol. 722, pp. 1–9, 1979.
[13] B. M. Williams and L. A. Hoel, “Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results,” J. Transp. Eng., vol. 129, no. 6, pp. 664–672, Nov. 2003.
[14] M. Van Der Voort, M. Dougherty, and S. Watson, “Combining Kohonen maps with Arima time series models to forecast traffic flow,” Transp. Res. Part C: Emerg. Technol., vol. 4, no. 5, pp. 307–318, Oct. 1996.
[15] L. Zheng, C. Zhu, N. Zhu, T. He, N. Dong, and H. Huang, “Feature selection-based approach for urban short-term travel speed prediction,” IET Intell. Transp. Syst., vol. 12, no. 6, pp. 474–484, Aug. 2018.
[16] Z. Zhu, B. Peng, C. Xiong, and L. Zhang, “Short-term traffic flow prediction with linear conditional Gaussian Bayesian network,” J. Adv. Transp., vol. 50, no. 6, pp. 1111–1123, Oct. 2016.
[17] 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.
[18] 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. 114–121, Feb. 2006.

KUNPENG ZHANG received the Ph.D. degree from the College of Mechanical and Vehicle Engineering, Hunan University, China, in 2019. He is currently working with the College of Electrical Engineering, Henan University of Technology, Zhengzhou, China. His research interests focus on intelligent transportation systems and urban traffic control and management.
LAN WU received the bachelor’s degree in engineering from Henan Normal University, Xinxiang, China, in 2003, and the master’s and Ph.D. degrees in microelectronics from the Xi’an University of Technology, Xi’an, China, in 2006 and 2009, respectively. She is currently a Professor, College of Electrical Engineering, Henan University of Technology, Zhengzhou, China. Her current research interests include information security, artificial intelligence, multisensor networked information fusion theory, intelligent transportation systems, and intelligent information processing. She is also a Committee Member of Intelligent Automation Committee of Chinese.

JIANG DENG received the M.S. degree from the College of Management and Economics, Tianjin University, China, in 2006. He is currently the Chief Executive Officer of Qingdao Fantaike Bearing Company, Ltd. His research interests include consumer finance, Internet finance, and financial information systems.

ZHAOJU ZHU received the Ph.D. degree in mechanical engineering from Shandong University, China, in 2019. He currently holds a Lecturer position with Fuzhou University and a postdoctoral position with the Institute of Automation, Chinese Academy of Sciences. His research interests include advanced manufacturing and data analysis.