Proceedings of the ASME 2021 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2021
August 17-20, 2021, Online, Virtual

IDETC2021-69058

SHORT-TERM VEHICLE VELOCITY FORECASTING VIA CYCLE SEGMENTATION MODEL SELECTION

Yuanzhi Liu*
The University of Texas at Dallas
Richardson, TX 75080
Email: yuanzhi.liu@utdallas.edu

Jie Zhang†
The University of Texas at Dallas
Richardson, TX 75080
Email: jiezhang@utdallas.edu

ABSTRACT

Vehicle velocity forecasting plays a critical role in scheduling the operations of varying systems and devices in a passenger vehicle. This paper first generates a repeated urban driving cycle dataset at a fixed route in the Dallas area, aiming to contribute to the improvement of vehicle energy efficiency for commuting routes. The generated driving cycles are divided into cycle segments based on intersection/stop identification, deceleration and reacceleration identification, and waiting time estimation, which could be used for better evaluating the effectiveness of model localization. Then, a segment-based vehicle velocity forecasting model is developed, where a machine learning model is trained/developed at each segment, using the hidden Markov chain (HMM) model and long short-term memory network (LSTM). To further improve the forecasting accuracy, a localized model selection framework is developed, which can dynamically choose a forecasting model (i.e., HMM or LSTM) for each segment. Results show that (i) the segment-based forecast could improve the forecasting accuracy by up to 24%, compared the whole cycle-based forecast; and (ii) the localized model selection framework could further improve the forecasting accuracy by 6.8%, compared to the segment-based LSTM model. Moreover, the potential of leveraging the stopping location at an intersection to estimate the waiting time is also evaluated in this study.

Keywords: Repeated driving cycle dataset, cycle segment, vehicle velocity forecasting, model selection, intersection waiting time, deep learning

1 Introduction

Short-term traffic forecasting has been extensively investigated in the past decade as a potential feasible solution to mitigate the growing concern of traffic congestion, especially with the advent of big data, artificial intelligence, and internet of things [1, 2]. It has been shown that the traffic can be significantly improved by integrating the existing road network with smart traffic light control and intelligent transportation system, based on the real-time traffic measuring and forecasting, such as the traffic flow volume, traffic density, average traffic velocity, and travel time [3].

1.1 Network Traffic Forecasting

The majority of these studies usually focus on a broader scope traffic forecasting from point-level, street-level, or network-level, serving to provide further insights for transportation management and policy making. At the very-beginning stage, several basic statistic and machine learning-based algorithms were exploited for varying traffic conditions. For instance, the auto-regressive inte-

*Ph.D. Student, Department of Mechanical Engineering, ASME Student Member.

†Assistant Professor, Department of Mechanical Engineering, ASME Professional Member. Address all correspondence to this author.

grated moving average (ARIMA) method has been broadly adopted for point-level or street-level traffic forecasting [4]. ARIMA was adopted as a benchmark to be compared with other advanced approaches, such as support vector machine (SVM) [5], back propagation neural network (BPNN) [6], radial basis function neural network (RBFNN), Kalman filter, hidden Markov train (HMM) [7], and long short-term memory network (LSTM) [8, 9]. Historical driving records are usually considered as the training dataset for forecasting. Though only the temporal relationship is taken into account, the aforementioned algorithms are capable of predicting the traffic situations in most cases with a reasonable accuracy. It is also worth noting that there is no published report that indicates the superiority of any of these algorithms, due to diverse traffic data sources.

However, some of these forecasting algorithms may not perform well for a large area with complicated transportation networks and uncertain environmental factors. To alleviate this arising challenge, advanced deep neural networks with topological feature embedding have been employed to characterize the spatial-temporal characteristics in forecasting. Massive efforts have been performed on convolutional neural network (CNN)-based algorithms [10], which generally falls into two categories: convolution-based LSTM that integrates CNN and LSTM [11, 12], and temporal graph convolutional networks that combine graph neural network with gated recurrent units/networks [13, 14, 15]. By comparing with the reported forecasting performance, these deep learning-based methods have shown overwhelming superiority over the aforementioned classic parametric or simple-structured machine learning-based approaches. This newly-emerging trend is very likely to continue in traffic forecasting [16].

1.2 Individual Vehicle Velocity Forecasting

It should be noted that for the aforementioned studies, all the data has been collected from a single or a series of fixed observation locations using sensors like inductive-loop detector, wireless magnetometer, microwave radar, and video image processor. These studies emphasize more on the networked vehicles rather than an individual passenger vehicle. However, velocity forecasting for individual vehicles has drawn significant attention in the past decade, especially along with the fast-growing demand of vehicle electrification. Velocity forecasting plays a critical role in improving the energy efficiency for electric or hybrid vehicles. Velocity forecasting serves as the system input for a model predictive control-assisted or reinforced learning-based energy management system to optimize the charging/discharging schedule, the regenerative power harvest, and the operation of an on-board air-conditioning system [17, 18, 19, 20, 21, 22, 23, 24].

There are generally two major discrepancies that the individual vehicle velocity forecasting differs from the network traffic forecasting. First, individual vehicle velocity forecasting utilizes the floating velocity trajectory as the data source instead of the network traffic records. Second, individual vehicle velocity forecasting requires a significantly shorter prediction horizon at seconds, compared to the network traffic forecasting at minutes or hours timescales. The traffic forecasting algorithms discussed above are still applicable and practicable to individual vehicle velocity forecasting, which can generally be categorized into stochastic and deterministic approaches. For instance, as one of the most popularly used stochastic methods, HMM was modified by Jing et al. [25] with a fuzzy logistic model to predict individual vehicle speed 8 seconds ahead, and Zhou et al. [26] developed a self-learning multi-step Markov chain model based on simulated data. Sun et al. [27] showed that deterministic algorithms like RBFNN and BPNN performed better than HMM, and Liu et al. [28] also reported similar results that both LSTM and ARIMA outperformed HMM in 10 seconds ahead speed forecasting based on a real urban driving dataset. Moreover, the feasibility of embedding LSTM on board has been validated by Gaikwad et al. [29] with an on-board processor. Besides, Lemieux et al. [30] have investigated the deep belief network and the stacked auto-encoder for speed forecasting based on a highway speed dataset.

It is noteworthy that the individual vehicle velocity forecasting performance could be improved by considering the surrounding traffic situations when integrating network traffic forecasting. Moreover, Moser et al. [31] and Zhang et al. [32] showed that individual vehicle velocity forecasting could be improved by knowing the states of traffic lights in advance and leveraging vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. However, one of the challenges is that the policy-making and construction of intelligent transportation system is falling far behind the electrification and intellectualization of passenger vehicles. It is still economically prohibitive to enable all passenger vehicles being connected to the transportation control system and receive real-time traffic information. Based on the existing urban transportation infrastructure and vehicle installations, how to improve the velocity forecasting accuracy is still a stringent and challenging problem for vehicle energy management.

1.3 Research Objective

To further improve the performance of individual vehicle velocity forecasting , a hybrid velocity forecasting algorithm is developed in this study, by leveraging the fact that most of the newly-released electric or hybrid vehicles are equipped with on-board GPS devices. To validate the

proposed algorithm, a repeated urban driving cycle dataset is first generated by a same driver (with the same driving habits) in the Dallas area. The driving patterns between weekdays and weekends are investigated, and road segments are also identified. A forecasting pool that consists of HMM and LSTM is established, and a localized model selection framework is developed to dynamically choose a forecasting model at each road segment. This study seeks to enhance the forecasting accuracy with the currently available urban transportation infrastructure, and our contributions are threefold: i) generate a publicly available dataset with repeated driving cycles; ii) develop a segment-based vehicle speed forecasting model, in conjunction with a localized model selection framework; iii) explore the feasibility of leveraging the stopping position at an intersection for estimating the waiting time and thus improving the forecasting accuracy.

The remainder of the paper is organized as follows. First, the driving cycle dataset, pattern recognition, and road segments are analyzed in Section 2. The developed segment-based forecasting algorithm with model selection is applied to the dataset and compared with benchmarks in Section 3. Concluding remarks and future work are discussed Section 4.

2 Data Collection and Analyses

2.1 Data Collection

There exist several repeated driving cycles based on a fixed route published in the literature, such as the Connected Ann Arbor (A2) dataset [28], the Gothenburg driving dataset [33], the Seoul expressway testing dataset [34], and the Fort Collins repeated driving cycle dataset [29]; however, few datasets are publicly available. Another piratical way to obtain repeated driving cycles is to extract the repeated routes from large-scale traffic or vehicle energy consumption dataset that covers a certain period of time, such as the Geolife Trajectories[1], the vehicle energy dataset (VED) [35][2], the performance measurement system (PeMS) dataset[3], the Roma taxi dataset[4], and the San Francisco bay area taxi dataset[5].

To better understand the impacts of vehicle speed forecasting on energy management, we have generated a Dallas repeated driving cycle (DRD) dataset[6]. In this dataset, dozens of driving cycle tests have been performed on a fixed route in the Dallas area to simulate a typical commuting route for passenger vehicles, which consists of an expressway test of 5 kilometers and a local urban road test of 20 kilometers, as shown in Fig. 1. The dataset was acquired between December 2020 to January 2021 at around 4:45 PM to 6:00 PM, in which each cycle takes approximately 30 minutes using a conventional internal combustion engine passenger vehicle. All the testing cycles were conducted with a fixed route by the same driver with similar driving manners. It needs to be noted that some of the traffic signals in this area are adaptively controlled in a daily or continuous real-time manner, according to the progress report of a regional traffic retiming program [36]. More repeated commuting cycles by different drivers are expected to be included in the DRD dataset in the coming future.

Several critical dynamic driving parameters and indices are recorded using a GPS logger (brand: Garmin, model: eTrex 10) and a camera. All the data are time-stamped with a time interval of 1 second for the sake of accuracy and consistency, including the vehicle velocity, altitude and longitude information, elevation, heading direction, and traffic light picture/video. Compared with other datasets, a remarkable merit of this DRD dataset is that it covers a broader road types with tens of intersections, and traffic light images can be used as a potential tool for intersection waiting time prediction.

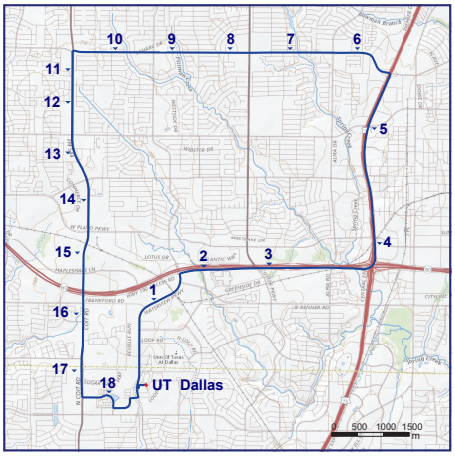


FIGURE 1: The testing route of DRD repeated driving cycles, which is close to the University of Texas at Dallas

2.2 Data Processing

As a preprocessing step, clustering plays an important role in improving the forecasting accuracy. For velocity forecasting using historical data, there are two general approaches to cluster the data and identify major spatial-

[1] https://www.microsoft.com/en-us/download/details.aspx?id=52367
[2] https://github.com/gsoh/VED/blob/master/README.md
[3] https://archive.ics.uci.edu/ml/datasets/PEMS-SF
[4] https://crawdad.org/roma/taxi/20140717/
[5] https://crawdad.org/epfl/mobility/20090224/cab/
[6] Available at: https://github.com/UTD-DOES/Dallas-Repeated-Driving-Cycle-Dataset

temporal patterns: i) cluster the cycles for pattern recognition, split the grouped cycles into segments, perform another round of clustering for the segments, and then establish the forecasting models; ii) skip the first-round clustering, and follow the rest steps [24, 37]. Note that unsupervised clustering can be transformed into supervised classification by manually labeling the data with expert knowledge, i.e., the cycles can be classified directly into weekday and weekend/holiday conditions based on daily driving experiences.

To quantify the traffic discrepancies between weekdays and weekends, a congestion index ε_{it} (of a location i at time t) is modified here by comparing the measured floating vehicle velocity with the free-flow velocity [38], as defined in Eq. 1. The parameter S_{it} represents the free-flow velocity defined by the maximum value recorded, while V_{it} represents the current vehicle velocity. A larger index ε_{it} indicates a relatively more severe traffic congestion. Here, a total of 18 locations away from the intersections are selected randomly using the stratified sampling algorithm, as shown in Fig. 1. The discrepancies between weekdays and weekends/holidays are illustrated in Fig. 2. It is seen that there do not exist many significant differences between weekdays and weekends/holidays, except the saturated section ranging from location 2 to 5 on the upper-right corner, which illustrates an improvement of the highway traffic on weekends and holidays. The possible reason behind this phenomenon is the current remote-work environment due to the COVID-19 pandemic. It is also noticed that the whole trip takes an averaged time of 1,801 seconds on weekends and holidays, which is 193 seconds shorter than that on weekdays. By investigating the details of route segments, it is observed that moving through the intersections in weekdays requires extra time, as shown in Fig. 3.

$$\varepsilon_{it} = \frac{s_{it}}{v_{it}} - 1 \tag{1}$$

2.3 Intersection/stop Identification

Given the preceding analysis, the second clustering approach discussed in Section 2.2 (i.e., split the grouped cycles into segments, perform another round of clustering for the segments, and then establish the forecasting models) is employed here for feature identification in the study. The driving cycles are directly divided into segments, followed by a time sequence clustering of the segments if necessary. Specifically, all the intersections are extracted from the route as separated segments because of their significant impacts on the whole driving time. A location is identified as an intersection or a T-junction with stop (yield) sign if it is detected with a complete stop or low velocity (i.e., 10

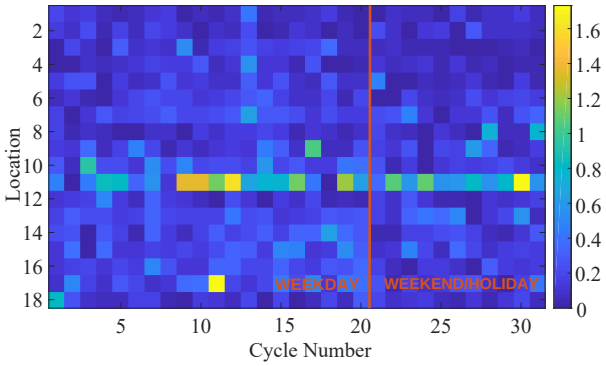


FIGURE 2: A comparison of the congestion index between weekdays and weekends/holidays

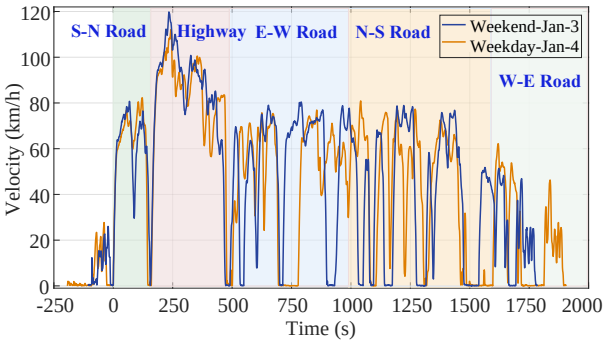


FIGURE 3: The velocity trajectories of typical driving cycles in weekdays and weekends/holidays (The location annotations are based on the weekday cycle.)

km/h) more than twice. Once the intersections are located, the routes in between will also be defined as independent cycle segments. It is worth mentioning that there are two major reasons why traffic signal identification is necessary rather than using the labeled data from public map sources directly: i) the vehicles have a high probability of moving through some labelled intersections or T-junctions without any interruption; ii) the vehicles need to wait two cycles at some traffic lights due to the heavy traffic conditions, resulting in another indirect hidden stop.

As can be seen from Fig. 4, the final location of a vehicle is scattered at an intersection, depending on the traffic volume and its arrival time. The furthest point downstream among all is treated as the location of the intersection. The primary step here is to organize the stop points that belong to the same intersection or potential stop into a group. General machine learning-based clustering algorithms like K-means and hierarchical clustering have been tested however with unsatisfied performance, since it is challenging to determine the appropriate number of clusters. In this study, we attempt to cluster the stop points using the maximum

distance method, in which only connections with a distance smaller than the maximum threshold will be considered to form a same group, i.e., 40 m for this DRD dataset. Given the specific velocity profile, the maximum distance approach has a great superiority regarding the accuracy over iterative methods, as it determines a group in a more straightforward manner by merely using a hard distance threshold. When applying to a large driving cycle dataset, the divided-and-conquered algorithm could be employed to further reduce the computational complexity in calculating the distances among varying points. The clustering results are also highlighted in Fig. 4. A group consisting of fewer than three points is ignored and will not be treated as an intersection or a stop, given a low stopping probability of 6.4% (2/31).

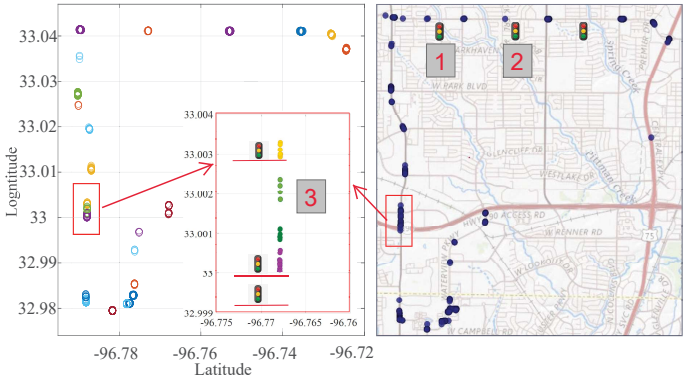


FIGURE 4: Intersection detection using the velocity profile and its lactation on a map. The left figure shows the locations of intersections after identification. The onward route before the first stop sign has been removed, so has the return route. Annotations 1 and 2 illustrate that these two intersections are equipped with traffic lights but will be ignored as a normal straight road due to a low stopping probability. Annotation 3 indicates that vehicles are very likely to stop moving somewhere between two intersections because of heavy traffic, which may be regarded as a stop.

2.4 Road Segment

To better model the velocity trajectory passing through an intersection, besides the final stopping location, it is also crucial to identify the deceleration and reacceleration processes and divide the routes into varying segments, as illustrated in Fig. 5. An intersection segment consists of a deceleration, a waiting, and a reacceleration process, while a normal road segment refers to a continuous move at a steady speed. When it comes to the local street with a lower speed limit and a smaller traffic volume, i.e., a stop sign, the deceleration and acceleration prepossess are captured with a very similar pattern, i.e., same stopping locations and almost equal waiting time.

Following the principle discussed above, the whole cycle is divided into 42 segments using location coordinates, and a portion of the segments are presented in Figs. 6 and 7. It is seen from the figures that within a same road segment, the velocity patterns may still differ, especially for intersection segments. However, it is observed from Fig. 7 that the trend of velocity trajectories versus location for intersection segments are more uniformed than that versus time in Fig. 6. The possible discrepancies are mainly due to the waiting time for traffic lights. Thus, no further clustering is implemented in this study. We also assume that the final stopping location may have considerable influences on the waiting time at an intersection, which is crucial to vehicle energy management and will be further investigated in the following section.

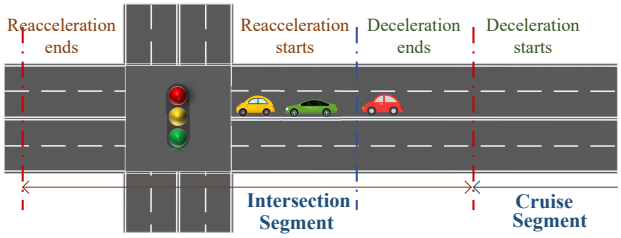


FIGURE 5: The schematic diagram for road segment division

3 Velocity Forecasting Methods and Results

From the perspective of energy management with a specific driving route, it is expected that accurate traffic forecasting, including the averaged velocity, the deceleration and acceleration processes, and the waiting time on intersections in particular, could significantly enhance the efficiency via energy scheduling and planning. In this study, we develop a method to dynamically select the LSTM or the HMM model to forecast the velocity at each segment, and also develop a back-propagation neural network (BPNN) model to estimate the waiting time at intersections.

3.1 Long-short Term Memory Model

Long-short Term Memory Model is an improvement over the recurrent neural network (RNN) with feedback connections designed to address the long-term dependency challenge when modeling sequential events by propagating through time. In addition to existing structures of RNN like hidden states, LSTM employs a novel layer named cell states, which is utilized to selectively store the previous event information and alleviate the vanishing/exploding gradient issue.

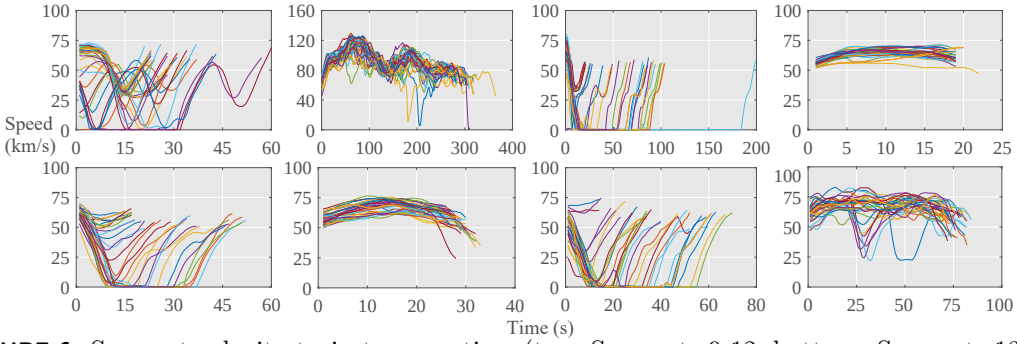


FIGURE 6: Segment velocity trajectory *vs.* time (top: Segments 9-12; bottom: Segments 13-16)

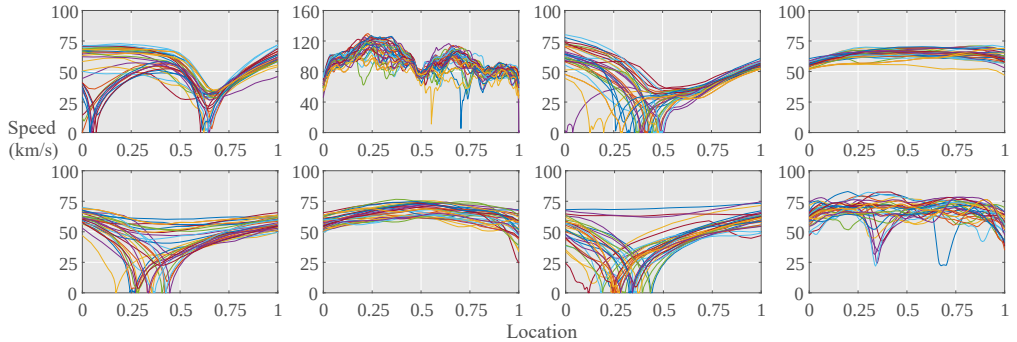


FIGURE 7: Segment velocity trajectory *vs.* location (top: Segments 9-12; bottom: Segments 13-16)

In this study, a typical LSTM structure with a forget gate and around 225 hidden units is employed to forecast velocity 10 steps/seconds ahead, which takes approximately 300 epochs to converge with the adaptive learning rate optimization algorithm Adam. A total of 25 randomly-selected cycles are utilized as the training data, and another 3 cycles are used for verification. The detailed settings of the models may differ from segment to segment. The forecasting outputs of both the segment-based method and the whole cycle-based method are presented in Figs. 8(a) and 8(b), respectively.

To evaluate the accuracy of the forecasts, two evaluation metrics including the mean absolute error (MAE) and the root-mean-square error (RMSE) are adopted here, expressed as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{2}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{3}$$

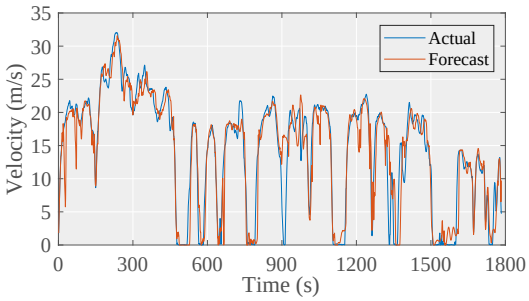
where $\hat{y}_i$ and y_i are the forecasted and actual value of sample index i, respectively. The segment-based forecast shows an MAE of 1.91 m/s and an RMSE of 3.2 m/s; while the MAE and RMSE of the cycle-based forecast are 2.52 m/s and 3.78 m/s, respectively. The segment-based forecast has reduced the MAE and RMSE by 24% and 15%, respectively, compared the whole cycle-based forecast. It is also observed from the figures that the segment-based forecast tends to follow the general features within the cycle segment. For example, it mistakenly predicts two decelerations and misses one moving stop. Overall, the results validate the effectiveness and importance of the segment analysis discussed in Section 2.

3.2 Hidden Markov Chain Model

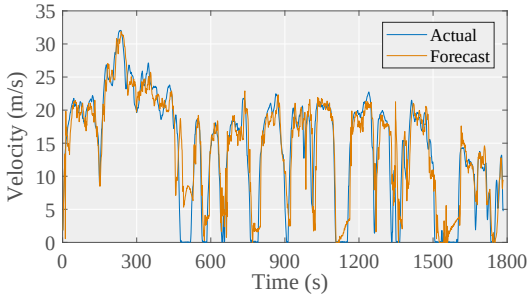
Hidden Markov chain is a discrete-time stochastic memory-less process to model a sequence of events, in which the future state or action only depends on the current state. The chaining process is characterized by a set of implicit hidden states S and its transition probabilities matrix A, and each a_{ij} represents the probability of moving from a state i to a new state j, s.t. $\sum_{j=1}^{N} a_{ij} = 1, \forall i$, which is also referred to as the Markov assumption, expressed as:

$$P(s_i|s_{i-1}, s_{i-2}, \cdot, \cdot, \cdot, s_1) = P(s_i|s_{i-1}) \tag{4}$$

Another fundamental assumption of HMM is that the explicit observation O only relies on the state that generates the observation with a probability of $B = [b_j(O_t)]$. Starting with an initial probability distribution over states π, an HMM process can be modelled as $\lambda = (A, B, \pi)$.

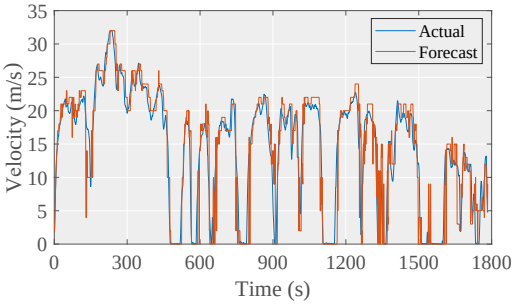


(a) Segmented-based forecasting (MAE: 1.91 m/s, RMSE: 3.2 m/s)

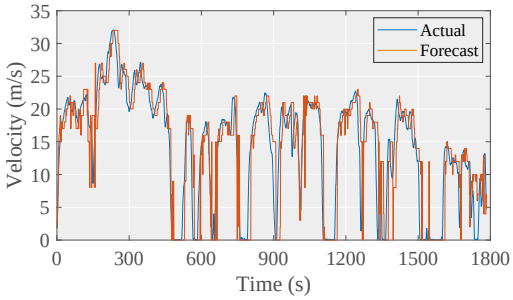


(b) Cycle-based forecasting (MAE: 2.52 m/s, RMSE: 3.78 m/s)

FIGURE 8: A comparison between segment- and cycle-based LSTM forecasting



(a) 2 steps ahead forecasting based on a 5-second interval (MAE: 2.01 m/s, RMSE: 3.33 m/s)



(b) 5 steps ahead forecasting based on a 2-second interval (MAE: 2.86 m/s, RMSE: 4.52 m/s)

FIGURE 9: Different approaches in segment-based HMM forecasting

To apply HMM for velocity prediction, it is required to transform the historical speed records into a set of observations indexed by integers. If the numbers of states and observations are known and set to be equal, the probability matrix A and B can be solved by calculating the frequency counts of a labeled state transition among all the transitions or a specific observation among the observations. In this study, the DRD dataset contains a very limited number of data points, posing challenges in directly solving this problem, as the matrix might be singular. Here, we obtain a model by employing the Baum-Welch algorithm that treats the hidden states as implicit variables using an expectation maximization algorithm, and the detailed derivation and explanation can be found in Ref. [39]. After the model is trained, the prediction procedure uses a sequence of velocity as the observation input, and a dynamic programming (i.e., Viterbi) algorithm as the solver to acquire the most probable state path. The prediction results can be achieved one step ahead along the HMM chain.

To predict the 10 seconds ahead velocity, approaches with different step windows can be used, e.g., 5 steps ahead with a 2-second interval or 2 steps ahead with a 5-second interval. The reason to adopt these recursive multi-step approaches is that most of the decelerations and accelerations in the DRD dataset occur within 12 seconds, and it is challenging for a one-step 10-second ahead direct prediction to capture these processes using HMM. The 2 steps and 5 steps ahead predictions are shown in Figs. 9(a) and 9(b), respectively. The recursive 2-step method yields an MAE of 2.01 m/s and an RMSE of 3.33 m/s, compared to the 2.86 m/s MAE and the 4.52 m/s RMSE of the 5-step method. The accuracy discrepancies are mainly due to error accumulations, where a double recursion performs better. Compared with the LSTM forecasting, there exist several drastic fluctuations in the HMM forecasting, which may lead to undesired disturbances when applying to the energy management system, though both the LSTM and HMM accuracies are similar. One potential reason to cause these drastic fluctuations is due to the sparsity of the transition and observation matrix, which could be mitigated by collecting more driving cycles or leveraging data augmentation techniques [34]. Moreover, the whole cycle-based HMM forecast shows a lower accuracy than the segment-based HMM, with an MAE of 3.36 m/s and an RMSE of 5.13 m/s, which also validates the effectiveness of segment-based forecasting.

3.3 Localized Model Selection

We have observed that the accuracies of LSTM and HMM models differ from segment to segment. To further

improve the forecasting accuracy, a localized model selection framework is developed for determining the best model of each segment with a higher accuracy. Specifically, the LSTM model acts as a base model due to its higher global accuracy. For certain segments, the LSTM model will be replaced by an HMM model if the MAE difference is higher than a threshold, which is set to be 1 m/s for this DRD dataset. Ideally the model selection framework should be established by extensive comparisons based on a large number of historic records and forecasting outcomes. However, due to the lack of data, we only utilize the comparative results of two driving cycles conservatively, as shown in Fig. 10. A group of 8 segments are obtained with better performances when using the HMM model. Another potential method for model selection is to utilize a reinforcement learning approach as illustrated in Ref. [40].

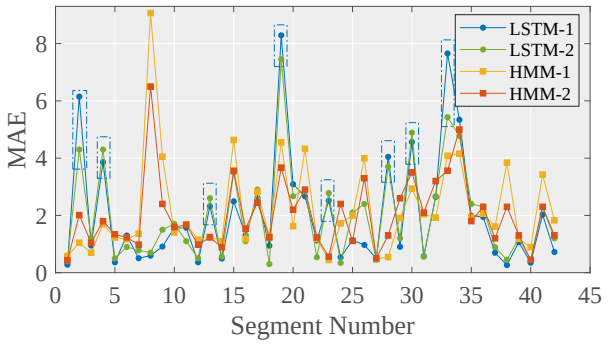


FIGURE 10: The segmental error distribution of two driving cycles. The dashed boxes indicate the segments where HMM dominates LSTM, i.e., 2, 4, 13, 19, 23, 28, 30, 33. LSTM-1 and HMM-1 represent one driving cycle; LSTM-2 and HMM2 represent the other driving cycle.

The localized model selection framework is applied to velocity forecasting with a new driving cycle. The forecasting results of both the hybrid model and the base LSTM model are compared in Fig. 11. It is observed that the hybrid forecasting based on the localized and segmental models has improved the segment-based LSTM forecasting accuracy by 6.8%, as tabulated in Table 1. This improvement mainly results from the replacement of LSTM with HMM in these predefined segments, as illustrated in Fig. 12. The model selection yields an accuracy of 75% (6/8) and a recall (i.e., the ratio of the segments that have passed the accuracy threshold but not been selected) of 25% (2/8). Overall, the localized model selection framework is able to further improve the forecasting accuracy for this DRD dataset.

TABLE 1: Global forecasting error evaluation

	HMM	LSTM	Hybrid
MAE (m/s)	1.96	1.75	1.63
RMSE (m/s)	3.19	2.65	2.55

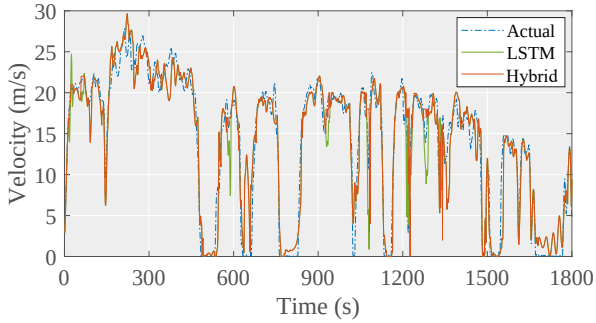


FIGURE 11: Forecasts based on localized model selection. The solid line in green represents the baseline segment-based LSTM forecast)

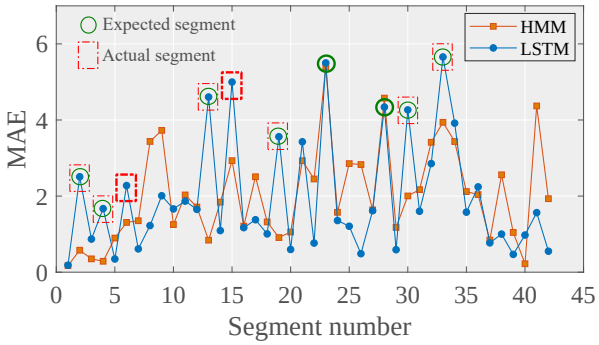


FIGURE 12: The segmental error distribution between LSTM and HMM. The solid circles indicate the predefined segments where HMM will replace LSTM, and the dashed boxes indicate the actual segments that should be replaced by HMM.

3.4 Estimation of Waiting time

As discussed in Section 2.4, it is assumed that the waiting time at an intersection can be estimated based on the stopping location at roughly the same time period of a day. We extract all the intersections and the waiting time in the dataset, and establish BPNN models for the intersection segments. The neuron numbers in the two hidden layers are different, ranging from 2 to 6 among varying cases, which are determined via an exhausting search method. The model uses 25 cycles for training, 3 cycles for validation, and the remaining 3 cycles for testing. The actual and

estimated waiting time of one testing case are compared in Fig. 13, yielding an overall averaged MAE of 18.36 seconds and an RMSE of 25.67 seconds. This is an acceptable result, since the waiting time estimation is mainly used to help schedule the operations of the on-board air conditioning system and other systems to avoid overlapping with the motor energy demand during acceleration.

The waiting time highly depends on arrival time counting based on the traffic light cycles. The mechanism behind this BPNN approach is that the arrival time can be estimated based on the location of the vehicle under the condition of a constant traffic volume. However, the location of the vehicle could be affected by the actual space between two vehicles. An alternative way to further improve the waiting time estimation accuracy is to analyze the traffic signal via image detection, which will be further investigated in our future study.

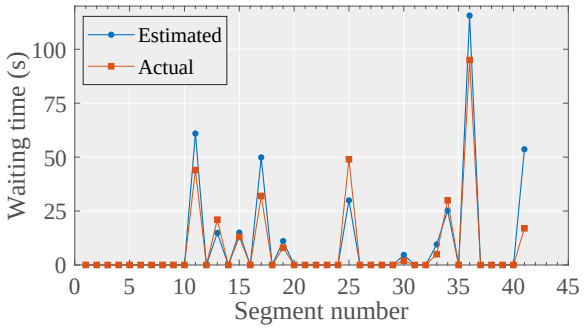


FIGURE 13: The estimated waiting time *vs.* the actual waiting time

4 Conclusion

This paper generated a repeated urban driving cycle dataset at a fixed route in the Dallas area. Based on the data preprocessing and intersection identification, cycle segmentation was conducted to provide location-dependent segmental data for improving velocity forecasting. A segment-based velocity forecasting method was developed, by using LSTM and HMM, to perform 10 seconds ahead forecasting. Results showed that the segment-based LSTM forecast has reduced the MAE and RMSE by 24% and 15%, respectively, compared the whole cycle-based LSTM forecast.

To further improve the forecasting accuracy, a hybrid approach based on a localized model selection framework was developed. Specifically, the LSTM model was selected as a base model, which will be replaced by the HMM model at certain segments, when the LSTM has a lower accuracy. Results showed that a 6.8% improvement was obtained from the localized model selection framework. A BPNN-based intersection waiting time estimation model was also established and validated with an acceptable accuracy. The improvements in both velocity forecasting and waiting time estimation will lead to better energy management, especially for electric or hybrid vehicles.

Potential future work will (i) further improve the localized model selection by using more data or exploring reinforcement learning for model selection, and (ii) detect/identify the traffic signals via CNN-based image identification techniques to further improve the waiting time estimation.

REFERENCES

[1] Torre-Bastida, A. I., Del Ser, J., Laña, I., Ilardia, M., Bilbao, M. N., and Campos-Cordobés, S., 2018. "Big data for transportation and mobility: recent advances, trends and challenges". *IET Intelligent Transport Systems,* ***12***(8), pp. 742–755.

[2] Vlahogianni, E. I., Karlaftis, M. G., and Golias, J. C., 2014. "Short-term traffic forecasting: Where we are and where we're going". *Transportation Research Part C: Emerging Technologies,* ***43***, pp. 3–19.

[3] Lana, I., Del Ser, J., Velez, M., and Vlahogianni, E. I., 2018. "Road traffic forecasting: Recent advances and new challenges". *IEEE Intelligent Transportation Systems Magazine,* ***10***(2), pp. 93–109.

[4] Guo, J., and Williams, B. M., 2010. "Real-time short-term traffic speed level forecasting and uncertainty quantification using layered kalman filters". *Transportation Research Record,* ***2175***(1), pp. 28–37.

[5] Wang, J., and Shi, Q., 2013. "Short-term traffic speed forecasting hybrid model based on chaos–wavelet analysis-support vector machine theory". *Transportation Research Part C: Emerging Technologies,* ***27***, pp. 219–232.

[6] Ye, Q., Szeto, W. Y., and Wong, S. C., 2012. "Short-term traffic speed forecasting based on data recorded at irregular intervals". *IEEE Transactions on Intelligent Transportation Systems,* ***13***(4), pp. 1727–1737.

[7] Jiang, B., and Fei, Y., 2015. "Traffic and vehicle speed prediction with neural network and hidden markov model in vehicular networks". In 2015 IEEE Intelligent Vehicles Symposium (IV), IEEE, pp. 1082–1087.

[8] Zhao, Z., Chen, W., Wu, X., Chen, P. C., and Liu, J., 2017. "Lstm network: a deep learning approach for short-term traffic forecast". *IET Intelligent Transport Systems,* ***11***(2), pp. 68–75.

[9] Lee, K., Eo, M., Jung, E., Yoon, Y., and Rhee, W., 2020. "Short-term traffic prediction with deep neural networks: A survey". *arXiv preprint arXiv:2009.00712*.

[10] Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., and Wang,

Y., 2017. "Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction". *Sensors,* ***17***(4), p. 818.

[11] Wu, Y., and Tan, H., 2016. "Short-term traffic flow forecasting with spatial-temporal correlation in a hybrid deep learning framework". *arXiv preprint arXiv:1612.01022*.

[12] Yu, H., Wu, Z., Wang, S., Wang, Y., and Ma, X., 2017. "Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks". *Sensors,* ***17***(7), p. 1501.

[13] Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., and Li, H., 2019. "T-gcn: A temporal graph convolutional network for traffic prediction". *IEEE Transactions on Intelligent Transportation Systems*.

[14] Lv, M., Hong, Z., Chen, L., Chen, T., Zhu, T., and Ji, S., 2020. "Temporal multi-graph convolutional network for traffic flow prediction". *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12.

[15] Mengzhang, L., and Zhanxing, Z., 2020. "Spatial-temporal fusion graph neural networks for traffic flow forecasting". *arXiv preprint arXiv:2012.09641*.

[16] Fan, X., Xiang, C., Gong, L., He, X., Qu, Y., Amirgholipour, S., Xi, Y., Nanda, P., and He, X., 2020. "Deep learning for intelligent traffic sensing and prediction: recent advances and future challenges". *CCF Transactions on Pervasive Computing and Interaction*, pp. 1–21.

[17] Liu, Y., and Zhang, J. "Electric vehicle battery thermal and cabin climate management based on model predictive control". *Journal of Mechanical Design,* ***143***(3).

[18] Liu, Y., and Zhang, J., 2020. "Self-adapting j-type air-based battery thermal management system via model predictive control". *Applied Energy,* ***263***, p. 114640.

[19] Liu, Y., and Zhang, J., 2019. "Design a j-type air-based battery thermal management system through surrogate-based optimization". *Applied Energy,* ***252***, p. 113426.

[20] Wang, X., Liu, Y., Sun, W., Song, X., and Zhang, J., 2018. "Multidisciplinary and multifidelity design optimization of electric vehicle battery thermal management system". *Journal of Mechanical Design,* ***140***(9), p. 094501.

[21] Wang, X., Li, M., Liu, Y., Sun, W., Song, X., and Zhang, J., 2017. "Surrogate based multidisciplinary design optimization of lithium-ion battery thermal management system in electric vehicles". *Structural and Multidisciplinary Optimization,* ***56***(6), pp. 1555–1570.

[22] Amini, M. R., Kolmanovsky, I., and Sun, J., 2021. "Hierarchical mpc for robust eco-cooling of connected and automated vehicles and its application to electric vehicle battery thermal management". *IEEE Transactions on Control Systems Technology,* ***29***(1), pp. 316–328.

[23] Zhou, Y., Ravey, A., and Péra, M.-C., 2019. "A survey on driving prediction techniques for predictive energy management of plug-in hybrid electric vehicles". *Journal of Power Sources,* ***412***, pp. 480–495.

[24] Le Rhun, A., Bonnans, F., De Nunzio, G., Leroy, T., and Martinon, P., 2019. "A bi-level energy management strategy for hevs under probabilistic traffic conditions".

[25] Jing, J., Filev, D., Kurt, A., Özatay, E., Michelini, J., and Özgüner, Ü., 2017. "Vehicle speed prediction using a cooperative method of fuzzy markov model and auto-regressive model". In 2017 IEEE Intelligent Vehicles Symposium (IV), IEEE, pp. 881–886.

[26] Zhou, Y., Ravey, A., and Pera, M.-C., 2019. "A velocity prediction method based on self-learning multi-step markov chain". In IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society, Vol. 1, IEEE, pp. 2598–2603.

[27] Sun, C., Hu, X., Moura, S. J., and Sun, F., 2014. "Velocity predictors for predictive energy management in hybrid electric vehicles". *IEEE Transactions on Control Systems Technology,* ***23***(3), pp. 1197–1204.

[28] Liu, K., Asher, Z., Gong, X., Huang, M., and Kolmanovsky, I., 2019. Vehicle velocity prediction and energy management strategy part 1: Deterministic and stochastic vehicle velocity prediction using machine learning. Tech. rep., SAE Technical Paper.

[29] Gaikwad, T., Rabinowitz, A., Motallebiaraghi, F., Bradley, T., Asher, Z., Fong, A., and Meyer, R., 2020. Vehicle velocity prediction using artificial neural network and effect of real world signals on prediction window. Tech. rep., SAE Technical Paper.

[30] Lemieux, J., and Ma, Y., 2015. "Vehicle speed prediction using deep learning". In 2015 IEEE Vehicle Power and Propulsion Conference (VPPC), pp. 1–5.

[31] Moser, D., Waschl, H., Schmied, R., Efendic, H., and del Re, L., 2015. "Short term prediction of a vehicle's velocity trajectory using its". *SAE International Journal of Passenger Cars-Electronic and Electrical Systems,* ***8***(2015-01-0295), pp. 364–370.

[32] Zhang, F., Xi, J., and Langari, R., 2016. "Real-time energy management strategy based on velocity forecasts using v2v and v2i communications". *IEEE Transactions on Intelligent Transportation Systems,* ***18***(2), pp. 416–430.

[33] Johannesson, L., Asbogard, M., and Egardt, B., 2007. "Assessing the potential of predictive control for hybrid vehicle powertrains using stochastic dynamic programming". *IEEE Transactions on Intelligent Transportation Systems,* ***8***(1), pp. 71–83.

[34] Shin, J., Yeon, K., Kim, S., Sunwoo, M., and Han, M., 2021. "Comparative study of markov chain with recurrent neural network for short term velocity prediction implemented on an embedded system". *IEEE Access,* ***9***, pp. 24755–24767.

[35] Oh, G., Leblanc, D. J., and Peng, H., 2019. "Vehicle energy dataset (ved), a large-scale dataset for vehicle energy consumption research". *arXiv preprint arXiv:1905.02081*.

[36] NCTCG. Government report transportation system management. [Online] Available at: `https://www.nctcog.org/trans/manage/its/transportation-systems-management`.

[37] Cheng, S., Lu, F., and Peng, P., 2020. "Short-term traffic forecasting by mining the non-stationarity of spatiotemporal patterns". *IEEE Transactions on Intelligent Transportation Systems*.

[38] Nair, D. J., Gilles, F., Chand, S., Saxena, N., and Dixit, V., 2019. "Characterizing multicity urban traffic conditions using crowdsourced data". *PLoS one,* ***14***(3), p. e0212845.

[39] Blunsom, P., 2004. "Hidden markov models". *Lecture notes, August,* ***15***(18-19), p. 48.

[40] Feng, C., Sun, M., and Zhang, J., 2020. "Reinforced deterministic and probabilistic load forecasting via q -learning dynamic model selection". *IEEE Transactions on Smart Grid,* ***11***(2), pp. 1377–1386.