2020

Towards Attention based ConvLSTM for Long-Term Travel Time Prediction of Bus Journey

Jianqing Wu
University of Wollongong, jw937@uowmail.edu.au

Qiang Wu

Jun Shen
University of Wollongong, jshen@uow.edu.au

Chen Cai

Towards Attention based ConvLSTM for Long-Term Travel Time Prediction of Bus Journey

Abstract

Travel time prediction is critical for advanced travelerinformation systems (ATISs), which provides valuable information for enhancing the efficiency and effectiveness of the urban transportation systems. However, in the area of bus trips, existing studies have focused on directly using the structured data to predict travel time for a single bus trip. For state-of-the-art public transportation information systems, a bus journey generally has multiple bus trips. Additionally, due to the lack of study on data fusion, it is even inadequate for the development of underlying intelligent transportation systems. In this paper, we propose a novel framework for a hybrid data- driven travel time prediction model for bus journeys based on open data. We explore a convolutional long short-term memory (ConvLSTM) model with a self-attention mechanism that accurately predicts the running time of each segment of the trips and the waiting time at each station. The model is more robust to capture long-range dependence in time series data as well.

Keywords

convlstm, long-term, attention, bus, towards, travel, time, prediction, journey

Disciplines

Engineering | Science and Technology Studies

Publication Details

Towards Attention-Based Convolutional Long Short-Term Memory for Travel Time Prediction of Bus Journeys

Jianqing Wu [1], Qiang Wu [2], Jun Shen [1,*] and Chen Cai [3]

[1] School of Computing and Information Technology, University of Wollongong, Wollongong, NSW 2522, Australia; jw937@uowmail.edu.au

[2] School of Information and Engineering, Lanzhou University, Lanzhou 730000, China; wuq17@lzu.edu.cn

[3] Data 61, CSIRO, Eveleigh, NSW 2015, Australia; chen.cai@data61.csiro.au

* Correspondence: jshen@uow.edu.au

Abstract: Travel time prediction is critical for advanced traveler information systems (ATISs), which provides valuable information for enhancing the efficiency and effectiveness of the urban transportation systems. However, in the area of bus trips, existing studies have focused on directly using the structured data to predict travel time for a single bus trip. For state-of-the-art public transportation information systems, a bus journey generally has multiple bus trips. Additionally, due to the lack of study on data fusion, it is even inadequate for the development of underlying intelligent transportation systems. In this paper, we propose a novel framework for a hybrid data-driven travel time prediction model for bus journeys based on open data. We explore a convolutional long short-term memory (ConvLSTM) model with a self-attention mechanism that accurately predicts the running time of each segment of the trips and the waiting time at each station. The model is more robust to capture long-range dependence in time series data as well**.**

1. Introduction

The usage of intelligent transportation systems (ITSs) is motivated in a significant part by passenger increases and sustainable development [1,2]. The ITS has a direct impact on energy consumption, personal living expenses, public health and safety. Seamless integration of vehicles and sensing devices has made it possible to capture and collect large amounts of sensor data from various data sources in real time. Developing sustainable and intelligent transportation applications operate and manage real-time and historical data efficiently, which has become an increasingly important yet challenging task. It also plays a vital role in achieving the main objectives of ITS, which include accessibility and mobility, environmental sustainability and economic development [3,4]. With the advent of artificial intelligence (AI), machine learning and expert system-based paradigms have driven the development of society and the steady growth of the economy. Besides, deep learning can discover patterns in complex data sets, which could not be found via conventional methods. The merging of machine learning and transportation science has tremendous potential to enhance the performance of ITS.

Travel time refers to a period spent traveling from the origin to the destination. Providing real-time travel information is indispensable for ITS. However, real-time travel time is unlikely to be observed because it is already historical data rather than 'real-time data' since it was collected [5].

Using predictive methods to estimate future travel time is an effective way to provide real-time information. Furthermore, travel time prediction is a known and challenging research area because of the inherent uncertainty [6]. Existing studies on bus travel time prediction mainly focus on improving the prediction accuracy of a single trip. This is inadequate for implementing efficient applications in an intelligent transportation system, where a bus journey has multiple bus trips [7]. Although the ConvLSTM has shown excellent performance in travel time prediction, adding the attention mechanism to LSTM-based models has the potential to improve the predictive accuracy [8,9]. The integration of their strengths remains an unsolved research task. Studies have applied LSTM-based deep learning methods with applications to journey travel time prediction that rely on high-quality labeled data. However, data acquisition is a challenging task.

The contributions of this study are summarized as follows:

(1) We designed and developed an open-source data collection framework that can automatically collect and pre-process large amounts of high-quality data over a long period without involving personal privacy, for example, an entire season or even several years.
(2) This paper proposes a hybrid model that applies the ConvLSTM network with an attention mechanism to explore a suitable model for the bus journey time prediction on open data.
(3) We also discuss input features for journey travel time prediction and suggest directions for future research.

The remainder of the paper is organized as follows. Firstly, we demonstrate a brief overview of the basic definitions. Secondly, an integrated system framework is introduced to target the problem of bus journey time prediction and provides a ConvLSTM-based method with self-attention. Furthermore, the datasets' baseline and evaluation metrics are used in this study. Finally, the findings and suggestions for further studies are summarized.

2. Related Works

The sustainable development of *smart cities* requires reliable and efficient transportation systems [10]. Internet of Things (IoT) can be applied with the existing infrastructure and service networks for the design of transportation systems, such as software-defined networks and communication technologies [11–13]. IoT-based intelligent transportation systems (IoT-ITSs) can be classified into four main fields: Advanced traveler information system (ATIS), advanced public transportation system (APTS), advanced traffic management system (ATMS) and emergency management system (EMS) [13]. Transportation systems are shifting from conventional technology-driven systems to more powerful multifunctional data-driven ITSs [14–16]. Massive traffic sensor data gathered by various sensors are vital for informed scientific decision-making processes in traffic operation, pavement design and transportation planning [17]. Data analytics in ITSs consider important factors that influence decision-making processes, such as travel time or traffic congestion of public transport services [18,19]. The fusion of traffic data from multiple sources produces a better understanding of the observations to reach a better inference in ITSs [20–23].

Accurate estimation of travel time is essential to the success of ATMS and ATIS [24]. The approaches to studying travel time prediction can be mainly divided into three categories: Knowledge-driven, model-driven and data-driven approaches. Knowledge-driven approaches usually employ a database, a knowledge base in the form of rules and an inference engine in the form of algorithms [25]. Lee et al. proposed a knowledge-based expert system that predicted travel time by combining general rules from location-based service applications and meta-rules from human domain experts [26]. Nonetheless, as the knowledge base becomes increasingly large, the time to obtain accurate predictions increases as well. Model-driven approaches can be divided into four levels: Macroscopic (e.g., TOPL [27]), mesoscopic (e.g., DynaMIT [28] and Dynasmart [29]), cellular automaton (CA) (e.g., OLSIM [30]) and microscopic methods (e.g., AIMSUM online [31]) [32]. In the past, most of the studies on travel time forecasting have focused on model-based methods. Transport simulation software is intended for simulating traffic state information on virtual networks. It is primarily focused on research in traffic control and management, such as the effects of ramp

metering, variable speed limits and traffic incidents. To perform research on model-based practices, we need to acquire and use travel demand data, which is known as an origin-destination (OD) matrix or population data [5]. Nevertheless, accurate OD data is difficult to obtain, time-consuming and expensive. Presently, only a few institutions have accumulated essentially useful OD data to build integrated travel time forecasting systems.

Recently, data-driven approaches have been receiving increased attention and gained interest within the transportation research community due to the increased computing power available and the vast amount of data collected in ITSs. Deep learning leads to an advantage over conventional machine learning algorithms with big data analytics of urban traffic. Kumar et al. compared the performance of the data-driven artificial neural network (ANN) approach and the model-based Kalman filter (KF) approach concerning bus travel time prediction in [33]. The experimental results showed that the data-driven ANN can achieve better performance, but compared to KF, the model needs a rich set of data for neural network training. Hou and Edara proposed long short-term memory (LSTM) and convolutional neural network (CNN) to predict travel time in a road network; compared to CNN, random forests (RFs) and gradient boosting machines (GBMs); the computation time of LSTM was the shortest in the model training process and prediction process [34]. Petersen et al. utilized the convolutional LSTM to propose a multi-output multi-time-step system for bus travel time prediction [8]. Yu et al. presented a random forest based on the near neighbor (RFNN) model to predict the travel times of buses between bus stops, which include the running time and waiting time as two input variables separately. Correspondingly, the model also considers traffic conditions, which is an essential factor affecting bus travel time [35]. However, studies on bus journey time forecasting is rather limited. Our work focuses on forecasting the travel time of the bus journey for travelers. A trip is to use one transport mode to travel on a single line or route, and a journey has one or more trips, where transfers occur between bus services during a period of travel time [7]. Therefore, there is still a need for developing a well-designed system framework to discover the advantages of various methods that achieve a deterministic and meaningful outcome, which is closer to the real world's needs.

However, none of the existing studies have considered the travel time problem of a bus journey via the ConvLSTM with the self-attention mechanism. Thus, the objective of our study was to predict the travel time of bus journeys by leveraging a data fusion component, which offers appropriate inputs to deep learning models.

3. Methodology

3.1. Bus Travel Time

In this section, we define some terms in Table 1, which will be used throughout the rest of the paper.

Table 1. List of important notations.

Symbol	Description
T	bus trip id T
n	number of bus stops in T
S	a bus stop in a trip T
t_d	bus departure time from the station S
t_a	bus arrival time at the station S
t_{total}	total time of a trip T
R	actual running time in T
D	actual waiting time in T
$\hat{R}$	predicted running time in T
$\hat{D}$	predicted waiting time in T
Y	actual value of evaluation metrics
$\hat{Y}$	predicted value of evaluation metrics

A bus usually runs along a fixed route based on a regular schedule. The travel time depicted in Figure 1 is the time cost to complete a trip, which departs at time t. It follows an itinerary characterized by an original station A, a destination station B and some stops (e.g., station S_1 and station S_2).

Figure 1. Running time and waiting time for a bus trip.

In this paper, we predict the total travel time of a bus journey by using the actual running time and waiting time from open data. For any stops in the trip, a bus is scheduled to arrive and depart from a stop S at different specified times, defined in the timetable, respectively, $t_d(T,S)$ and $t_a(T,S)$. In general, travel time forecasting is an estimate of the trip from a station of origin to a station of destination. The running time is the absolute difference between the arrival time of the current station and the departure time of the previous station, such as $R_2 = t_a(T,S_2) - t_d(T,S_1)$. The waiting time is the absolute difference between the departure time and the arrival time in a fixed stop station, such as, $D_1 = t_d(T,S_1) - t_a(T,S_1)$.

Our study defines segments based on information about the stops of a trip pattern. The segment-based method divides the stop points into running time and waiting time segments. Our predictive models predict the running and waiting times based on different t_a and t_d. According to Figure 1, it is evident that the numbers of input data for the prediction of running time and waiting time are different. This is because for each trip of a specific bus, the running time will have one more record than the waiting time. The total travel time of a bus journey can be described with Equation (1):

$$t_{total} = \sum_i^n \hat{R} + \sum_i^{n-1} \hat{D}_i. \tag{1}$$

3.2. Leveraging Machine Learning and Logical Reasoning

With the rapid development of ITSs in recent years, data availability issues have always plagued researchers. Notably, the studies of multi-modal transport require a large amount of data from diverse data sources. Open data platforms release a variety of data that is freely available to everyone to reuse. Moreover, domain experts structure and classify data, such as general transit feed specification (GTFS) and GTFS-Realtime [36]. Researchers can create structured data, namely the process of data curation, for the corresponding studies through data cleansing and data fusion. To predict a complex and uncertain event, we need to have multiple sources of data to provide more information for generating a predictive model.

Figure 2 illustrates the framework of an integrated system for journey time prediction, which consists of six components: GTFS-Realtime and GTFS static data stores, data fusion, knowledge base, feature extraction, deep learning models, and running time prediction and waiting time prediction. As Figure 2 shows, in the first step, we collected data from two types of GTFS and cleansed them, for example, by deleting duplicate data and sorting the data in chronological order. In order to build a knowledge base, the data fusion approach plays an essential role. Data from different data sources sometimes cannot be integrated and saved into a relational database or a two-dimensional data format, due to some data failing to match one-to-one or one-to-many mapping relationships, such as the running time from the station S_1 to S_2 and probe vehicle speed data. The use of the knowledge base enables deep learning models to exploit logical reasoning from data. Applying domain knowledge to classify the raw data not only avoids the impact of irrelevant data but also reduces the computation time of the model. Furthermore, data fusion employs mathematical methods and

programming languages to synthesize useful information or inferences. The theoretical framework can also be developed as an extended version to involve verification mechanisms [37].
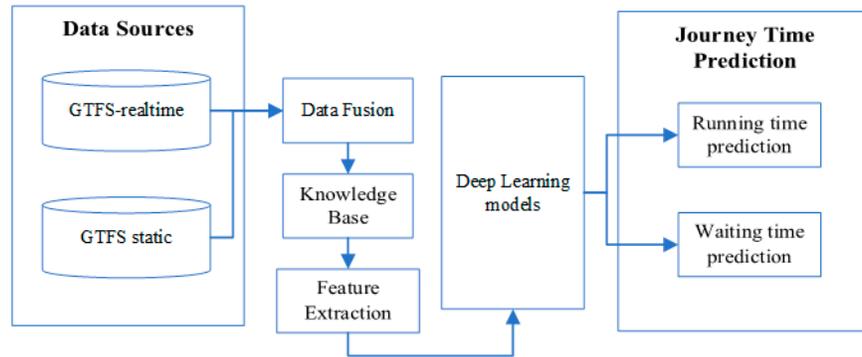
Figure 2. The framework of journey time prediction.

3.3. Bus Journey Travel Time with Multi-Step Time Series Prediction

The ConvLSTM model is a powerful kind of recurrent neural network (RNN), with a combination of convolutional and LSTM layers, which contains the operation inside the LSTM cell [38]. On the other hand, the travel time prediction of a bus journey can be treated as a time series prediction problem. In recent years, LSTM is an elegant solution to the time series analysis by exploiting spatiotemporal data. Additionally, the ConvLSTM applies the convolution operators to capture the spatial and temporal dependencies in the dataset so that it generally performs better than fully connected LSTM (FC-LSTM) [38]. The calculation steps are as follows:

Firstly, calculate the input gate:

$$i_t = \sigma(W_{xi} * x_t + W_{hi} * h_{t-1} + W_{ci} \circ c_{t-1} + b_i), \tag{2}$$

Forget gate:

$$f_t = \sigma(W_{xf} * x_t + W_{hf} * h_{t-1} + W_{cf} \circ c_{t-1} + b_f), \tag{3}$$

Cell state:

$$c_t = f_t \circ c_{t-1} + i_t \circ tanh(W_{xc} * x_t + W_{hc} * h_{t-1} + b_c), \tag{4}$$

Output gate:

$$o_t = \sigma(W_{xo} * x_t + W_{ho} * h_{t-1} + W_{co} \circ c_t + b_o), \tag{5}$$

Hidden state:

$$h_t = o_t \circ tanh(c_t), \tag{6}$$

where σ is a sigmoid function, $\circ$ is the Hadamard product, and $*$ is the convolution operator. W_{xi}, W_{xf}, W_{xc} and W_{xo} are the weight matrices connecting the inputs $x_1,…, x_t$ to three gates and the cell input; W_{hi}, W_{hf}, W_{hc} and W_{ho} are the weight matrices connecting the hidden states $h_1,…, h_{t-1}$ to three gates and the cell input; W_{ci}, W_{cf} and W_{co} are the weight matrices connecting the $c_1,…, c_t$ to three gates; and b_i, b_f, b_c and b_o are the bias terms of three gates and the cell state.

Recently, the attention mechanism has succeeded in a wide range of sequence-to-sequence learning tasks [39–41]. Liang et al. presented a multi-level attention-based recurrent neural network for predicting geo-sensory time series [42]. The attention model focuses on the vital issue with the LSTM-based model for bus travel time prediction, which tends to select near-term data that is highly correlated to future travel time. In our experiments, the encoder is the underlying ConvLSTM model generating the hidden state representation h_t. We leverage a self-attention mechanism to the inputs after the operations of Equations (1)–(6):

$$m_{t,t'} = tanh(W_m h_t + W_{m'} h_{t'} + b_m), \tag{7}$$

$$e_{t,t'} = \sigma(W_a m_{t,t'} + b_a), \tag{8}$$

$$a_t = softmax(e_t), \tag{9}$$

$$l_t = \sum_{t'=1}^{n} a_{t,t'} \cdot h_{t'}, \tag{10}$$

where $a_{t,t'}$ is an attention matrix; b_m and b_a express bias terms; W_m and $W_{m'}$ express weight matrices corresponding to the hidden states h_t, $h_{t'}$; and finally, l_t represents a weighted sum of $h_{t'}$ [43].

Figure 3 demonstrates an overview of our proposed model, which consists of two main components: Running time prediction and waiting time prediction, which are two independent components for estimating running and waiting times based on GTFS-Realtime. The first step is to divide the historical observations from a sequence dataset into two smaller sequence datasets so that the input data of the ConvLSTM model are arranged into a 3-D-tensor for a single bus line. For example, in N day samples and time steps k, a sequence of running times R_i with a single bus line can be represented as (N, k, R_i). Secondly, l_1 and l_2 show how much the weight of the historical observations affects the predicted values. Finally, the outputs are merged to get the results by using Equation (1).
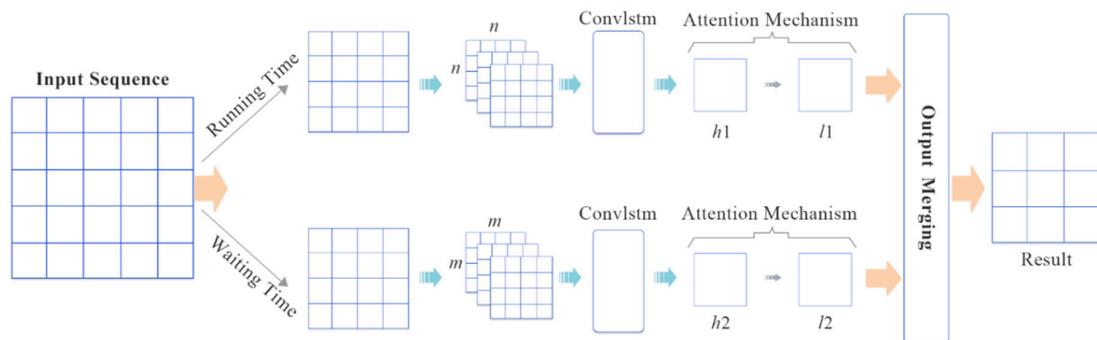

Figure 3. Self-attention-based ConvLSTM network.

The entire training process of an attention ConvLSTM is presented in Algorithm 1. We firstly construct multiple historical observation sequences as inputs. Then, the model is trained to predict the running time and waiting time separately.

Algorithm 1: Attention-based ConvLSTM Training Algorithm

Require:

Historical running time and waiting time observations:

$(R_1^T, R_2^T \ldots R_n^T)$ *and* $(D_1^T, D_2^T \ldots D_{n-1}^T)$;

Sequence length: n;

Lengths of running time, waiting time: l_R, l_D ;

running time: R;

waiting time: D.

Ensure: Attention-based ConvLSTM Model

for $epoch = 1$ to *max–epoch* do

Perform forward propagation recurrently using Equation (2)–(10) to

calculate

$$S_R = (R_1^T, R_2^T \ldots R_n^T)$$
$$S_D = (D_1^T, D_2^T \ldots D_n^T)$$

   compute output error:

$Y_R - \hat{Y}_R$

$Y_D - \hat{Y}_D$

merging the predicted outputs to obtain the total travel time:

$t_{total} = \hat{Y}_R + \hat{Y}_D$

end for

---

4. Experiments and Discussion

4.1. Dataset Description and Preprocessing

We verified our model on real-world traffic datasets from TfNSW (Transport for NSW) Open Data Bus Realtime Trip Update (BRTU) collected by a Python program that read the TfNSW real-time feed application programming interfaces (APIs) [44]. The dataset contains key attributes of bus journey information with corresponding timestamps, as detailed below.

BRTU was gathered from Sydney's bus system in real time. For our experiment, the data was collected every 60 s, about 12 GB of data a day. Note that the better frequency is 10 s, around 60 GB a day). The period used was from 6th May 2019 to 28th June 2019 except the weekends. We selected the first three weeks of historical travel time records as a training set and the rest served as a test set, respectively. BRTU has information about the departure time, arrival time, delay and route. The GTFS-static contains station names, coordinates and route names.

The proposed model and other comparative models were implemented in Python via the TensorFlow Framework [45] and trained with the Adam algorithm [46]. The proposed network was composed of several layers: A ConvLSTM2D [38], a flatten layer, a RepeatVector layer, a self-attention layer and two TimeDistributed layers. The training details about the network are presented in Table 2.

Table 2. Training details about self-attention-based ConvLSTM.

Variable	Value
learning rate	0.001
epochs	20
batch size	16
loss	Mean Squared Error
optimizer	Adam

4.2. Evaluation Metrics and Results

In our experiments, we applied two standard metrics to evaluate the performance of running time prediction and waiting time prediction, including root mean square errors (RMSEs) and mean absolute errors (MAEs). They were defined as presented in Equations (11) and (12), where y_t represents the actual value for sample t and $\hat{y}_t$ represents the predicted value. As the multi-time-step model predicts bus travel time for all stops for the next n time-steps, bot y_t and $\hat{y}_t$ have the dimensionality (N, k, R_i):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_t - \hat{y}_t)^2}, \tag{11}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_t - \hat{y}_t|. \tag{12}$$

We explored the patterns of the bus running time and waiting time on weekdays. Respectively, Tables 3 and 4 present the results of the trip id "27134" from Campbelltown station to Narellan Town Centre station. The trip "27134" has 37 records per day. As evidenced by the results, the performance of three types of LSTM does not have many differences. The output of our experiments is consistent with Greff et al.'s findings as well [47]. Standard LSTM and variant versions do not have significant performance differences.

Our design explores the pattern of each record (a stop). As can be seen from Tables 3 and 4, we found that the attention ConvLSTM is a more stable model by observing each prediction result. It adjusts the predictions reasonably based on previous inputs. However, it cannot model very long-range temporal dependencies (e.g., period and trend), and training becomes more complicated when the depth increases [48].

Simply put, when the amount of input data increases, the time calculated by the model will increase dramatically. The attention mechanism can effectively overcome the drawbacks of modeling long-range temporal dependencies. Additionally, it could reduce the computation time in every training by using less training data.

To further verify the performance, we used LSTM and attention-based ConvLSTM to predict the running time and waiting time of one of the stops, "Mt Annan Leisure Centre, Welling Dr" (stop 18). In Table 3, a significant difference is shown. By observing each predicted value of the CNN model, we find that there is a significant difference between the upper and lower bounds for the CNN model. In this case, the prediction of the model is very unreliable. Compared with the results of LSTM models, it can be seen that the forecast results are improved in Tables 3 and 4. Attention-based ConvLSTM's mean errors and standard deviation (SD) are the lowest. In conclusion, attention-based ConvLSTM achieves the best overall performance compared to the other models in Tables 3 and 4. It is a more reliable model for the prediction of travel time on data with large residuals than other models.

**Table 3.** Performance comparison of the bus running time prediction models for a stop.

Models	RMSE (s)		MAE (s)	
	Mean	SD	Mean	SD
CNN	121.770	15.350	115.095	18.318
LSTM	49.849	5.046	47.146	4.583
ConvLSTM	43.720	15.468	37.533	13.821
Attention-ConvLSTM	**41.449**	5.623	**36.328**	4.539

Table 4. Performance comparison of the bus waiting time prediction models for a stop.

Models	RMSE (s)		MAE (s)	
	Mean	**SD**	**Mean**	**SD**
CNN	7.891	6.415	6.912	1.747
LSTM	6.415	0.283	5.544	0.284
ConvLSTM	5.683	0.113	5.060	0.134
Attention-ConvLSTM	**3.740**	0.227	**3.166**	0.441

It is worth mentioning that our aim was not to solely improve the accuracy of predictions, as deep neural networks are less interpretable. Instead, we strived to find a practical data-driven model on open data by exploring the combination of deep learning methods and domain knowledge. Moreover, GTFS provides uncertainty values, which can be utilized to test the robustness of the generic model. The model based on GTFS will have a level of portability and reproducibility to the application in real scenarios.

Figure 4 reports the performance of CNN, LSTM, ConvLSTM and Attention-ConvLSTM for the prediction of the running time and waiting time. The *y*-axes of RMSE and MAE from (a), (b), (c) and (d) represent the errors in seconds, respectively. All models have significant prediction errors (mean and standard deviation) in running time predictions. Especially, CNN reaches the most significant prediction errors in all cases. The waiting times indicate small variations, which are to a great extent explained by the input in the corresponding models. A weak dependence on the journey travel time prediction is established. However, the variability of the running times cannot be fully explained by the selected input variables. Additionally, it shows that Attention-ConvLSTM effectively reduces errors. The proposed model needs to use more relevant factors to improve the predictions, such as vehicle speed or weather information.

(**a**)

(**b**)

(**c**)   (**d**)
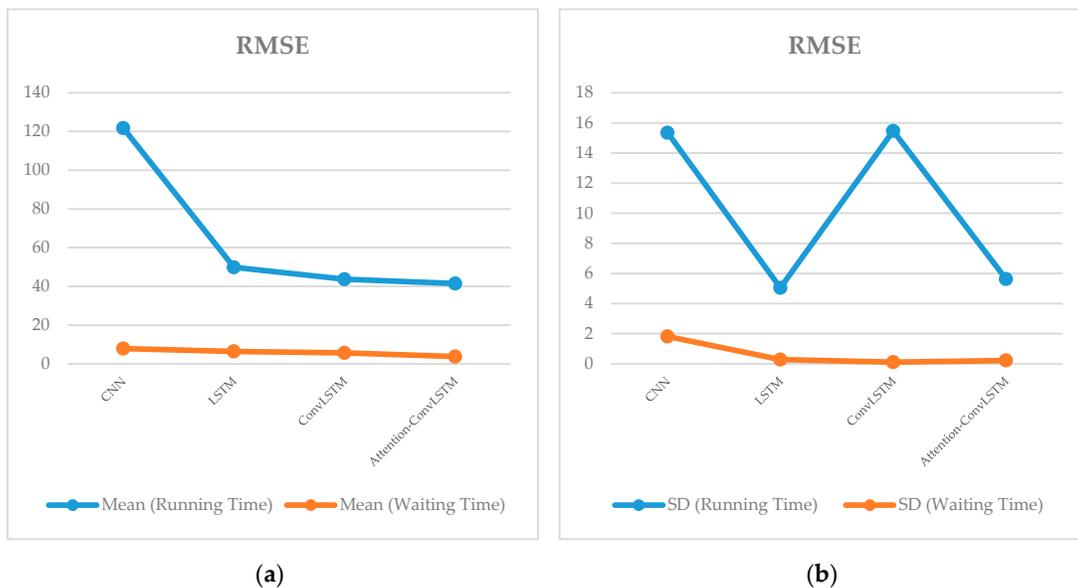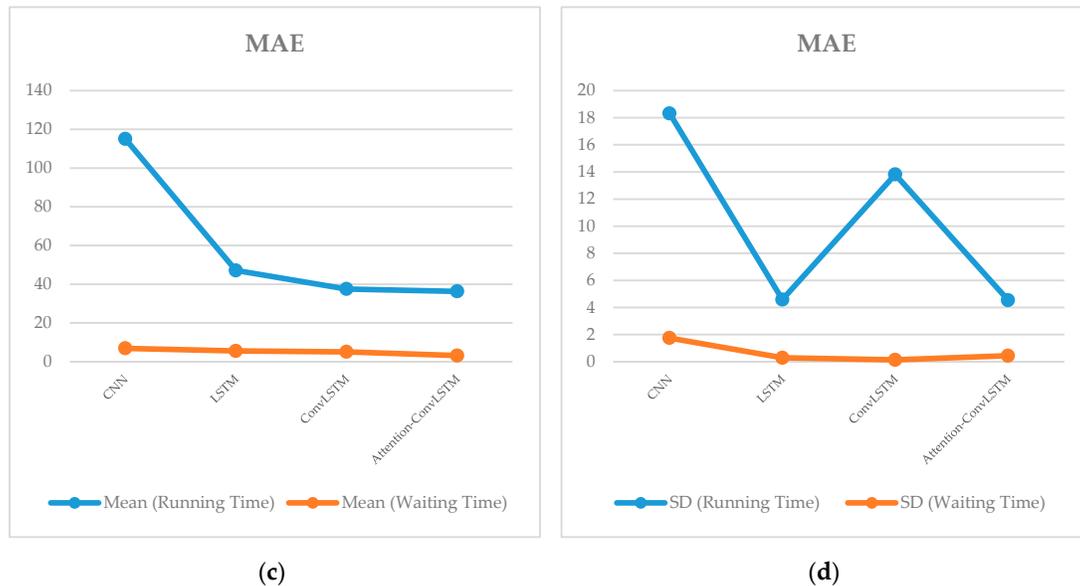
Figure 4. RMSE and MAE for the journey travel time prediction listed as: (**a**) The mean RMSE for the running time and waiting time; (**b**) The standard deviation of RMSE for the running time and waiting time; (**c**) The mean of MAE for running time and waiting time; (**d**) The standard deviation of MAE for the running time and waiting time.

5. Conclusions and Future Work

In this paper, we investigated the problem of predicting bus journeys' travel time with publicly available GTFS data by taking into account the bus running time along routes and the waiting time at stop points. The basic idea was to use domain knowledge to classify raw data to obtain a knowledge base, which can offer useful information for assisting in deep learning models to explore the hidden patterns of data. Thus, we proposed a comprehensive framework using open data to bridge deep learning models and logical reasoning from a knowledge base. We used an attention-based ConvLSTM to predict the running time and waiting time separately. Ultimately, the total travel time prediction was obtained by merging the predicted outputs.

In the future, we will consider adding weather information, vehicle speed and traffic condition data into our deep learning models. Furthermore, we will explore evolutionary algorithms to find the best dataset size for the accurate prediction of travel time, and to find the best model number of layers and number of units per layer. According to our experiments, the use of GTFS data exchanged API will make it easier to obtain high-quality input data for multi-modal traffic prediction studies. Our future work will also focus on employing more advanced data-driven models to shift from single-mode prediction to multi-modal prediction.

Author Contributions: J.W., J.S. and C.C., . conceived and designed the experimental setup and algorithms; J.W. and Q.W. developed main approaches; performed the experiments; Q.W. contributed data pre-processing and benchmarking data; and C.C. provided raw data. All authors contributed to discussion and analysis of the research and to writing of the paper. All authors have read and agreed to the published version of the manuscript.

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

References

1. Stawiarska, E.; Sobczak, P. The impact of Intelligent Transportation System implementations on the sustainable growth of passenger transport in EU regions. *Sustainability* **2018**, *10*, 1318.
2. U.N. ESCAP Information and Communications Technology and Disaster Risk Reduction Division. Intelligent Transportation Systems for Sustainable Development in Asia and the Pacific. 2015. Available online: http://www.unescap.org/sites/default/files/ITS.pdf (accessed on 3 May 2020).
3. Guerrero-ibanez, J.A.; Zeadally, S.; Contreras-Castillo, J. Integration challenges of intelligent transportation systems with connected vehicle, cloud computing, and internet of things technologies. *IEEE Wirel. Commun.* **2015**, *22*, 122–128, doi:10.1109/MWC.2015.7368833.
4. JSCE. Intelligent Transport Systems (ITS) Introduction Guide, 2016. Available online: https://arxiv.org/pdf/1412.6980.pdf (accessed on 11 June 2020). (accessed on 11 June 2020).
5. Duan,Y.j.; Lv, Y.S.;Wang, F.-Y. Travel time prediction with LSTM neural network. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 1053–1058.
6. Sullivan, A.O.; Pereira, F.C.; Zhao, J.; Koutso poulos, H.N. Uncertainty in Bus Arrival Time Predictions: Treating Heteroscedasticity With a Metamodel Approach. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3286–3296, doi:10.1109/TITS.2016.2547184.
7. He, P.; Jiang, G.; Lam, S.-K.; Tang, D. Travel-Time Prediction of Bus Journey With Multiple Bus Trips. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 4192–4205, doi:10.1109/TITS.2018.2883342.
8. Petersen, N.C.; Rodrigues, F.; Pereira, F.C. Multi-output bus travel time prediction with convolutional LSTM neural network. *Expert Syst. Appl.* **2019**, *120*, 426–435.
9. Ran, X.; Shan, Z.; Fang, Y.; Lin, C. An LSTM-based method with attention mechanism for travel time prediction. *Sensors.* **2019**, *19*, 861.
10. Menouar, H.; Guvenc, I.; Akkaya, K.; Uluagac, A.S.; Kadri, A.; Tuncer, A. UAV-Enabled Intelligent Transportation Systems for the Smart City: Applications and Challenges. *IEEE Commun. Mag.* **2017**, *55*, 22–28, doi:10.1109/MCOM.2017.1600238CM.
11. Maimaris, A.; Papageorgiou, G. A review of Intelligent Transportation Systems from a communications technology perspective. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 54–59.
12. Wu, J.; Huang, Y.; Kong, J.; Tang, Q.; Huang, X. A study on the dependability of software defined networks. In Proceedings of the International Conference on Materials Engineering and Information Technology Applications (MEITA 2015), Guilin, China, 30–31 August 2015; pp. 314–318.
13. Patel, P.; Narmawala, Z.; Thakkar, A. A Survey on Intelligent Transportation System Using Internet of Things. In *Emerging Research in Computing, Information, Communication and Applications*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 231–240.
14. Zhang, J.; Wang, F.; Wang, K.; Lin, W.; Xu, X.; Chen, C. Data-Driven Intelligent Transportation Systems: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 1624–1639, doi:10.1109/TITS.2011.2158001.
15. Qureshi, K.N.; Abdullah, A.H. A survey on intelligent transportation systems. *Middle-East J. Sci. Res.* **2013**, *15*, 629–642.
16. Yuan, Y.; Wang, F.-Y. Towards blockchain-based intelligent transportation systems. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 2663–2668.
17. Zhang, G.; Wang, Y. Machine Learning and Computer Vision-Enabled Traffic Sensing Data Analysis and Quality Enhancement. In *Data-Driven Solutions to Transportation Problems*; Elsevier: Amsterdam, The Netherlands, 2019; pp. 51–79.
18. Zhu, L.; Yu, F.R.; Wang, Y.; Ning, B.; Tang, T. Big Data Analytics in Intelligent Transportation Systems: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 383–398, doi:10.1109/TITS.2018.2815678.
19. Duleba, S.; Moslem, S. Examining Pareto optimality in analytic hierarchy process on real Data: An application in public transport service development. *Expert Syst. Appl.* **2019**, *116*, 21–30.
20. El Faouzi, N.-E.; Leung, H.; Kurian, A. Data fusion in intelligent transportation systems: Progress and challenges–A survey. *Inf. Fusion* **2011**, *12*, 4–10.
21. Bachmann, C.; Abdulhai, B.; Roorda, M.J.; Moshiri, B. A comparative assessment of multi-sensor data fusion techniques for freeway traffic speed estimation using microsimulation modeling. *Transp. Res. Part C Emerg. Technol.* **2013**, *26*, 33–48.

22. Shahrbabaki, M.R.; Safavi, A.A.; Papageorgiou, M.; Papamichail, I. A data fusion approach for real-time traffic state estimation in urban signalized links. *Transp. Res. Part C Emerg. Technol.* **2018**, *92*, 525–548.

23. Chang, T.-H.; Chen, A.Y.; Chang, C.-W; Chueh, C.-H. Traffic speed estimation through data fusion from heterogeneous sources for first response deployment. *J. Comput. Civ. Eng.* **2014**, *28*, 04014018.

24. Shen, L.; Hadi, M. Practical approach for travel time estimation from point traffic detector data. *J. Adv. Transp.* **2013**, *47*, 526–535.

25. Spring, G. Knowledge-based systems in transportation. In *Artificial Intelligence in Transportation: Information for Application, Transportation Research Circular, No. E-C113*; Transportation Research Board of the National Academies: Washington, DC, USA, 2007; pp. 7–16.

26. Lee, W.-H.; Tseng, S.-S.; Tsai, S.-H. A knowledge based real-time travel time prediction system for urban network. *Expert Syst. Appl.* **2009**, *36*, 4239–4247, doi:10.1016/j.eswa.2008.03.018.

27. Chow, A.; Dadok, V.; Dervisoglu, G.; Gomes, G.; Horowitz, R.; Kurzhanskiy, A.; Kwon, J.; Lu, X.-Y.; Muralidharan, A.; Norman, S. TOPL: Tools for operational planning of transportation networks. In Proceedings of the ASME 2008 Dynamic Systems and Control Conference, Ann Arbor, MI, USA, 20–22 October 2008; pp. 1035–1042.

28. Ben-Akiva, M.; Bierlaire, M.; Burton, D.; Koutsopoulos, H.N.; Mishalani, R. Network state estimation and prediction for real-time traffic management. *Netw. Spat. Econ.* **2001**, *1*, 293–318.

29. *DYNASMART-X Evaluation for Real-Time TMC Application: CHART Test Bed*; Available online: https://d1wqtxts1xzle7.cloudfront.net/44371326/DYNASMART-X_EVALUATION_FOR_REAL-TIME_TMC20160403-26688-1fbhpxs.pdf?1459742206=&response-content-disposition=inline%3B+filename%3DDYNASMART_X_evaluation_for_real_time_TMC.pdf&Expires=1591969100&Signature=T1g3Zq8PpabOFTvtZAjj~ptFsB2blEeGyfiGklN~FdVm8OFkmSrjsVyd7~sv~XcC9kzN~lA9t1zVGdMlDHDoOe1oOWpYjjhUUIrnfLdPP7gWbCqSfJatQEjUKrQ7-yPC-Kd3eT7FkdZRAWCv6XBQWrm5WWToLYQuiSIk~hUh-PRp3qOTDjNZvQUpKeMCmHq3gCYyex4WIBUAmgpIdXVNzcaXSjpIAgg1mWPBeBlalYWv-3VdDSBPBzYFZXWioajX1aaQFo2ATaMxEyt50ePasxB9OxVRi-UmT0wAW1rmrHkVaS6GM8JfNEnx9w0Wcm9Kk0smNClhV4HRYJKsjCQF5w__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

30. Chrobok, R.; Hafstein, S.F.; Pottmeier, A. Olsim: A new generation of traffic information systems. *Forsch. Wiss. Rechn.* **2004**, *63*, 11–25.

31. Casas, J.; Ferrer, J.L.; Garcia, D.; Perarnau, J.; Torday, A. Traffic simulation with aimsun. In *Fundamentals of Traffic Simulation*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 173–232.

32. Oh, S.; Byon, Y.-J.; Jang, K.; Yeo, H. Short-term travel-time prediction on highway: A review on model-based approach. *KSCE J. Civ. Eng.* **2018**, *22*, 298–310, doi:10.1007/s12205-017-0535-8.

33. Kumar, V.; Kumar, B.A.; Vanajakshi, L.; Subramanian, S.C. Comparison of model based and machine learning approaches for bus arrival time prediction. In Proceedings of the 93rd Annual Meeting, Washington, DC, USA, 12–16 January 2014; pp. 14–2518.

34. Hou, Y.; Edara, P. Network scale travel time prediction using deep learning. *Transp. Res. Rec.* **2018**, *2672*, 115–123.

35. Yu, B.; Wang, H.; Shan, W.; Yao, B. Prediction of bus travel time using random forests based on near neighbors. *Comput. Aided Civ. Infrastruct. Eng.* **2018**, *33*, 333–350.

36. Google. Google Transit APIs. Available online: https://developers.google.com/transit/ (accessed on 21 August 2019).

37. Schmidtke, H.R. A survey on verification strategies for intelligent transportation systems. *J. Reliab. Intell. Environ.* **2018**, *4*, 211–224, doi:10.1007/s40860-018-0070-5.

38. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-K.; Woo, W.-C. Convolutional LSTM Network: A machine learning approach for precipitation nowcasting. In Proceedings of the 28th International Conference on Neural Information Processing Systems, Montréal, QC, Canada, 7–10 December 2015; pp. 802–810.

39. Luong, T.; Pham, H.; Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1412–1421.

40. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.

41. Chorowski, J.K.; Bahdanau, D.; Serdyuk, D.; Cho, K.; Bengio, Y. Attention-based models for speech recognition. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 577–585.

42. Liang, Y.; Ke, S.; Zhang, J.; Yi, X.; Zheng, Y. GeoMAN: Multi-level Attention Networks for Geo-sensory Time Series Prediction. In Proceedings of the IJCAI, Stockholm, Sweden, 13–19 July 2018; pp. 3428–3434.

43. Zheng, G.; Mukherjee, S.; Dong, X.L.; Li, F. Opentag: Open attribute value extraction from product profiles. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 1049–1058.

44. NSW, T.F. General Transit Feed Specification (GTFS) and GTFS-Realtime (GTFS-R). Available online: https://opendata.transport.nsw.gov.au/documentation (accessed on 21 August 2019).

45. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for large-scale machine learning. In Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation, Savannah, GA, USA, 2–4 November 2016; pp. 265–283.

46. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. Available online: https://arxiv.org/pdf/1412.6980.pdf (accessed on 10 May 2020)

47. Greff, K.; Srivastava, R.K.; Koutník, J.; Steunebrink, B.R.; Schmidhuber, J. LSTM: A Search Space Odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 2222–2232, doi:10.1109/TNNLS.2016.2582924.

48. Zhang, J.; Zheng, Y.; Qi, D. Deep spatio-temporal residual networks for citywide crowd flows prediction. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 1655–1661.