Travel-Time Prediction With Deep Learning

Chaiyaphum Siripanpornchana*, Sooksan Panichpapiboon* and Pimwadee Chaovalit†
*Faculty of Information Technology
King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand
E-mail: 57606009@kmitl.ac.th, sooksan@alumni.cmu.edu
†Information Communication and Computing Research Unit
National Electronics and Computer Technology Center, Pathumthani 12120, Thailand
E-mail: pimwadee.chaovalit@nectec.or.th

***Abstract*—Travel time prediction is a challenging problem in Intelligent Transportation Systems (ITS). Accurate travel time information helps motorists plan their routes more wisely. This, in turn, alleviates traffic congestion and improves operation efficiency. A number of travel time prediction techniques exist; however, most of them are based on shallow learning architectures. In contrast to deep learning architectures, shallow learning architectures are lack of features-learning capability. In this paper, we propose an effective travel time prediction technique based on a concept of Deep Belief Networks (DBN). In our method, a stack of Restricted Boltzmann Machines (RBM) is used to automatically learn generic traffic features in an unsupervised fashion, and then a sigmoid regression is used to predict travel time in a supervised fashion. The experimental results, based on real traffic data, show that the proposed method can achieve great performance in terms of prediction accuracy.**

***Index Terms*—travel-time prediction, deep learning, deep belief networks (DBN)**

I. Introduction

Travel time prediction is a challenging problem in Intelligent Transportation Systems (ITS) [1]. Being able to estimate the time required to traverse a specific route is extremely useful. Accurate travel time information helps motorists plan their routes more wisely. This, in turn, alleviates traffic congestion and improves operation efficiency. Travel time is usually predicted from historical and real-time traffic data. These data are collected from fixed sensors such as inductive loop detectors and surveillance cameras or from mobile sensors such as vehicles with on-board Global Positioning System (GPS).

While a number of travel time prediction techniques exist, most of them are based on shallow learning architectures. In contrast to deep learning architectures, shallow learning architectures are lack of features-learning capability. In this paper, we propose a new travel time prediction technique based on a concept of Deep Belief Networks (DBN) [2]. One advantage of a deep learning method is that it works well with a large amount of traffic data. In our method, a stack of Restricted Boltzmann Machines (RBM) is used to automatically learn generic traffic features in an unsupervised fashion, and then a sigmoid regression is used to predict travel time in a supervised fashion. Unlike most travel time prediction methods which require the traffic data for each road link to be trained separately, our method can collectively train the traffic data on the entire road network all at once. The experimental results, based on real traffic data, confirm that the proposed method can achieve great performance in terms of prediction accuracy.

The rest of this paper is organized as follows. In Section II, we briefly discuss work related to travel time prediction. In Section III, we introduce a deep learning model for travel time prediction. The details of our experiments and results are described in Section IV. Finally, we conclude this paper in Section V.

II. Related Work

One of the most widely used techniques in travel time prediction is time-series analysis. It relies on the assumption that future values of data depend on the past values and random noise. The commonly used time-series models include Auto-Regressive Moving Average (ARMA) [3], generalized autoregressive conditional heteroscedasticity model [4], and seasonal ARIMA model [5].

Artificial neural network is also a popular technique in travel time prediction. Typically, traffic data such as speed, flow, and occupancy are used to train the neural networks [6]–[9]. A support vector regression (SVR) approach is used for travel time prediction in [10]. There are also studies on travel time prediction which rely on traffic models. Most of them are simulation-based studies. Examples of these studies can be found in [11], [12]. Other techniques such as Kalman filtering [13], linear regression with time-varying coefficients [14], and common least squares approach [15] are also considered.

Although a number of travel time prediction methods have been developed, most of them use shallow learning architectures. In other words, they do not have features-learning capability, which allow an algorithm to select optimal features automatically. Moreover, most of the existing approaches require the traffic data for each road link to be trained separately. In this work, we propose a new travel time prediction method which takes advantage of the deep learning architecture, making it more effective and more flexible.

III. Deep Learning Architecture

A. Deep Belief Network (DBN)

DBN is one of the most common and effective deep learning models. It has been applied with success in many applications,

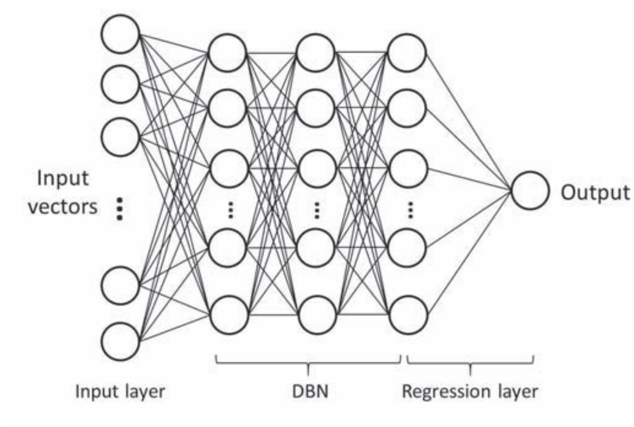

Fig. 1: Structure of the travel time prediction model.

including Intelligent Transportation Systems [16], [17]. DBN can be described as a stack of Restricted Boltzmann Machines (RBM). RBM is a two-layer network model, consisting of visible units and hidden units [18]. The visible units are for representing observable data while the hidden units are for capturing dependencies between observed variables.

DBN exploits both unsupervised learning and supervised learning. In DBN, a large amount of unlabeled data are first used for learning the features representation in an unsupervised fashion. Later, labeled data are used for fine-tuning the model in a supervised fashion. DBN is great at dealing with complicated relations in the data, and it is capable of learning features automatically. The features are learned on a layer-by-layer basis, where higher-layer features are learned from the previous layers. Higher-layer features are believed to be richer and better represent the information contained in the inputs.

DBN consists of two phases, namely the *pre-training* phase and the *fine-tuning* phase. In the pre-training phase, the weight of a RBM is calculated and updated via a contrastive divergence method and Gibbs-sampling [19]. The outputs of each RBM are then fed as inputs to the next RBMs. This process is repeated until all the RBMs are pre-trained. After the pre-training phase, in order to get better performance, the weights can be fine-tuned with labeled data through back-propagation.

B. Travel Time Prediction Model

Our travel time prediction model consists of three main parts as shown in Fig. 1. The first part is the input layer, which takes inputs from a one-dimensional vector of route travel time observations. The input vector can be described as a vector of historical travel time on an observation route. Particularly, the input vector for predicting the travel time on a route at time T_{t+d} will be a vector of the form $< T_t, T_{t-d}, T_{t-2d}, \ldots, T_{t-(k-1)d} >$, where T_t is the travel time at the current time t, d is the length of the prediction interval (e.g., 15 minutes, 30 minutes, etc.), and k is the number of past observations on a route.

The second part of the prediction model is the DBN layer. The DBN is used for automatic features-learning. Basically, the DBN will determine appropriate features from the vectors of inputs. This is the main advantage of using a deep learning architecture, in a sense that we do not have to manually select features. The model will take care of extracting important features from the inputs by itself. Once the DBN determines the features to be used, these will be fed to the third part of the prediction model, which is the sigmoid regression layer. Finally, the prediction model will return the predicted travel time on a specific route at time $t + d$ as an output.

IV. Experiments

A. Data Description

In this paper, we use real travel time data collected by the Caltrans Performance Measurement System (PeMS) [20], which is one of the most widely used data set in traffic analysis. The PeMS data set consists of both near real-time data and historical data, especially on urban freeways. The traffic data are collected from over 42000 inductive loop detectors in real-time. The collected data are then aggregated every five minutes into a travel time on each route.

In our study, we consider the travel time data on over five-hundred different routes of the 3514-km long network, which covers the entire State of California. This data set contains the travel time on each route during all the twelve months of 2015. The first nine months of data are used as the training set while the last three months of data are used as the testing set. Our prediction model is basically trained with the data on over five-hundred different routes in the network. In the testing set, we examine the travel time on eleven routes, ranging from 3-km to 131-km in length. The two directions of the freeways are considered separately.

Fig. 2 illustrates the weekly travel time patterns over the period of four weeks in November, 2015. It can be observed that the travel time on the weekdays exhibits a recurring pattern as expected. Basically, the travel time starts to increase in the morning rush hours, and it continues on this trend towards the evening hours. It then drops in the night hours.

B. Specific Model Structures and Parameters

Considering the proposed travel time prediction model shown in Fig. 1, there are choices of parameters that we need to specify. They are described here.

1) Inputs: For the input layer, we use the historical travel-time data as inputs to the model. The inputs are the travel-time observations in the past k time intervals on a route. A time interval is the length of time that we need to make a prediction on (e.g., 15-minute, 30-minute, etc.) We have experimented with many values of k, but setting k equal to 12 yields the best results. Thus, the number of units in the input layer will be equal to 12. The DBN layer in the prediction model is set up to learn the temporal relation between these historical travel time data.

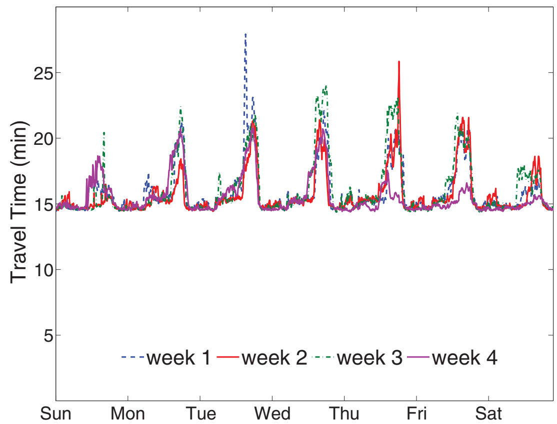

Fig. 2: Weekly travel time pattern on Route SR4-E from Hillcrest Ave to I-680 (29.13-km long). The travel time data shown are in the period between November 1 and November 28, 2015.

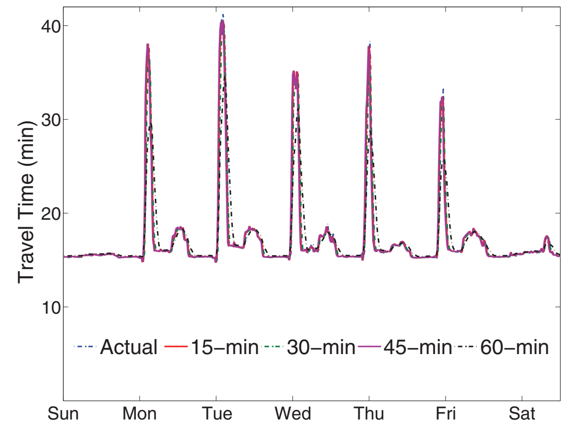

Fig. 3: A comparison between the actual travel time values and the predicted values on Route SR4-E from Hillcrest Ave to I-680 (29.13-km long). The travel time data shown are in the period between October 4 and October 10, 2015.

2) Structure of DBN: For the DBN layer, we need to specify the number of hidden layers and the number of hidden nodes in each layer. We have experimented with many combinations of number of hidden layers and number of nodes in each layer. After performing grid search runs, the best structure consists of three hidden layers, and the number of hidden nodes in each layer are 140, 200, and 10, respectively. In this work, we use an open-source deep-learning library called DL4J for the learning tasks [21].

C. Performance Evaluation

In this study, the performance of the travel time prediction method is evaluated with three metrics, which are mean absolute error (MAE), mean absolute percent error (MAPE), and root mean square error (RMSE). These metrics are defined as follows

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

$$\text{MAPE (\%)} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i}\times 100\% \tag{2}$$

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

where y_i is the actual value of observed travel time, $\hat{y}_i$ is the predicted value of travel time, and n is the number of observations.

D. Results and Discussion

The travel times on eleven routes in the testing set are used for performance evaluation. The data in the testing set are from October 1, 2015 to December 31, 2015. Table I shows the performance of our travel time prediction model. The first column lists the observation routes. The route distance varies from 3.38 km to 131.32 km. The remaining columns show the prediction results, in terms of the three metrics, for the next 15-minute, 30-minute, 45-minute and 60-minute intervals, respectively.

The proposed travel time prediction model yields a great performance. For the 15-minute travel time prediction, in terms of MAE, the prediction results are accurate to within 1 minute of the actual values on most of the routes. Similarly, in terms of MAPE, the prediction results are accurate to within 7% of the actual values on most of the routes. However, the prediction accuracy decreases as the length of the prediction interval increases. In other words, the farther we look into the future, the harder it gets to predict the travel time. Nonetheless, our prediction model can still perform quite well in the 60-minute prediction interval. Note that the maximum MAE in this case is only 7.26 minutes on a 131-km route. It is also interesting to observe from Table I that the travel time on a longer route is generally more difficult to predict.

Fig. 3 shows a comparison between the actual travel time values and the predicted values obtained from the prediction model. It can be observed that the predicted values closely follow the actual travel time values. Among all the interval lengths, the 60-minute interval seems to have the slowest response to the changes in the travel time pattern. This is expected because it is normally more difficult to predict the values in the more distant future.

TABLE I: Accuracy of the travel time prediction model

Route	15-min prediction			30-min prediction			45-min prediction			60-min prediction		
	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE
SR84-E (3.38-km)	0.29	9.39	0.97	0.37	12.41	1.06	0.39	13.61	1.04	0.41	14.43	1.04
I580-E (9.66-km)	0.45	6.12	1.23	0.58	7.93	1.39	0.64	8.97	1.43	0.69	9.82	1.45
I880-N (14.48-km)	0.79	6.42	2.24	1.07	8.87	2.67	1.15	9.55	2.79	1.28	10.87	2.83
I680-S (16.42-km)	0.84	6.59	2.15	1.12	8.93	2.54	1.20	9.48	2.64	1.29	10.36	2.70
I580-E (27.36-km)	0.92	7.55	2.00	1.29	10.81	2.42	1.45	12.22	2.54	1.57	13.11	2.61
I680-N (34.28-km)	0.94	3.96	2.09	1.35	5.78	2.82	1.59	6.80	3.16	1.92	8.41	3.47
I580-E (49.08-km)	0.91	3.00	1.38	1.34	4.44	1.88	1.67	5.50	2.34	2.28	7.52	2.97
I80-E Placer (62.44-km)	1.44	3.63	2.55	2.29	5.83	3.40	2.89	7.37	3.94	3.47	8.82	4.65
I80-E Solano (68.72-km)	1.73	7.44	3.39	2.56	11.28	4.48	3.02	13.39	4.82	3.35	14.79	5.09
US50-E (87.71-km)	2.32	3.29	3.96	3.46	4.95	5.11	4.41	6.32	5.86	5.57	7.97	7.23
US101-S (131.32-km)	3.25	3.87	5.70	4.81	5.77	7.41	6.05	7.28	8.31	7.26	8.71	9.74

V. Conclusion

In this paper, we propose a new travel time prediction model based on Deep Belief Network (DBN), which takes advantage of a deep learning architecture in acquiring the representative features from the data in an unsupervised fashion. The prediction model is tested on the PeMS data set, which is one of the most widely used traffic data. It is shown that our travel time prediction model can achieve great performance. For a practical 15-minute prediction interval, the predicted travel time values are accurate to within 1 minute of the actual values on most of the routes. In addition, our prediction model also performs quite well even if the prediction interval is large (e.g., 60-minute interval). The maximum error in such a case, in terms of MAE, is only around 7.26 minutes on a 131-km route.

Acknowledgment

This research is supported by the Thailand Graduate Institute of Science and Technology (TGIST) under grant number TG-44-22-57-024M.

References

[1] J. Zhang, F. Y. Wang, K. Wang, W. H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1624–1639, Dec 2011.

[2] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, Jul 2006.

[3] D. Billings and J. S. Yang, "Application of the ARIMA models to urban roadway travel time prediction - a case study," in *Proc. IEEE International Conference on Systems, Man and Cybernetics*, Taipei, Taiwan, Oct 2006, pp. 2529–2534.

[4] M. Yang, Y. Liu, and Z. You, "The reliability of travel time forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 1, pp. 162–171, Mar 2010.

[5] A. Guin, "Travel time prediction using a seasonal autoregressive integrated moving average time series model," in *Proc. IEEE International Conference on Intelligent Transportation Systems*, Washington, DC, USA, Feb 2006, pp. 493–498.

[6] D. Park, L. R. Rilett, and G. Han, "Spectral basis neural networks for real-time travel time forecasting," *Journal of Transportation Engineering*, vol. 125, no. 6, pp. 515–523, Nov 1999.

[7] S. Li, "Nonlinear combination of travel-time prediction model based on wavelet network," in *Proc. IEEE International Conference on Intelligent Transportation Systems*, Washington, DC, USA, Feb 2002, pp. 741–746.

[8] J. Van Lint, S. Hoogendoorn, and H. Van Zuylen, "Freeway travel time prediction with state-space neural networks: modeling state-space dynamics with recurrent neural networks," in *Proc. 81st Annual Meeting Transportation Research Board*, 2002, pp. 30–39.

[9] J. Van Lint, "Online learning solutions for freeway travel time prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 1, pp. 38–47, Mar 2008.

[10] C.-H. Wu, J.-M. Ho, and D.-T. Lee, "Travel-time prediction with support vector regression," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 276–281, Dec 2004.

[11] S. C. Calvert, J. Van Lint, and S. Hoogendoorn, "A hybrid travel time prediction framework for planned motorway roadworks," in *Proc. IEEE International Conference on Intelligent Transportation Systems*, Madeira, Portugal, Sep 2010, pp. 1770–1776.

[12] T.-Y. Hu, C.-C. Tong, T.-Y. Liao, and W.-M. Ho, "Simulation-assignment-based travel time prediction model for traffic corridors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1277–1286, Sep 2012.

[13] Y. Yuan, J. Van Lint, R. E. Wilson, V. Wageningen-Kessels, S. P. Hoogendoorn *et al.*, "Real-time lagrangian traffic state estimator for freeways," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 59–70, Mar 2012.

[14] J. Rice and E. Van Zwet, "A simple and effective method for predicting travel times on freeways," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 3, pp. 200–207, Sep 2004.

[15] J. Kwon, B. Coifman, and P. Bickel, "Day-to-day travel-time trends and travel-time prediction from loop-detector data," *Journal of the Transportation Research Board*, no. 1717, pp. 120–129, 2000.

[16] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: Deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2191–2201, Oct 2014.

[17] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, Apr 2015.

[18] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul 2006.

[19] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, Mar 2002.

[20] State of California, "Caltrans performance measurement system (PeMS)," [Online], Available: http://pems.dot.ca.gov/, Feb 2016.

[21] Deeplearning4j Development Team, "Deeplearning4j," [Online], Available: http://deeplearning4j.org/, Feb 2016.