

Short-Term Transit Decision Support System Using Multi-Task Deep Neural Networks

Fangzhou Sun, Abhishek Dubey, Chinmaya Samal
Department of EECS, Vanderbilt University
Nashville, TN, USA
{fangzhou.sun, abhishek.dubey, chinmaya.samal.1}@vanderbilt.edu

Hiba Baroud
Department of CEE
Vanderbilt University
Nashville, TN, USA
hiba.baroud@vanderbilt.edu

Chetan S. Kulkarni
SGT. Inc, NASA Ames Research Center
Moffett Field, CA, USA
chetan.s.kulkarni@nasa.gov

Abstract—Unpredictability is one of the top reasons that prevent people from using public transportation. To improve the on-time performance of transit systems, prior work focuses on updating schedule periodically in the long-term and providing arrival delay prediction in real-time. But when no real-time transit and traffic feed is available (e.g., one day ahead), there is a lack of effective contextual prediction mechanism that can give alerts of possible delay to commuters. In this paper, we propose a generic tool-chain that takes standard General Transit Feed Specification (GTFS) transit feeds and contextual information (recurring delay patterns before and after big events in the city and the contextual information such as scheduled events and forecasted weather conditions) as inputs and provides service alerts as output. Particularly, we utilize shared route segment networks and multi-task deep neural networks to solve the data sparsity and generalization issues. Experimental evaluation shows that the proposed toolchain is effective at predicting severe delay with a relatively high recall of 76% and F1 score of 55%.

Keywords—public transportation; delay prediction; neural network; deep learning; multi-task learning;

I. INTRODUCTION

Emerging Trends. Traffic congestion is one of the major quality of life concerns in urban areas, and it disproportionately affects the less affluent who live far from city cores and employment opportunities. City planners in major metropolitan areas are considering a potpourri of approaches to manage congestion, which include economic incentives such as dynamic toll pricing, high-occupancy vehicle lane (HOV) lanes, and better infrastructure for alternative modes of transportation, such as biking, among others. However, the pillar of the congestion mitigation mechanisms remains the public transit system as it provides a high capacity low-impact mode of transportation. Yet, as seen by several studies around the country, public transit is not the preferred mode of travel in many cities [1].

Besides lack of cross-town routes and low service reliability, unpredictability is one of the top reasons that prevent people from using public transportation [2], [3]¹. The variance of travel time on a bus route can very high due to a number of reasons, including passenger load and unload times, traffic congestions, weather conditions, events such as football and hockey games. For example, Figure 1(a) shows a bus route

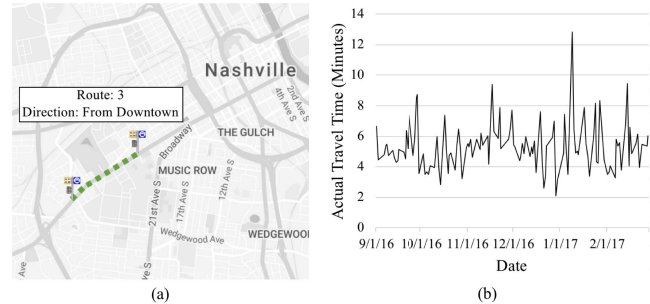


Fig. 1. (a) A route segment on bus route 3 leaving downtown; (b) The variance of actual travel time on a bus route segment is very high in time period between Sept. 1, 2016 and Feb. 28, 2017.

segment on route 3 departing from downtown; Figure 1(b) illustrates the travel times of a bus trip (departs at the same time of day) on the same bus route segment for six months. Since commuters want to arrive on time, they have to compromise with the unreliable transit service and accommodate some extra time in their schedule, which causes inconvenience and dissatisfaction to bus passengers.

A number of strategies have been used to improve transit vehicle performance. For example, controlled time points [5] have been used to distribute the available slack time across the route. Periodic schedule update [6], [7], [8] and dedicated bus travel lanes [9] have also gained attention. The availability of real-time data collected from Automated Vehicle Locator's (AVL) has helped significantly with such long-term planning and schedule updates [10], [11], [12], [13]. As timetable scheduling problem is recognized to be an NP-hard problem [14], [15], researchers have implemented heuristic search algorithms to solve the problem. While heuristics-driven schedule update focuses on the long-term planning, the availability of AVL data has also enabled real-time delay prediction. These AVL based methods can be roughly classified into statistical [16], [17], [18], [19], Kalman filter models [20], [21], [22], [23], and machine learning models [22], [24], [18], [25], [26].

Research Gap and Challenges. Commuters typically rely on timetables that are scheduled according to long-term patterns, and delay status updated in real time to plan transit trips. However, for short-term transit scheduling (e.g., one day before the travel day) when there is no real-time data available yet, commuters have no clue how to choose a proper route and departure time, and it's even more challenging for those who

¹In 2015, a mass transit survey from Nashville, USA revealed that people who regularly use the transit system focus mostly on creating a system in which buses arrive on time [4].

do not take public transportation on regular basis. It has been observed that there are recurrent traffic congestion [27] and transit delay (see Section II-A) patterns associated with events happening in the city. For example, for a given day with a big football game, the bus routes that bypass the area around the stadium are more likely to have a more significant delay than usual. Such contextual information (e.g., scheduled events and weather forecast conditions) could be utilized to get a better estimation of the expected delay.

However, as illustrated in Figure 2, most of the prior research work is focused on long-term delay pattern analytics [10], [11], [12], [13] and real-time delay prediction [26], [22], [22], [23], [28]. Long-term analytics provide statistical decision supports such as the mean and confidence interval of route segment delay, which are useful for city planners and metro transportation authority (MTA) engineers to gain a deep insight into the actual transit performance for scheduling and planning. On the other hand, real-time delay prediction utilizes real-time data such as trip updates and vehicle locations to inform engineers and commuters on the expected arrival time at bus stops. However, there are still gaps in effective prediction mechanisms that work in the short-term phase (i.e., hours or days ahead of the scheduled travel time when real-time data is unavailable) to help commuters make decisions by informing of possible severe delay.

Developing an effective short-term prediction for transit systems is currently an open problem to solve. Firstly, route segments typically have different delay patterns and are affected by the contextual factors in varying degrees (see Section II-A and Figure 3). Technically, multiple historical models can be built for different contextual features. However, since there are so many event types and their impacts on transit delay interact with each other and have varying spatiotemporal characteristics, advanced prediction models are needed to effectively integrate all contextual features. Additionally, due to the limited budget and computation resources, it may not be feasible to create and train a single prediction model for each separate segment in practice because of the high computation time. Furthermore, the transit frequency in mid-sized cities like Nashville is relatively low compared to the large metropolitan areas. Therefore, there is never enough data to train the prediction models and results in data sparsity issues. Also, data samples with known contextual information are rarer compared with other samples². The biased datasets will produce challenges if we try to use regression or random forest based approaches.

Solution Approach and Contributions. In this paper, a short-term transit decision support system is being proposed that predicts severe delay days ahead to help users to schedule transit plans. Compared to providing just static schedules or historical patterns, the context-aware short-term delay prediction model can identify the severe delay that does not follow normal patterns days ahead of time and help commuters choose optimal routes, which gives them more confidence when choosing the public transportation. This system is being currently integrated into our transit decision support system called transit hub [29].

The key contributions are as follows:

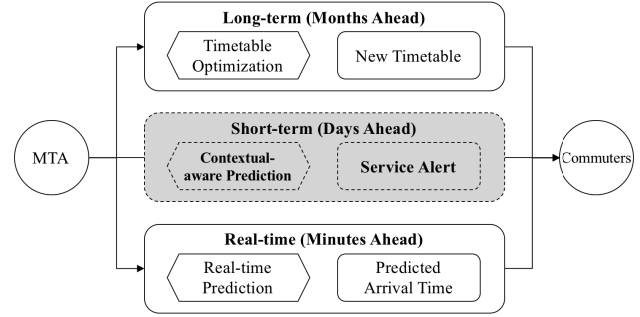


Fig. 2. MTA provides transit schedule and status update to commuters in three terms (i.e., long-term, short-term and real-time) to help them plan transit trips. An effective short-term prediction mechanism without using real-time data (e.g., transit, traffic, etc.) is currently missing.

- A generic tool-chain that takes transit feed (in standard and real-time GTFS format), forecasted weather condition, and time as input, is developed to provide expected delays and service alerts as output for short-term.
- The shared route segment networks, which are proposed in a previous work [29] for real-time delay prediction, are employed in this work as a data augmentation mechanism to solve data sparsity issues.
- A multi-task deep neural network architecture is presented that consumes contextual information in the augmented datasets and makes delay predictions for nearby segments in a bounding box all at once. A threshold-based mechanism is utilized to produce service alerts.
- Compared with single networks, the proposed multi-task learning architecture not only takes a shorter time to train but also reduces the risk of over-fitting to the limited training data. Utilizing the contextual event and weather features improves the performance of recall by 28% and F1 score by 13%.

Paper Organization. Section II provides a motivating example that illustrates the delay patterns during sports games, and discusses the key research challenges; Section IV introduces the datasets used in the study and presents a multi-task neural network architecture and details; Section V evaluates the performance of our system; Section III compares our solution with related work in bus delay prediction; Section VI gives concluding remarks and future work.

II. MOTIVATING EXAMPLE AND CHALLENGES

This section first provides a motivating example that illustrates the impact of big sports games on bus delay, and then discusses the key research challenges associated with predicting transit on-time performance using such scheduled contextual information without real-time data.

A. Motivating Example

To unveil the delay patterns associated with events, an analytics study using real transit and sports game data is conducted. The days between Sept. 1, 2016 and Jan. 1, 2017 are selected as the study period and the start time, end time, and attendance of eight football games occurred in the period are collected manually. We divide the time period before football games into four one-hour time windows $[-4, -3]$, $[-3,$

²There are a limited number of football and hockey games for example.

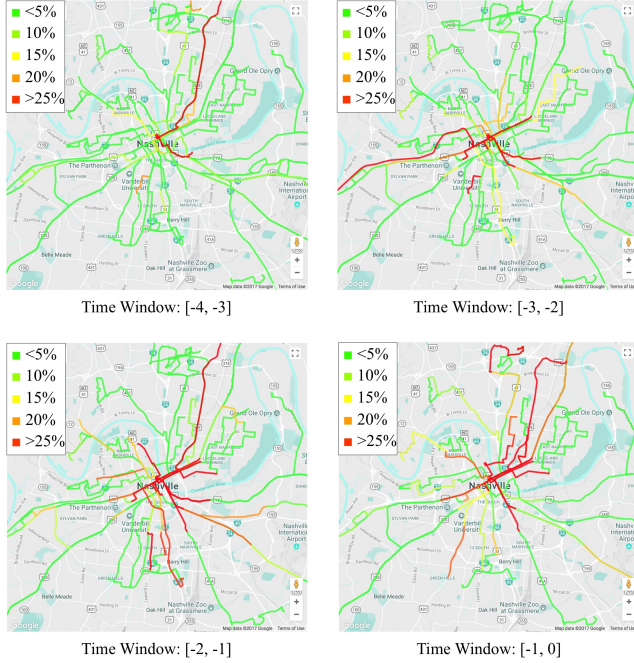


Fig. 3. The impact of football games on travel delay of bus route segments in four one-hour time windows before 8 football games: (1) from 4 to 3 hours, (2) from 3 to 2 hours, (3) from 2 to 1 hour, (4) within 1 hour. The green colors are the baselines (i.e., average delay of bus route segments on non-game days). Other colors show the difference of average delay on games days compared to the baselines.

-2], [-2, -1], [-1, 0]) and compare average bus delay on bus route segments between game days and non-game days using the following equation:

$$D_{PI} = \max\left(\frac{TT_{GD} - TT_S}{TT_S}\right) - \max\left(\frac{TT_{NGD} - TT_S}{TT_S}\right), 0 \quad (1)$$

where D_{PI} = delay impact, TT_{GD} = actual travel time on a game day, TT_S = scheduled travel time, TT_{NGD} = Actual travel time on a non-game day.

The results are visualized using heatmaps (see Figure 3). Generally, there are two patterns: (1) route segments have more delay as the time is closer to the game start time, (2) route segments that are closer to the stadium have more delay.

B. Research Challenges

The number of available historical data samples for each transit route and segment depends on the service frequency. However, compared to the large metropolians, mid-sized cities like Nashville suffer from data sparsity issue since the frequency of bus services is relatively low. Furthermore, the availability of event information is also limited and constrained by the manual data collection. This limitation of contextual information (games for example) will result in biased training data, which makes the data sparsity issue worse.

Machine learning methods have demonstrated superior performance in the transit domain [22], [24], [18], [25], [26]. However, the machine learning models in those works were trained separately for a particular metric by training a single

model or ensemble models, which may suffer from insufficient training data in reality. Training deep learning models will be more challenging since they have multiple hidden layers and there are much more parameters to optimize. This often results in overfitting issues and make the models difficult to generalize to new data.

To solve the challenges, we utilize a data augmentation structure called shared route segment network [29], and adopt the idea of multi-task learning to develop multi-task deep neural networks. Neural networks, which have been studied in many related work [18], [24], [26], are effective in prediction and have the potential to be trained online. Compared with single models, our proposed multi-task neural networks can not only produce more data for each task (i.e., predicting travel time and delay for a route segment) and are faster to train, but also reduces the overfitting issues for individual tasks. The details can be found in Section IV.

III. RELATED WORK

This section compares our system with related work on transit travel time and delay prediction. The work can be classified into two categories: (1) models utilizing real-time data feeds, (2) models not relying on real-time data. The models that do not rely on real-time data use average delay from historical data as the estimation for future. These types of models are often constructed for comparison purposes. For example, Jeong et al. [18] developed a basic average model and found that the basic average model was outperformed by regression models and artificial neural network (ANN) models for bus arrival time prediction. The reason is that the basic average models does not account for any predictors and only use historical data and perform simple average analysis, the model does not reflect real-time conditions and is limited by the consistency of route delay patterns.

The development of automatic vehicle location (AVL) technology enables accurate prediction of transit travel time in real-time, which is critical for transit planning and user notification. A number of researchers have conducted studies that utilize real-time transit as well as historical data.

Statistical Models. Weigang et al. [16] presented a model to estimate bus arrival time at bus stops using the real-time GTFS data. Their model contains two sub-algorithms to determine the bus speed using the historical average speed and the real-time speed information from GPS. Their main algorithm utilizes the calculated real-time speed to predict the arrival time. Sun et al. [19] proposed a prediction algorithm that combines real-time GPS data and average travel speeds of route segments.

Kalman Filter Models. Kalman filters have been used widely for bus delay prediction because of their ability to filter noise and continuously estimate and update actual states from observed real-time data. Chien et al. [20] presented a dynamic travel time prediction model that used real-time and historical data collected on the New York State Thruway (NYST). Yang et al. [30] developed a discrete-time Kalman filter model to predict travel time using collected real-time Global Positioning System (GPS) data. Bai et al. [22] proposed a dynamic travel time prediction model that employed support vector machines to provide a base time estimate and a Kalman filter to adjust the prediction using the most recent bus trips on multiple routes.

Machine Learning Models. Machine learning models such as artificial neural network (ANN) [24], [18], [26] and support vector machines (SVM) [31], [25], [26], [22] have been widely used for transit travel time and delay prediction. Kumar et al. [32] compared machine learning models with other approaches for bus arrival time prediction and found that with large datasets ANN is better than Kalman filtering. Jeong et al. [33] developed an ANN model for bus arrival time prediction using Automatic Vehicle Location (AVL) data. Mazloumi et al. [34] used real-time traffic flow data to develop ANN models to predict bus travel times. Yu et al [26] proposed a machine learning model that used bus running times of multiple routes for predicting arrival times of each bus route and proposed bus arrival time prediction models that include Support Vector Machine (SVM), Artificial Neural Network (ANN), k-nearest neighbors algorithm (k-NN) and linear regression (LR).

Comparison with Our Work. Existing work mainly focuses on building models that rely heavily on the availability of real-time data feeds, such as traffic conditions and transit location status. However, these models only work when real-time or near real-time data is available. Models not relying on real-time data learn the travel time and delay patterns from historical data, but since the operation of transit vehicles are affected by various factors, these models typically have less accuracy and low prediction granularity. In contrast, our model integrates contextual information, especially those that have large impact on generating severe delays. Also, to the best of our knowledge, we are the first to apply multi-task learning (MTL) on transit short-term delay prediction, which greatly reduces the possibility of overfitting to the limited historical dataset and improve the generalization ability in reality.

IV. OUR APPROACH

This section presents a short-term arrival delay prediction model that utilizes multi-task neural network architecture to make accurate and context-aware delay estimations. The overall workflow of the system is shown in Figure 4. We first introduce the datasets used in the study. As discussed in Section II-B, the data sparsity issue on relatively low frequent bus routes is solved by combining shared route segment network [29] and multi-task neural networks. The architecture also improves the training and prediction efficiency and reduces the risk of overfitting. The system can also notify transit riders of possible severe delay via service alerts if the predicted delay is beyond a set of threshold. All symbols used in the paper are listed in Table I

A. Datasets

Nashville Metro Transportation Authority (MTA) provides access the static and real-time bus data of Nashville. Along with bus data, the information of football games, hockey games, and weather conditions are also integrated into the system. The datasets are listed in Table II. The details are described as follows:

- **Static Bus:** This dataset defines the static bus information: time schedules, routes, trips, stops, etc. The data is in General Transit Feed Specification (GTFS) static format [35], which is a common format for public transportation schedules and associated geographic information.

TABLE I. SYMBOLS USED IN THE PAPER

D_{PI}	the delay impact of events on route segments compared with normal days
TT_{GD}	actual travel time on a segment on a game day
TT_S	scheduled travel time on a segment
TT_{NGD}	actual travel time on a segment on a non-game day
t_{point}	the estimated arrival time at a point on a segment
t_i	recorded timestamp at record index i
d_{point}	a bus's distance from the current point to the route starting point along the route path
d_i	a bus's distance from the location of record index i to the route starting point along the route path

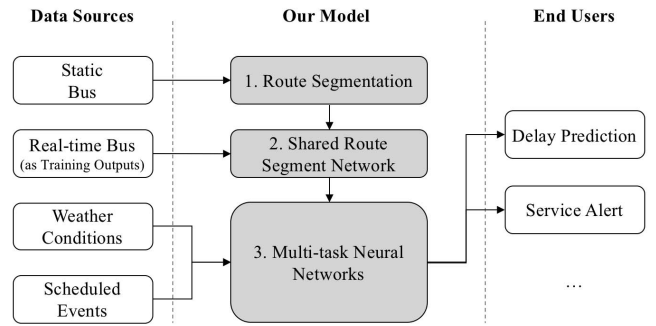


Fig. 4. Overall workflow of the short term delay prediction toolchain: (1) route segmentation, (2) historical data pre-processing, (3) training multi-task neural networks.

- **Real-time Bus:** The dataset provides historical transit data updates, which are collected and stored in one-minute interval. The data is in GTFS Realtime format [36], including bus locations, service alerts, and trip updates.
- **Weather Conditions:** The dataset stores historical weather conditions with a granularity of five minutes.
- **Sports Games:** The dataset contains game and schedule statistics that are manually collected from online resources. The features include start time, end time, and attendance.

The following assumptions are made when using the datasets: (1) the weather condition data that we collected in real-time is used as forecasted data; (2) we assume the game information is available days before the games start.

B. Data Augmentation: Shared Route Segments

Route Segmentation. Studies have shown that when predicting the travel time for a specific bus route, the travel time data from multiple routes that share the same road segments can be integrated to make more accurate predictions [26], [22], [37], [38]. In our previous work [29], an efficient algorithm is discussed that generates shared bus route segments from standard GTFS datasets. The basic idea is that the algorithm divides all bus routes in a transit system into shared segments according to how roads are shared by multiple routes. The algorithm is implemented as a data augmentation method to

TABLE II. REAL-TIME AND STATIC DATASETS COLLECTED IN THE SYSTEM.

	Sports Game	Bus Schedules	Real-time Transit	Weather
Format	JSON	Static GTFS	Real-time GTFS	JSON
Source	Web	Nashville MTA	Nashville MTA	Dark Sky
Updating Interval	Manually	Every public release	Every minute	Every 5 minutes
Size	28 Games	134 MB	723 GB	82.4 MB
Date Range	10/2016 - 12/2016	09/2015 - present	02/2016 - present	03/2016 - present

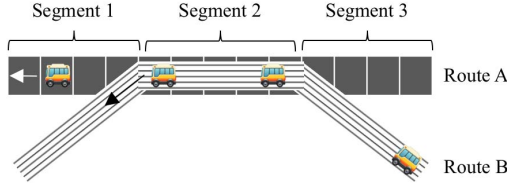


Fig. 5. Shared route segment network generated from the GTFS dataset. Route segmentation is conducted to the transit routes according to how road segments are shared by bus routes.

conduct route segmentations and get more data samples for each segment. Each route segment will be used as a minimum unit for training and prediction. As illustrated in Figure 5, all 57 bus routes (as Oct. 2016) in Nashville are divided into 4638 shared route segments.

Pre-processing Real-time GTFS Locations. Even though devices like automatic passenger counter (APC) can provide accurate arrival and departure times at bus stops and the recorded times can be used to calculate historical travel time at road segments, such data is not available in many cities. To make the short-term delay prediction toolchain more generic, the standard GTFS feeds are considered, which contain real-time trip updates, vehicle locations, and service alerts. The vehicle locations are updated from seconds to minutes. We use the following equation to calculate the bus arrival time at the start and end points of a segment:

$$t_{point} = t_{i-1} + (t_i - t_{i-1}) \frac{d_{point} - d_{i-1}}{d_i - d_{i-1}} \quad (2)$$

where t_{point} denotes the estimated arrival time at the start or end point of a segment, d_i is the bus's distance from the current location to route start location along the route path at time t_i , and d_{point} is the distance from point's location to route start location along the route path.

C. Delay Prediction by Developing Multi-task Neural Network

1) *Feature Engineering:* The features of different data sources are represented differently using numeric or one hot encoding according to their attributes as follows:

- **Event Features:** The information of scheduled events is represented by one-hot vectors. The dimensionality of the vectors is the size of effective time windows before and after events plus a numerical class for attendance and an

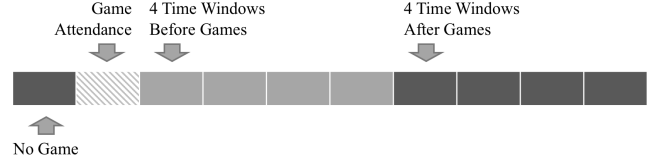


Fig. 6. The one hot encoded feature vector for football games.

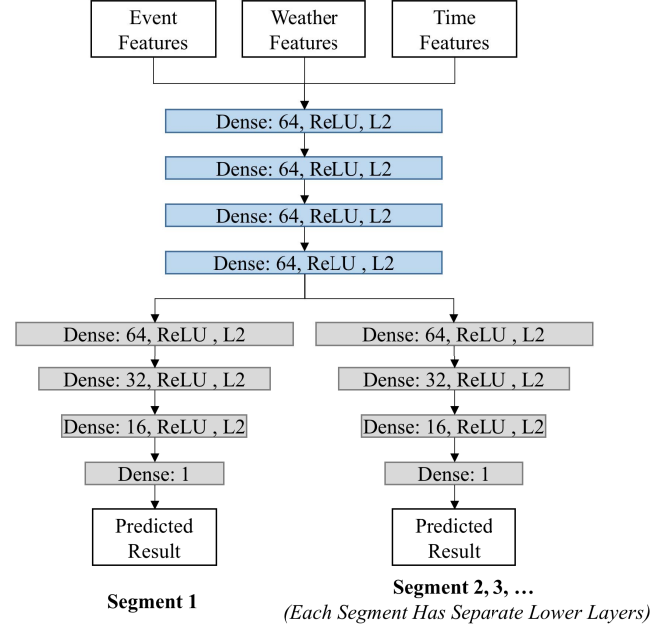


Fig. 7. Proposed Multi-task Neural Network. Blue blocks are shared layers and gray blocks are independent layers for different segments.

additional class for no event. The effective time windows when events have impact on bus delay varies on different event types. For example, from the motivation study it was found that on average the impact of football games on bus delay starts as early as 4 hours before and as late as 4 hours after games, so the length of the feature vectors for football games is ten (ie, no football game, attendance, and eight time windows). An example is illustrate in Figure 6.

- **Weather Features:** The forecasted weather conditions contains seven features: temperature, nearest storm distance, humidity, ozone, pressure, wind speed and visibility. Each weather condition sample is converted into a seven-dimensional vector of numerical features.
- **Time Features:** Time features contains two classes: time of day and day of week. The 24 hours in a day is divided into 48 half hours and time of day is represented using one hot encoding of 48 classes. Similarly, day of week is represented using a feature vector of 7 classes.

2) *Multi-task Neural Network:* Deep learning techniques have gained great success in various fields, such as neutral language processing (NLP), image processing, information retrieval (IR), among others. Researchers start to develop deep learning techniques for transportation [39], [40], [41].

Compared with existing studies which focus on real-time delay prediction and rely on real-time data feeds (e.g., transit and traffic), our model only assumes the availability of forecasted weather conditions and information of scheduled events that have proved to have significant impact on transit delay.

Multi-task learning (MTL) share representations between related tasks. By leveraging the domain-specific information contained in the training signals of related tasks, MTL improves the generalization ability on original tasks [42]. In the transit domain, the travel delay on road segments in a nearby area is usually impacted by the same events at the same time and show similar patterns. Therefore, we aggregate the original tasks of predicting delay for nearby segments together in a multi-task learning architecture. We developed multi-task deep neural networks to get more data for training since they leverage supervised data from multiple nearby road segments. Furthermore, the use of multi-task networks can also reduce overfitting to specific tasks and better generalizes to new data [43].

The architecture of the proposed multi-task neural network is shown in Figure 7. We apply the approach of hard parameter sharing in the neural networks. Generally, it consists of upper layers that connect directly to input feature vectors and are shared across different segments, and lower layers that are specific to different segments. When training the models, Adam algorithm [44] is used for optimization and mean square errors are used as loss functions. The architecture greatly reduces the risk of overfitting and it is generally faster to train and can predict for multiple segments at the same time.

3) *Service Alert Generation*: The predicted delay can be utilized to decide whether a service alert can be generated for each route. Our system collects the historical travel times for each segment and sends out a service alert if the predicted delay is larger than 90th percentile in the data.

V. EVALUATION

In this section, we evaluate the proposed model using two experiments by (1) comparing the multi-task neural networks with single models, and (2) comparing different feature vectors. Keras Python deep learning library with TensorFlow backend is used in the implementation [45].

A. Scenarios

The experiment scenario is illustrated in Figure 8. Between Oct. 1, 2016 and Jan. 1, 2017, 7 NFL football games held at the Nissan Stadium and 19 NHL hockey games at the Bridgestone Arena in Nashville. We selected the bounding box between coordinates of 36.175106, -86.760105 and 36.161903, -86.773335. Real-world bus, event and weather data within the time period and the bounding box is used in the experiments. The data between Oct. 1, 2016 and Dec. 11, 2016 is used for training. The data between Dec. 12, 2016 and Jan. 1, 2017 when there is at least an event happened is used for validation. Particularly, the following decisions are used to mark if a generated service alert is positive or not: (1) an output is considered to be positive if the delay is more than 90th percentile in the historical (training) dataset; (2) otherwise the output is negative. Recall performance, which indicates how many relevant items are selected in a classification task, is

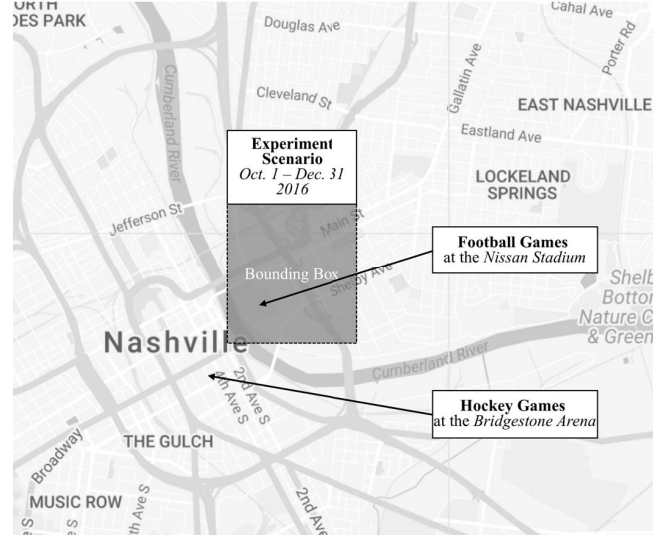


Fig. 8. Experiment scenario. The selected bounding box is close to the Nissan Stadium where football games play and the Bridgestone Arena that hockey games play.

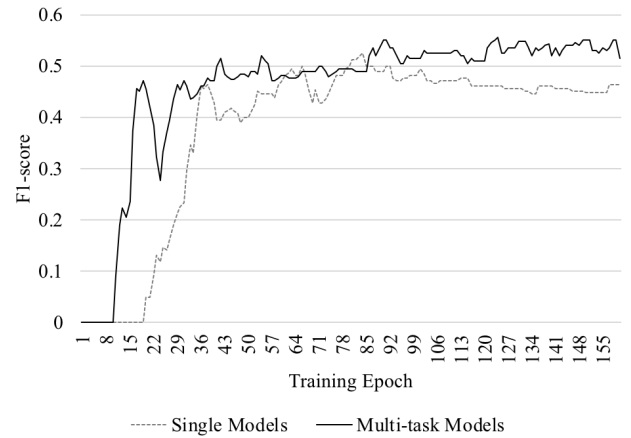


Fig. 9. The F1 score between single models and multi-task models

the key metric for evaluating our models since the metric was selected for this study to output notifies users to avoid severe delays as much as possible. The models are also evaluated using F1 score [46], which can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0.

B. Experiment 1: Comparing single models and multi-task learning models

We assume the multi-task learning models leverage supervised data by getting more training data for each segment and reduce overfitting for better generalization ability. To validate the assumption, we build the same neural network layers architecture for each segment and uses the same training and validating datasets to evaluate the single models. The difference is that the upper layers of the single models don't

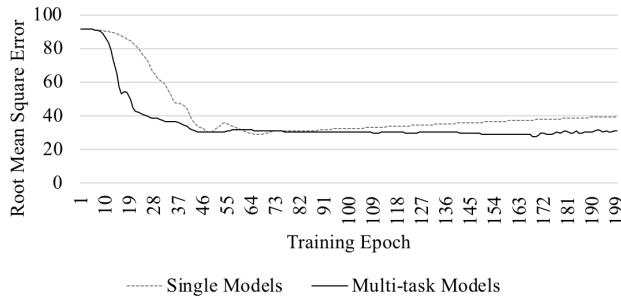


Fig. 10. The mean square error between single models and multi-task models

share parameters anymore and the parameters can only be optimized using data belong to the individual segments.

The root mean square error of the two models during training epochs are illustrated in Figure 10. The multi-task models are trained faster than single models. After 70 epochs, the single models' performance on the validating dataset becomes worse, which means it starts to overfit the training dataset. On the contrary, our multi-task neural network doesn't have a significant overfitting problem. The F1 score of the multi-task neural network is also higher than the single models (see Figure 9).

C. Experiment 2: Comparing different feature vectors using multi-task deep neural networks

In the second experiment, we compare the performance of the proposed multi-task deep neural networks using different feature vectors: (1) Time feature vectors: [day of week, time of day], (2) Contextual feature vectors: [football game time window, football game attendance, hockey game time window, weather conditions, day of week, time of day]. The recall and F1 score are calculated using the validation dataset. As shown in Figure 11 and Figure 12, compared with the time feature vectors, the contextual feature vector gets both higher recall (about 0.76) and F1 score (about 0.54), which means the model predicts more relevant (severe) delays and is more effective to warn commuters of real possible delays.

VI. CONCLUSION AND FUTURE WORK

In this paper, a generic tool-chain is proposed that takes transit feed (in standard and real-time GTFS format), forecasted weather condition, and time as input, and provide service alerts and expected delays as output for short-term. Compared with providing just static schedules or historical patterns, contextual-aware short-term delay prediction could identify the severe delay that does not follow normal patterns days ahead of time and help commuters to avoid the delays and choose optimal routes, which gives them more confidence to choose the public transportation.

Future Work. Even though the proposed model is evaluated using transit data, the multi-task learning architecture is extensible for other prediction tasks (e.g., traffic congestion prediction, travel demand prediction), as long as the tasks have some kind of spatiotemporal relations. In addition, the

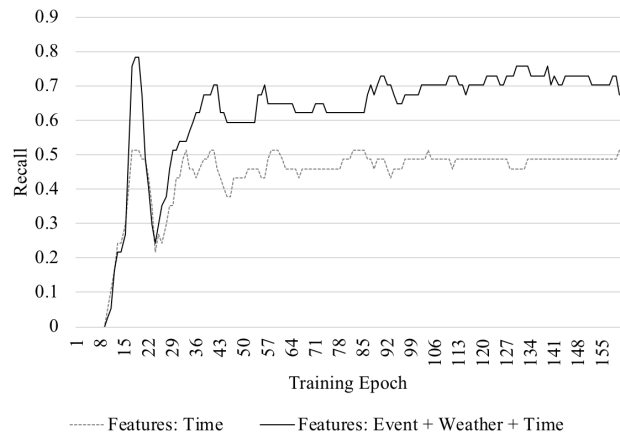


Fig. 11. The recall using time feature vectors vs. using contextual feature vectors

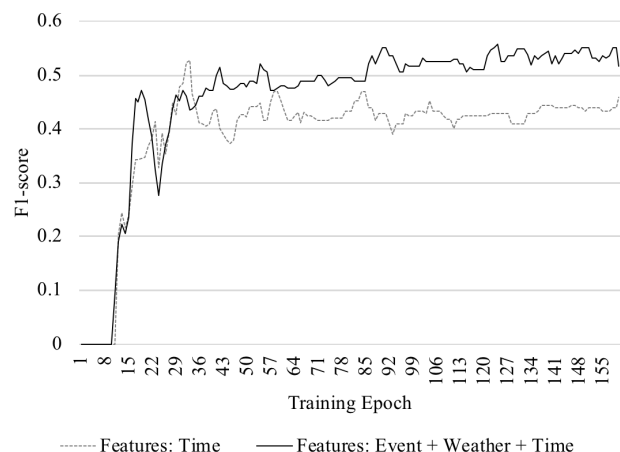


Fig. 12. The F1 score using time feature vectors vs. using contextual feature vectors

contextual information currently considered in the paper only includes sports games and weather. Parsing data from online social networks like Twitter will result in a much richer dataset, and will probably improve the short-term prediction accuracy further.

ACKNOWLEDGMENTS

This work is supported by The National Science Foundation under the award numbers CNS-1528799 and CNS-1647015 and a TIPS grant from Vanderbilt University. We acknowledge the support provided by our partners from Nashville Metropolitan Transport Authority.

REFERENCES

- [1] A. P. T. Association, "Ridership report archives," 2017.
- [2] J. Bates, J. Polak, P. Jones, and A. Cook, "The valuation of reliability for personal travel," *Transportation Research Part E: Logistics and Transportation Review*, vol. 37, no. 2, pp. 191–229, 2001.

- [3] A. Gooze, K. Watkins, and A. Borning, "Benefits of real-time transit information and impacts of data accuracy on rider experience," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2351, pp. 95–103, 2013.
- [4] "nmotion 2015," <https://www.surveymonkey.com/r/3KVYKVL?sm=UBsMYkzrQRFWv3mJyRiN5Q%3d%3d>, 2015, accessed: 2017-12-31.
- [5] C. Avishai, "Public transit planning and operation theory," modelling and practice," *Public transit planning and operation, Civil and Environmental Faculty*, 2007.
- [6] S. Pattnaik, S. Mohan, and V. Tom, "Urban bus transit route network design using genetic algorithm," *Journal of transportation engineering*, vol. 124, no. 4, pp. 368–375, 1998.
- [7] P. Chakroborty, "Genetic algorithms for optimal urban transit network design," *Computer-Aided Civil and Infrastructure Engineering*, vol. 18, no. 3, pp. 184–200, 2003.
- [8] Y. Hairong and L. Dayong, "Optimal regional bus timetables using improved genetic algorithm," in *Intelligent Computation Technology and Automation, 2009. ICICTA'09. Second International Conference on*, vol. 3. IEEE, 2009, pp. 213–216.
- [9] J.-Q. Li, M. Song, M. Li, and W.-B. Zhang, "Planning for bus rapid transit in single dedicated bus lane," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2111, pp. 76–82, 2009.
- [10] M. A. Nayeem, M. K. Rahman, and M. S. Rahman, "Transit network design by genetic algorithm with elitism," *Transportation Research Part C: Emerging Technologies*, vol. 46, pp. 30–45, 2014.
- [11] W. Y. Szeto and Y. Wu, "A simultaneous bus route design and frequency setting problem for tin shui wai, hong kong," *European Journal of Operational Research*, vol. 209, no. 2, pp. 141–155, 2011.
- [12] J. Hora, T. G. Dias, and A. Camanho, "Improving the robustness of bus schedules using an optimization model," in *Operations Research and Big Data*. Springer, 2015, pp. 79–87.
- [13] F. Sun, C. Samal, J. White, and A. Dubey, "Unsupervised mechanisms for optimizing on-time performance of fixed schedule transit vehicles," in *Smart Computing (SMARTCOMP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1–8.
- [14] Y. Wu, H. Yang, J. Tang, and Y. Yu, "Multi-objective re-synchronizing of bus timetable: Model, complexity and solution," *Transportation Research Part C: Emerging Technologies*, vol. 67, pp. 149–168, 2016.
- [15] A. Diveev and O. Bobr, "Variational genetic algorithm for np-hard scheduling problem solution," *Procedia Computer Science*, vol. 103, pp. 52–58, 2017.
- [16] L. Weigang, W. Koendjibiharie, R. de M Juca, Y. Yamashita, and A. MacIver, "Algorithms for estimating bus arrival times using gps data," in *Intelligent Transportation Systems, 2002. Proceedings. The IEEE 5th International Conference on*. IEEE, 2002, pp. 868–873.
- [17] S. I.-J. Chien, Y. Ding, and C. Wei, "Dynamic bus arrival time prediction with artificial neural networks," *Journal of Transportation Engineering*, vol. 128, no. 5, pp. 429–438, 2002.
- [18] R. H. Jeong, "The prediction of bus arrival time using automatic vehicle location systems data," Ph.D. dissertation, Texas A&M University, 2005.
- [19] D. Sun, H. Luo, L. Fu, W. Liu, X. Liao, and M. Zhao, "Predicting bus arrival time on the basis of global positioning system data," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2034, pp. 62–72, 2007.
- [20] S. I.-J. Chien and C. M. Kuchipudi, "Dynamic travel time prediction with real-time and historic data," *Journal of transportation engineering*, vol. 129, no. 6, pp. 608–616, 2003.
- [21] A. Shalaby and A. Farhan, "Bus travel time prediction model for dynamic operations control and passenger information systems," *Transportation Research Board*, vol. 2, 2003.
- [22] C. Bai, Z.-R. Peng, Q.-C. Lu, and J. Sun, "Dynamic bus travel time prediction models on road with multiple bus routes," *Computational intelligence and neuroscience*, vol. 2015, p. 63, 2015.
- [23] F. Sun, Y. Pan, J. White, and A. Dubey, "Real-time and predictive analytics for smart public transportation decision support system," in *Smart Computing (SMARTCOMP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1–8.
- [24] M. Chen, X. Liu, J. Xia, and S. I. Chien, "A dynamic bus-arrival time prediction model based on apc data," *Computer-Aided Civil and Infrastructure Engineering*, vol. 19, no. 5, pp. 364–376, 2004.
- [25] Y. Bin, Y. Zhongzhen, and Y. Baozhen, "Bus arrival time prediction using support vector machines," *Journal of Intelligent Transportation Systems*, vol. 10, no. 4, pp. 151–158, 2006.
- [26] B. Yu, W. H. Lam, and M. L. Tam, "Bus arrival time prediction at bus stop with multiple routes," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 6, pp. 1157–1170, 2011.
- [27] F. Sun, A. Dubey, and J. White, "Dxnat - deep neural networks for explaining non-recurring traffic congestion," in *Big Data (Big Data), 2017 IEEE International Conference on*. IEEE, 2017.
- [28] A. Oruganti, F. Sun, H. Baroud, and A. Dubey, "Delayradar: A multivariate predictive model for transit systems,"
- [29] F. Sun, A. Dubey, J. White, and A. Gokhale, "Transit-hub: A smart public transportation decision support system with multi-timescale analytical services," *Cluster Computing*, 2017.
- [30] J.-S. Yang, "Travel time prediction using the gps test vehicle and kalman filtering techniques," in *Proceedings of the 2005, American Control Conference, 2005*. IEEE, 2005, pp. 2128–2133.
- [31] C.-H. Wu, J.-M. Ho, and D.-T. Lee, "Travel-time prediction with support vector regression," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 5, no. 4, pp. 276–281, 2004.
- [32] V. Kumar, B. A. Kumar, L. Vanajakshi, and S. C. Subramanian, "Comparison of model based and machine learning approaches for bus arrival time prediction," in *Proceedings of the 93rd Annual Meeting. Transportation Research Board*, 2014, pp. 14–2518.
- [33] R. Jeong and R. Rilett, "Bus arrival time prediction using artificial neural network model," in *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*. IEEE, 2004, pp. 988–993.
- [34] E. Mazloumi, S. Moridpour, G. Currie, and G. Rose, "Exploring the value of traffic flow data in bus travel time prediction," *Journal of Transportation Engineering*, vol. 138, no. 4, pp. 436–446, 2011.
- [35] "Gtfs static overview," <https://developers.google.com/transit/>, 2017, accessed: 2018-01-31.
- [36] "General transit feed specification (gtfs) real-time overview," <https://developers.google.com/transit/gtfs-realtime/>, 2018, accessed: 2018-01-30.
- [37] F. Sun, Y. Pan, J. White, and A. Dubey, "Real-time and predictive analytics for smart public transportation decision support system."
- [38] T. Yin, G. Zhong, J. Zhang, S. He, and B. Ran, "A prediction model of bus arrival time at stops with multi-routes," *Transportation research procedia*, vol. 25, pp. 4623–4636, 2017.
- [39] 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, 2014.
- [40] Y.-y. Chen, Y. Lv, Z. Li, and F.-Y. Wang, "Long short-term memory model for traffic congestion prediction with online open data," in *Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on*. IEEE, 2016, pp. 132–137.
- [41] Y. Wang, F. Currim, and S. Ram, "Deep learning for bus passenger demand prediction using big data," 2016.
- [42] R. Caruana, "Multitask learning," in *Learning to learn*. Springer, 1998, pp. 95–133.
- [43] X. Liu, J. Gao, X. He, L. Deng, K. Duh, and Y.-Y. Wang, "Representation learning using multi-task deep neural networks for semantic classification and information retrieval," 2015.
- [44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [45] Keras, "Keras: The python deep learning library," 2017, [Online; accessed 30-September-2017]. [Online]. Available: <https://keras.io/>
- [46] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," 2011.