

# Bus Journey and Arrival Time Prediction based on Archived AVL/GPS data using Machine Learning

Ankit Taparia

*School of Computer Science and Statistics*  
Trinity College Dublin, The University of Dublin  
Dublin, Ireland  
tapariaa@tcd.ie

Mike Brady

*School of Computer Science and Statistics*  
Trinity College Dublin, The University of Dublin  
Dublin, Ireland  
brady@cs.tcd.ie

**Abstract**—With a surge in the number of vehicles, urban traffic congestion has increased in recent years. This has led to increased travel times and decreased accessibility and mobility. One option to mitigate this issue is to promote the use of public transport, including buses. To encourage the use of buses, there is a need to provide reliable travel time and arrival information to commuters.

In this study, we propose and develop predictive models to predict bus journey and arrival times based on historical AVL/GPS data, bus route and bus stop information. There were two parts to this study. The first was to predict overall journey times and the second was to predict bus arrival times at bus stops.

To estimate total bus journey times, three models were developed using Linear Regression, Artificial Neural Network (ANN) and Long Short Term Memory Network (LSTM). Evaluation on a ground-truth dataset shows that LSTM outperformed the Linear Regression model and its performance was comparable to that of ANN. To predict bus arrival times at bus stops, three different models, namely Historical Averaging, Linear Regression and Gradient Boosting are proposed. Experimental results show that Gradient Boosting outperformed the other models and is more robust in predicting arrival times.

Our study supports the idea that it is possible to predict bus journey time with reasonable accuracy using historical GPS observations and bus route information only.

**Index Terms**—Bus Journey Time Prediction, GPS Data, ANN, LSTM, Gradient Boosting

## I. INTRODUCTION

In recent years, urban traffic congestion has increased significantly across the world [1]. Congestion creates burdens on transportation infrastructure, increases travel time, fuel consumption and pollution, and reduces accessibility and mobility. One way to mitigate this problem is to encourage the use of public transport while efficiently manage existing resources using Intelligent Transportation Systems (ITS).

Buses are considered one of the important means of public transport owing to their coverage and accessibility. Buses are more economical and eco-friendly [2] than private vehicles. Moreover, in some cities buses have their own dedicated lanes which makes them faster than cars and helps reduce travel times in heavy traffic conditions.

Advanced Public Transportation Systems (APTs) are an integral component of Intelligent Transportation Systems. With the advent of intelligent transport systems in cities, buses are typically fitted with GPS-enabled in-vehicle navigation systems. These tracking systems can generate Automatic Vehicle Location (AVL) data which can be used to provide accurate bus arrival information to passengers waiting at bus stops, potentially leading to a decrease in waiting times.

Travel and arrival times depend on many external factors such as passenger load, passenger boarding/alighting time, number of signalised intersections, traffic congestion and weather, among other factors. Moreover, it is not guaranteed to have information about all the above factors to predict bus journey times. Hence there is a need to develop intelligent models which can estimate reliable journey and arrival times using minimal features for situations where all features are not available.

## II. RELATED WORK

Bus travel and arrival time prediction has been an active area of research for decades. Researchers have explored and applied various approaches and techniques for accurate bus arrival time prediction. In general, those approaches and techniques can be grouped into historical data approaches, statistical approaches and machine learning approaches – e.g. Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), K-Nearest Neighbors (KNNs), Long Short Term Memory (LSTMs) – and model based approaches (e.g. Kalman filtering).

Jeong and Rilett [3] presented an ANN model to estimate bus arrival times using automatic vehicle location. In addition to past journey data, the ANN model incorporated traffic information and bus dwell times at stops. A historic data-based model and several linear regression models were developed and compared with the ANN model. The results showed that the ANN performed considerably better than the historical data based model and regression models in terms of Mean Absolute Percentage Error (MAPE).

Fan and Gurmu [4] developed and compared Historical Average, Kalman Filtering and ANN models for predicting bus travel times using GPS data. The models were evaluated for a bus route in Macae city, Brazil. Experimental results suggested that the ANN outperformed the other models in

prediction accuracy. The ANN performed best when observed travel times were in a range 20 to 50 minutes but gave large prediction errors for very short and long trips.

In another study, Chen et al. [5] used automatic passenger counters (APCs) to predict bus arrival times for a specific bus route in New Jersey. The authors developed an ANN-based model for prediction which takes APC data features as input in addition to weather information during the same time period considered for analysis. It was observed that precipitation had a strong impact on bus delays.

A modified KNN method was developed by Liu et al. [6] using Principal Component Analysis to predict Bus Arrival Times (BATs) for a bus route in Beijing. The idea was to identify the most similar past sequences of trips to predict the BAT of the current trip. To evaluate the performance of the KNN model, the authors did an ANN model analysis and found that the proposed KNN performed better than the ANN.

Shalaby and Farhan [7] used AVL and APC data from a bus route in downtown Toronto to predict bus arrival times taking into account the effect of dwell times. Bus dwell time was calculated as a function of number of passengers boarding and alighting at a bus stop, which was obtained using APC data. The authors claim that, as the proposed model takes the effect of dwell times on bus arrival times into account, it outperformed regression and neural network models and was more robust.

In a study by Chien et al. [8], two ANNs were trained using link-based and stop-based data to estimate transit arrival times for bus route 39 of the New Jersey Transit Corporation. The authors used the microscopic simulation model CORSIM [9] to simulate bus operations and generate real time AVLS data for the specific route in the absence of GPS based AVL data. To further enhance the prediction accuracy, the authors introduced an adaptive algorithm and integrated it with the ANN models to take care of the prediction errors in real time.

Reddy et al. [10] developed a Support Vector Machine (SVM) model to predict bus arrival time and travel time under Indian traffic conditions, which are said to be prone to high variability due to lack of lane discipline and have heterogeneous vehicle profiles. The SVM model developed was compared with a Kalman filtering based model.

Several studies have used hybrid models combining two or more models to predict bus arrival times. In one such study, Zaki et al. [11] proposed a model in which an ANN is used along with Kalman filtering. The ANN predicts the time based on historical trips data and Kalman filtering adjusts the predictions made based on real time GPS information from the bus. In another study, Yu et al. [12] presented a hybrid model in which an SVM is used to predict the baseline travel times on the basis of historical trips while a Kalman filtering-based dynamic algorithm uses the latest bus arrival information to compute unexpected delays. Both combined together predict bus arrival times for subsequent stops. A similar approach is used by Seng et al. [13] who used a combination of static algorithm (SVM) and dynamic algorithm (Kalman filtering) for bus arrival time prediction. The authors

claim that the proposed model is more stable and can make accurate predictions even for unexpected situations.

This paper introduces a number of machine learning approaches for predicting two different aspects of bus journeys: the overall duration, and the duration of segments of a journey.

### III. DATA

#### A. Dataset Used and its Features

We used a dataset provided by Dublin City Council which was acquired from Smart Dublin [14]. It is a collection of global positioning system (GPS) points for buses in Dublin, Ireland from 6<sup>th</sup> November 2012 to 30<sup>th</sup> November 2012.

It contains  $\approx 35$  million rows and 15 features as explained below. Each row corresponds to a GPS observation and includes the following feature variables: *Timestamp*, *Line ID*, *Direction*, *Journey Pattern ID*, *Production TimeFrame*, *Vehicle Journey ID*, *Operator*, *Congestion*, *Long*, *Lat*, *Delay*, *Block ID*, *Vehicle ID*, *Stop ID*, *At Stop*. The ‘*delay*’ feature specifies the delay in seconds relative to the schedule whereas ‘*congestion*’ is a boolean feature. It is not known how the ‘*congestion*’ feature is generated and consequently is not used in this study.

New features – ‘*day of week*’, ‘*hour*’ are extracted from the Timestamp for each of the observations. A feature – ‘*distance from city centre*’ which denotes the distance of a bus from the Dublin City Centre for an observation is created. For this purpose, a point on O’Connell Street was designated the centre of the city. The Haversine formula [15] is used to calculate the distance between the designated centre point and the GPS coordinates of the bus.

### IV. TOTAL BUS JOURNEY TIME PREDICTION

After pre-processing the data and analysing journey trips for most observed bus route, we build predictive models to predict journey times for bus trips based on journey features.

#### A. Linear Regression

Linear regression [16] is a very simple and useful approach in supervised learning to predict a quantitative response. It models linear relationships between dependent and independent variables and helps to determine how strong those relationships are.

We use the features ‘*day of week*’, ‘*hour*’ and ‘*delay*’ to build the model. Therefore these are the independent variables whereas ‘*journey time*’ is the dependent variable which the model will predict.

From the data, maximum GPS data observations were recorded for three routes (Line IDs) - 46, 40 and 145. Route 46 was considered for analysis as it was possible to extract a significant number of trips from this particular route. A total of 1989 journeys of bus route 46 are used of which 80% are used for training and the remaining 20% for testing the model.

#### B. Artificial Neural Network

Artificial Neural Networks (ANN) [17] are a powerful technique for capturing and modelling non linear complex relationships between inputs and outputs. Artificial Neural

Networks have been a popular choice among the researchers for travel time prediction [4] [5] [11]. They can identify relationships and patterns in datasets and can approximate any arbitrary input-output mapping.

We use a four layered feed-forward neural network which consists of an input layer, two dense hidden layers and an output layer. The features are fed into the network through input layer which has 24 neurons. The hidden layers consist of 25 neurons each with each neuron in one hidden layer connected to other neurons in the next layer. Hidden layer neurons are activated by a Rectified Linear Unit (ReLU) [18] activation function. Finally, the output layer consists of one node with linear activation function which gives the predicted journey time. Biases are included in hidden and output layers. A regularisation technique - Dropout [19] with value of 0.3 is applied to the hidden layers to prevent the neural network model from overfitting and making it robust to unseen inputs.

The model is trained using 1591 bus trips and the network is validated over 200 trips of bus route 46 with a batch size of 50 for 50 epochs. ReLU activation functions are used in hidden layers. An Adam optimiser is used to update network weights with a learning rate of 0.01.

### C. LSTM

Long Short Term Memory (LSTM) networks [20] are a variant of Recurrent Neural Networks (RNNs) which are suitable for processing long sequence of inputs and predicting time series whereas RNNs suffer from the short term memory.

The LSTM model used has one input layer, two dense LSTM layers and an output layer. It is a double stacked LSTM with the output from the first LSTM layer at each time step being fed to the second LSTM layer. The output of the second LSTM layer goes into a dense layer, which is a fully connected neural network. Finally, the dense layer consists of one node on which tanh activation is applied and gives the predicted journey time. An important model parameter – look\_back which denotes the number of past journey times to use as input variables to predict the next journey time – is set to 15. The inputs are normalised before being fed into the network. In addition, dropout of 0.2 is applied to LSTM layers to prevent overfitting.

Unlike previous models, our univariate LSTM model takes in only one feature – past sequence of ‘Journey times’ as input. The LSTM model learns a function that maps a sequence of past journey times as input to an output observation. One major difference between ANN and LSTM is that during the training process the ANN assumes that the data samples are independent of each other whereas the LSTM assumes each sample is dependent on the previous samples.

The model is trained on a training set consisting of journeys of bus route number 46 from 6<sup>th</sup> November 2012 to 25<sup>th</sup> November 2012. It is then tested to predict journey times for the same bus route from 26<sup>th</sup> November 2012 to 30<sup>th</sup> November 2012. The training set consists of 1609 journeys and the test set comprises 402 journeys. The model is trained with a batch size of 50 samples for 200 iterations.

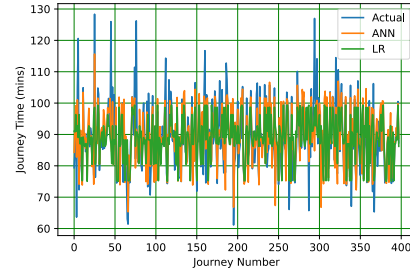


Fig. 1. Comparison: Linear Regression vs ANN

### D. Results and Evaluation

To visually compare the performance of the Linear Regression and ANN for predicting total journey time, we plot their corresponding predictions on the same graph as shown in Figure 1.

It can be observed that the ANN performs better than Linear Regression for almost all the journeys. In particular, for journeys which have lower travel times, the ANN’s predictions are closest to the actual travel time. To further strengthen the argument, evaluation metrics – MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), RMSE (Root Mean Square Error) are calculated for both models and are summarized in Table I.

TABLE I  
RESULTS OBTAINED

Model	MAE (mins)	MAPE (%)	RMSE (mins)
<b>Linear Regression</b>	7.6130	7.2406	10.0342
<b>Artificial Neural Network</b>	4.3491	4.8364	6.7541

It can be observed that the ANN outperformed the Linear Regression model for all the three evaluation metrics. However, both models make large prediction errors for some trips with longer travel times. It may be that those trips were completed during rush hour and thus models could not make reliable predictions due to lack of traffic congestion information in the dataset.

Figure 2 represents the actual and the predicted total journey times by LSTM for bus journeys of route 46 completed from 6<sup>th</sup> November to 30<sup>th</sup> November 2012 in chronological order.

It can be observed that the LSTM is able to predict the peaks and drops in journey times much better than the Linear Regression model and its performance is comparable to that of the ANN we discussed earlier. It reveals that the time taken to complete a future journey depends on the previous sequence of journeys completed. An MAE of 4.28 minutes for LSTM is an improvement of around 43% over the Linear Regression. Values of other evaluation metrics – MAPE of 4.7312% and RMSE of 6.78 minutes are recorded for the test data.

One key observation to note is that even without taking traffic and time-related features such as ‘day of week’, ‘hour’,

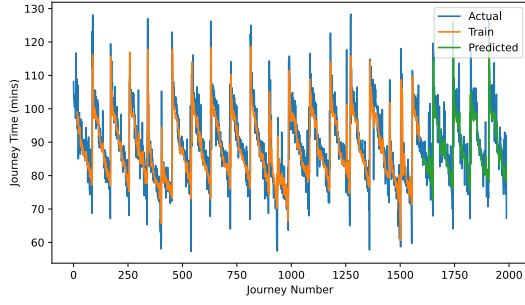


Fig. 2. LSTM: Actual vs Predicted Journey Time and Future Trends for bus route 46

‘delay’ as input, the LSTM is able to distinguish between peak hours and non peak hours, and also differentiate between weekdays and weekends for which journey times follow different patterns. It suggests that LSTM derives some intrinsic traffic information from the past journey times sequence, memorizes it, retains the information and uses it for future predictions.

#### V. JOURNEY TIME PREDICTION OF ROUTE SEGMENTS

In the prior sections, we developed various models to predict the journey time a bus takes to traverse the whole route. Besides this, it is also useful to predict bus journey times for segments of a route. A segment of a bus route is a section of route between any two bus stops. If we can predict how much time a bus takes to travel from Stop A to Stop B, it would provide bus arrival information to a passenger waiting at Stop B and potentially reduce waiting time for passengers at the stop.

#### VI. PREDICTIVE MODELS FOR BUS ROUTE SEGMENT TRAVEL TIME/ARRIVAL TIME

Three predictive models are studied and evaluated.

##### A. Historical Averaging Model

The Historical Averaging model predicts the travel time of a future trip from observed historical bus travel time data of past journeys completed in the same daily time period over different days. It assumes the traffic conditions in the current time to be same as in the past while making predictions. Analysing the previous trends in bus journey times, a day has been divided into four time periods: Period-1 (06:00—10:00), Period-2 (10:00—14:00), Period-3 (14:00—18:00) and Period-4 (18:00—22:00). Now suppose we have  $N$  past bus trips data completed during Period-1 such that  $t_{a,b}^i$  represents the time taken by the bus to travel from stop  $a$  to  $b$  in the  $i^{th}$  trip where  $i = 1$  to  $N$ ,  $a < b$  and  $a, b \in k$ . So for a current bus trip  $c$  operating during Period-1, the historical averaging model predicts the time the bus will take to travel from stop  $a$  to stop  $b$  is given as:

$$t_{a,b}^c = \frac{\sum_{i=1}^N t_{a,b}^i}{N} \quad (1)$$

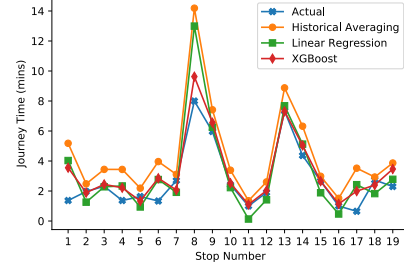


Fig. 3. Predicted and Actual travel times from the previous stop

##### B. Linear Regression Model

The Linear Regression model which captures linear relationships between dependent and independent variables is used to predict the travel times between the stops.

The model takes in input the features – ‘distance between the stops’, ‘distance of source stop from city centre’, ‘distance of destination stop from city centre’, ‘day of week’, ‘hour of day’ and ‘delay’. It outputs the time taken to travel between the source stop and the destination stop.

##### C. Gradient Boosting Model

Gradient Boosting [21] is a machine learning technique in which models are created and trained in a gradual, additive and sequential manner. Each model learns from the mistakes of previous models to minimize the error. The key principle behind the gradient boosting algorithm is to find a new sub-model to compensate for the residual error created by the previous submodel [22]. For implementation purposes, we used the XGBoost [23] library of Python. The same set of feature variables as used in the Linear Regression model are fed to the Gradient Boosting model to predict the bus travel times between adjacent stops.

##### D. Results and Evaluation

By considering the 20 most frequently marked stops of bus route 46, a total of 459 bus trips having the same sequence of 20 marked stops are obtained for the period between 6<sup>th</sup> November and 30<sup>th</sup> November 2012. Of these, 369 trips took place from 6<sup>th</sup> to 26<sup>th</sup> November 2012. We considered it as our training set. The remaining 90 trips made between 27<sup>th</sup> and 30<sup>th</sup> November 2012 comprises our test set for which we make the predictions.

The prediction of the three models and the actual arrival times are shown in Figure 3 which represents the predicted and the actual travel time between two adjacent stops for one of the test journeys. Figure 4 represents the absolute deviation, that is, the absolute difference between the predicted and actual travel time between stops for the particular bus trip shown in Figure 3.

It is evident from both the plots that the Gradient Boosting (XGBoost) gives better predictions, more closely approximates the actual travel times and outperforms the other two models. To reinforce this claim and further assess the performance of

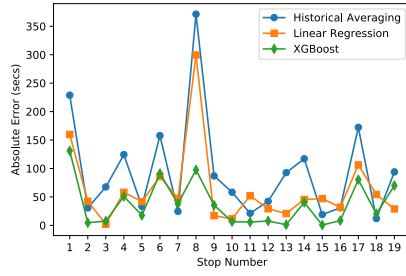


Fig. 4. Absolute Deviation between Predicted and Actual travel times from the previous stop

TABLE II  
EVALUATION METRICS RECORDED FOR THE MODELS

Model	MaxAE (mins)	MAE (mins)	RMSE (mins)
Historical Averaging	11.28	1.02	2.37
Linear Regression	10.58	0.93	1.45
Gradient Boosting (XGBoost)	9.62	0.80	1.16

the models, evaluation metrics – MaxAE (Maximum Absolute Error), MAE, RMSE are calculated for each model and listed in Table II.

It can be observed from the Table II that Gradient Boosting has the lowest values for each of the metrics. On the other hand, Historical Averaging has the highest values for all metrics. However, we observe that for all three models the values of MAE are small. This is due to the fact that bus stops in Dublin are quite close to each other with most of them within the range of 300 metres [24]. Travel times between adjacent stops are low. Therefore, even a small MAE can cause large deviation from actual time and is undesirable. Figure 5 shows the RMSE values of the models for the test journeys.

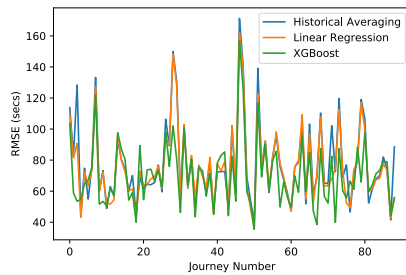


Fig. 5. Root Mean Squared Error (RMSE) for test journeys

To further assess the performance and the robustness of the models in predicting the segment travel time, three different segments shown in Table III are considered.

The MAPE is used as the measure of the models' performance. Figure 6 shows the MAPE values of the three models for the three segments. It can be observed that Historical Averaging has the highest MAPEs, whereas Gradient Boosting (XGBoost) has the lowest MAPEs for all three segments. The

TABLE III  
SEGMENTS

Segments	Start Stop No	End Stop No	Segment length (km)
Segment 1	0	7	3.96
Segment 2	0	9	10.927
Segment 3	0	19	17.54

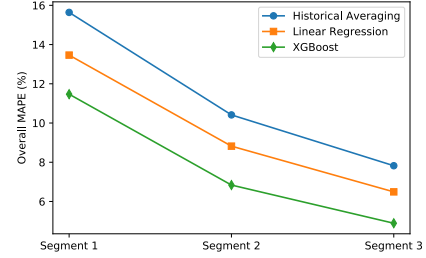


Fig. 6. Overall MAPE for segments

results also reveal a decreasing trend in MAPE with increase in segment length for all the models. Since smaller segments has fewer bus stops, fewer traffic signals and fewer intersections, large variances in traffic, bus dwell times and signal delays can be expected. For example, there may be instances for short segments when buses don't get stuck at traffic signals and reach the stops ahead of their scheduled time. But there may be other instances when the bus has to wait at every signal thus arriving late at the stops. Moreover, for short segments, journey times are small, and even a small difference between actual times and predicted times will lead to high MAPE. However, with the increase in segment length, the average values of the above mentioned factors tend to be more stable, and better prediction accuracy can be obtained.

A lower MAPE value usually suggests that the model is good in making predictions. However, a model with a small MAPE may occasionally yield a prediction with a large deviation. This is undesirable as it may predict the arrival time very different from the actual time. This might lead to passengers missing the buses. Therefore, it is critical to assess the robustness of the model to check if its maximum deviation is within a certain range. Hence, we use MaxMAPE (Maximum Absolute Percentage Error) which is defined as  $\text{MaxMAPE} = \max\{\text{MAPE}\}$  of a segment, to measure the robustness of the models. As it can be seen from Figure 7, the maximum MAPE values of the Historical Averaging and Linear Regression models are greater than those of Gradient Boosting for the three segments we considered for analysis. Therefore, Gradient Boosting is found to be more robust than the other two models.

## VII. CONCLUSION AND FUTURE WORK

In this research study we proposed and developed various models to predict overall bus journey times and arrival times using historical AVL/GPS data and prior routes information.

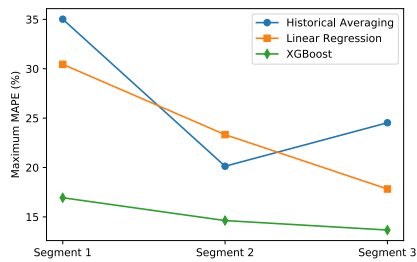


Fig. 7. MaxMAPE for segments

The models developed are evaluated on a ground-truth dataset of Dublin buses and their performances are compared.

In the first half of our study, we developed Linear Regression and ANN models to estimate total time a bus takes to traverse the whole route. We also proposed a univariate LSTM to predict the total bus journey times based on the time-tagged history of previous journeys. Experimental results reveals that the LSTM outperformed Linear Regression and its performance is comparable to that of the ANN. This is despite the fact that the LSTM did not take into account time-related features and traffic information for making the predictions.

The second half of our study explored techniques and methods to predict bus arrival times at stops. From the experimental results obtained, it is found that Gradient Boosting outperformed Historical Averaging and Linear Regression models on prediction accuracy and robustness. Its strategy of building and combining models in a sequential manner to minimize the errors made by the previous submodels helps to obtain better prediction accuracy. To further assess the performance and robustness of models, three route segments of varying lengths are considered. For each of the segments, Gradient Boosting gives the least MAPE and MaxMAPE of the three models and is found to be more robust in making predictions.

As part of future work, data regarding external factors such as traffic congestion, weather could be gathered and incorporated into the models for better predictions. One limitation of this study is that we assumed that the bus dwell times are included in the segment travel time, so they are not considered explicitly. To overcome this, some sophisticated techniques could be employed to derive the bus dwell times at stops from the data we used for our research. It could also give a measure of passenger load at the bus stops and might potentially improve bus arrival time predictions at downstream stops. Lastly, another interesting future work direction would be to develop a hybrid prediction model combining ANN, LSTM and Gradient Boosting. It would be interesting to see if the hybrid model could combine the individual strengths of the models to overcome their weaknesses and make better predictions.

## REFERENCES

- [1] TomTom Traffic Index, 01 2020. [Online]. Available: <https://www.globenewswire.com/news-release/2020/01/29/1976528/>

- 0/en/TomTom-Traffic-Index-Global-Traffic-Congestion-Up-as-Bengaluru-takes-Crown-of-World-s-Most-Traffic-Congested-City.html
- [2] M. Lowe, B. Aytekin, and G. Gereffi, *Public transit buses: A green choice gets greener*. Center on Globalization Governance and Competitiveness, 2009.
- [3] R. Jeong and R. Rilett, "Bus arrival time prediction using artificial neural network model," in *Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No. 04TH8749)*. IEEE, 2004, pp. 988–993.
- [4] W. Fan and Z. Gurm, "Dynamic travel time prediction models for buses using only GPS data," *International Journal of Transportation Science and Technology*, vol. 4, no. 4, pp. 353–366, 2015.
- [5] M. Chen, J. Yaw, S. I. Chien, and X. Liu, "Using automatic passenger counter data in bus arrival time prediction," *Journal of Advanced Transportation*, vol. 41, no. 3, pp. 267–283, 2007.
- [6] T. Liu, J. Ma, W. Guan, Y. Song, and H. Niu, "Bus arrival time prediction based on the k-nearest neighbor method," in *2012 Fifth International Joint Conference on Computational Sciences and Optimization*. IEEE, 2012, pp. 480–483.
- [7] A. Shalaby and A. Farhan, "Bus travel time prediction model for dynamic operations control and passenger information systems," *Transportation Research Board*, vol. 2, 2003.
- [8] 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.
- [9] A. Halati, H. Lieu, and S. Walker, "CORSIM-corridor traffic simulation model," in *Traffic Congestion and Traffic Safety in the 21st Century: Challenges, Innovations, and Opportunities*. Urban Transportation Division, ASCE; Highway Division, ASCE; Federal Highway Administration, USDOT; and National Highway Traffic Safety Administration, USDOT, 1997.
- [10] K. K. Reddy, B. A. Kumar, and L. Vanajakshi, "Bus travel time prediction under high variability conditions," *Current Science*, pp. 700–711, 2016.
- [11] M. Zaki, I. Ashour, M. Zorkany, and B. Hesham, "Online bus arrival time prediction using hybrid neural network and Kalman filter techniques," *International Journal of Modern Engineering Research*, vol. 3, no. 4, pp. 2035–2041, 2013.
- [12] B. Yu, Z.-Z. Yang, K. Chen, and B. Yu, "Hybrid model for prediction of bus arrival times at next station," *Journal of Advanced Transportation*, vol. 44, no. 3, pp. 193–204, 2010.
- [13] D. Seng, J. Peng, J. Chen, and N. Zheng, "Bus arrival time prediction based on static and dynamic algorithms," *Advances in Transportation Studies*, 2015.
- [14] Dublin Bus GPS Data. [Online]. Available: [https://data.smartdublin.ie/dataset/dublin\\_bus\\_sample/resource/26c468ec-8e19-4cd1-8ec4-1597b8a54fe2](https://data.smartdublin.ie/dataset/dublin_bus_sample/resource/26c468ec-8e19-4cd1-8ec4-1597b8a54fe2)
- [15] C. C. Robusto, "The cosine-haversine formula," *The American Mathematical Monthly*, vol. 64, no. 1, pp. 38–40, 1957.
- [16] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. Springer, 2013, vol. 112.
- [17] A. K. Jain, Jianchang Mao, and K. M. Mohiuddin, "Artificial neural networks: a tutorial," *Computer*, vol. 29, no. 3, pp. 31–44, 1996.
- [18] J. Brownlee, "A Gentle Introduction to the Rectified Linear Unit (ReLU)," *Machine Learning Mastery*, 01 2019. [Online]. Available: <https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/>
- [19] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [21] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.
- [22] L. Mason, J. Baxter, P. L. Bartlett, and M. R. Frean, "Boosting algorithms as gradient descent," in *Advances in neural information processing systems*, 2000, pp. 512–518.
- [23] T. Chen, T. He, M. Benesty, V. Khotilovich, and Y. Tang, "XGBoost: extreme gradient boosting," *R package version 0.4-2*, pp. 1–4, 2015.
- [24] National Transport Authority, "Core bus network report." [Online]. Available: [https://data.smartdublin.ie/dataset/dublin\\_bus\\_sample/resource/26c468ec-8e19-4cd1-8ec4-1597b8a54fe2](https://data.smartdublin.ie/dataset/dublin_bus_sample/resource/26c468ec-8e19-4cd1-8ec4-1597b8a54fe2)