Arrival Time Prediction for Autonomous Shuttle Services in the Real World: Evidence from Five Cities

Carolin Schmidt*, Mathias Tygesen*, And Filipe Rodrigues

Abstract—**Urban mobility is on the cusp of transformation with the emergence of shared, connected, and cooperative automated vehicles. Yet, for them to be accepted by customers, trust in their punctuality is vital. Many pilot initiatives operate without a fixed schedule, thus enhancing the importance of reliable arrival time (AT) predictions. This study presents an AT prediction system for autonomous shuttles, utilizing separate models for dwell and running time predictions, validated on real-world data from five cities. Alongside established methods such as XGBoost, we explore the benefits of integrating spatial data using graph neural networks (GNN). To accurately handle the case of a shuttle bypassing a stop, we propose a hierarchical model combining a random forest classifier and a GNN. The results for the final AT prediction are promising, showing low errors even when predicting several stops ahead. Yet, no single model emerges as universally superior, and we provide insights into the characteristics of pilot sites that influence the model selection process. Finally, we identify dwell time prediction as the key determinant in overall AT prediction accuracy when autonomous shuttles are deployed in low-traffic areas or under regulatory speed limits. This research provides insights into the current state of autonomous public transport prediction models and paves the way for more data-informed decision-making as the field advances.**

Index Terms—**Machine Learning, Autonomous Public Transport, Arrival Time Prediction, Graph Neural Networks**

I. Introduction

SHARED, connected, and cooperative automated vehicles offer a unique opportunity for a fundamental change in urban mobility. They can provide seamless door-to-door mobility of people and freight delivery services, which can lead to more accessible, greener, and more sustainable cities - provided they are integrated into an effective public transport system. For users to accept this new mode of transportation, they need to trust its punctuality, making reliable AT prediction systems crucial for a satisfactory user experience. Fortunately, connected automated vehicles also constitute a unique Big Data source. When coupled with the latest developments in Machine Learning and Artificial Intelligence, this data can enable the implementation of AT prediction systems. But the question arises - Do the findings about established machine learning models for conventional busses transfer to their automated counterparts? With the recent developments in both technological and regulatory matters, autonomous public transport is still in pilot testing and is constrained by local regulations. The data that these pilot projects yield, though

invaluable, is limited and varies in quality. Thus, it is unclear if this data can serve as a foundation for reliable prediction models. Moreover, so far, the primary function of autonomous shuttles in real-life case studies has been to connect existing public transport trunk lines with remote areas, such as schools, residential care facilities, and shopping centers. Typically, there is no historical data, such as passenger counts, from existing infrastructure that can be used as an indication for this new service. Furthermore, these shuttles operate without fixed schedules, rendering schedule adherence infeasible and thus making AT prediction especially challenging, yet, at the same time, an even more important task.

To our knowledge, this paper is the first to evaluate various AT prediction models incorporating dwell time and running time predictions in the context of real-life case studies of autonomous shuttles. We approach the AT prediction problem as two distinct regression problems for dwell and running time and put a spectrum of machine learning models, from the established tree-based methods to more complex graph neural networks, to the test, all while drawing insights from data from five diverse pilot sites. This work is an initial study of the performance of these models in autonomous public transport within a real-world autonomous public transport setting.

The structure of the paper is as follows. First, we give an overview of related work on bus AT prediction. Following, we introduce our methodology on our segment-based approach to distinguish between dwell and running time along with the corresponding prediction models. Then, we discuss the data pre-processing steps before presenting the final results. Finally, we conclude our work while giving future research directions.

II. Related work

A. Bus Arrival Time prediction

Bus AT prediction is a large research area within Intelligent Transport Systems. Historically, Kalman Filters has been a common approach to AT predictions due to their capability to maintain states between predictions and filter noise [1, 2, 3]. More recently, Machine Learning models have shown good performance for AT predictions. Shallow Machine Learning techniques like Support Vector machines [4] and Random Forest [5] have been utilized for AT; however, most work has focused on models based on Artificial Neural Networks (ANN), e.g. [6] created a hierarchical ANN model and showed that it outperformed Kalman Filters for short-distance routes. Long Short-Term Memory models (LSTMs) have been popular

for predicting AT due to their ability to capture the non-linear behavior of time-series [7, 8, 9]. In [10], the authors combine LSTMs with Convolutional Neural Nets to learn the temporal and spatial correlations between bus stops. [11] used Graph Neural Networks (GNNs) for city-wide bus travel time estimation. However, unlike our approach, they do not split travel time into dwell time and running time.

In [12], the authors show that dwell time is among the most important factors determining AT. As such, it can be beneficial for AT prediction to split the task into predicting the dwell and running time separately and then accumulate the predictions afterward. In [13], the authors propose a segment-based approach to predict bus AT by dividing bus routes into dwelling and transit segments. The authors find bus stops by clustering GPS signals and extracting dwell and running times from the GPS traces. The final AT predictions are then generated by accumulating the independent dwell and running time predictions. [10] also argue for splitting bus AT models into travel and dwell time models. The authors utilize LSTM and CNNs for the running time predictions and a more simple exponential smoothing model for dwell time predictions. For a more in-depth survey of general AT prediction research, we refer to [14].

As autonomous shuttles are a new technology, they have not yet seen much real-world application, and as such, the research on predicting arrival times for automated shuttles is lacking. [15] did an initial study of the running time of autonomous shuttles. The authors used the current location along the current segment, the elapsed time on the current segment, and the shuttle's speed and acceleration as input to a Gradient Boosting Regression Tree model to predict the remaining time on the current segment. The authors added artificial stops to the route as the study was done in the early stages of an autonomous shuttle pilot.

As our work is based on data from more advanced pilot sites, we are, to the best of the authors' knowledge, the first to do dwell and running time predictions on data from autonomous shuttles in real-life deployment. Furthermore, we are also the first to utilize GNNs and hierarchical GNNs for split dwell and running time predictions.

III. METHODS

Dwell times are mainly impacted by the uncertain numbers of boarding passengers and hence show structural differences to running times, which are primarily impacted by traffic conditions. Therefore, in this work, we adopt a segment-based approach, employing two separate models to predict the running and dwell times. Figure 1 provides an overview of the segment-based approach. The dwell time corresponds to the time the shuttle is stationary at a shuttle stop and the running time to the time it travels between two stops including acceleration and deceleration time. To obtain the final travel time $T_{i,j}$ of the autonomous shuttle from stop i to stop j, we aggregate the predictions for the individual segments for the dwell time dt and running time tt along the route:

$$T_{i,j} = \sum_{k=i}^{j-1} tt_{k,k+1} + \sum_{k=i+1}^{j-1} dt_k. \qquad (1)$$

IV. DWELL AND RUNNING TIME PREDICTION MODELS

Given that current data on autonomous shuttles is confined to test pilots, our models must operate on limited input. Many studies have equipped their models with real-time traffic information from available car data i.e. taxi data [13] to obtain accurate running time predictions. However, as autonomous shuttles often operate on both shared public roads and private lanes or share lanes only with pedestrians and cyclists, leveraging standard road traffic speed estimates becomes unfeasible. Moreover, these estimates might not be as impactful in the case of autonomous shuttles, as they adhere to specific speed regulations designed for such vehicles, which are often lower than the road's general speed limit. Hence, our approach adopts a time-series-based approach, where we include previous observations of the running and dwell time, respectively, on the segment into the regression task. However, given the data sparsity resulting from pilot testing, the most recent observations can quickly become outdated and, thus, less reliable. Hence, our models only include the two previous observations per segment. As such, our utilized features include the current position of the shuttle, current weather data, current time, and the ID of the segment and the vehicle in question, along with the two most recent observations. So given a vehicle id i, segment id j and time t the input to our prediction model are:

$$\boldsymbol{x}_{i,j,t} = \left[\tau_t, w_t, v_i, s_j, l_{j,t}^{(1)}, l_{j,t}^{(2)} \right] \qquad (2)$$

where τ_t is the geometrically transformed time-of-day and day-of-week features, w_t is the temperature, precipitation, and windspeed, v_i is a one-hot encoding of the vehicle id, s_j is a one-hot encoding of the segment id and $l_{j,t}^{(1)}$ and $l_{j,t}^{(2)}$ are the most recent and second most recent lag of segment j at time t. The lags are either historical dwell time or running time, depending on which is being predicted. Then we define a model $\mathcal{F}$ that takes in the input data $x_{i,j,t}$ and predicts the next running or dwell time for that vehicle on the segment, i.e.

$$\boldsymbol{x}_{i,j,t} \xrightarrow{\mathcal{F}} y_{i,j,t}. \qquad (3)$$

Our prediction models employ machine learning techniques that have proven successful in predicting both the dwell and the running times of conventional buses. Studies highlighted the competitive performance of tree-based methods, especially XGBoost, in both dwell time [16, 17], travel time ([18] and running time predictions [19]). XGBoost [20], which stands for eXtreme Gradient Boosting, is an optimized gradient boosting algorithm that builds on decision trees. In addition, our work also examines three deep learning models:

- A Multi-Layer Perceptron (MLP) that consists of fully connected layers.
- A Graph Convolutional Network (GCN) [21]. GCNs share information among nodes in the graph utilizing a permutation-invariant propagation rule. This way GCNs are able to capture spatial correlations in the data.
- A hierarchical model that combines a Random Forest (RF) [22] followed by GCN. Drawing from the tools from zero-inflated regression, the binary RF classifier handles
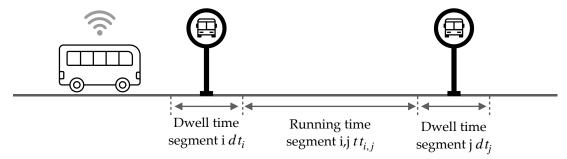
Fig. 1: Our segment-based approach. The route is divided into dwell and running time segments.

zero observation while the downstream GCN regresses the non-zero observations. We call this model RF-GCN. Given the high temporal resolution of our data sets, when the shuttle is skipping a stop, we can determine a dwell time of exactly 0s. In pilot tests, the demand is very volatile, resulting in a data set with a potentially significant number of zeros. Detecting those zeros is important for AT prediction, as overestimating the dwell time when it is actually zero leads to an overestimation of the total travel time. As a result, the shuttle departs earlier than predicted, which significantly deteriorates the customer experience. Further, for GCN-based approaches, the input is a matrix X where the k'th row is $x_{i,k,t}$. The GCN model then predicts a vector, $y_{i,t}$, with a prediction for each node. The prediction for segment j can then be taken out as the j'th position.

V. Data

The data utilized for our models were gathered from five cities conducting pilot tests with autonomous shuttles, all of which are part of the SHOW (SHared automation Operating models for Worldwide adoption) project [23]. The SHOW Project is a large-scale EU initiative aiming to demonstrate the viability and effectiveness of shared, connected, cooperative, and electrified fleets of Autonomous Vehicles (AVs) in real-world urban demonstrations across 20 cities in Europe. Altogether, the project will deploy a fleet of more than 79 SAE L4/L5 AVs for both passenger and cargo transport in dedicated lanes as well as mixed traffic. In this paper, we test our models on data from five of these pilot sites (Table I), where tests have been ongoing for several months. Currently, all vehicles are automated at level 4. The pilot sites in Table I represent not only different stages of deployment but also some have a particular focus, such as teleoperation or the challenge of interfacing with different road users.

The quality of the provided data varies significantly with the pilot site; while all provide GPS positioning and speed data, some have deviations. For instance, the speed data from Linköping is noisy, and in Les Mureaux, the sampling rate of the GPS data is decreased mid-pilot. Such variations are not limited to our study but are representative of many other pilots, both current and upcoming. Linköping stands out as the site with the longest ongoing pilot, as well as the longest route. At the other end is Graz, with the shortest route and a very limited number of observations. These distinct attributes

across pilot sites provide an optimal foundation for our model evaluations.

A. Data pre-processing

To enable a clear distinction between run and dwell time in public transport systems, careful data pre-processing is crucial. We consider two scenarios: One where we have both speed and GPS data and another where we rely solely on GPS positioning. In the first scenario, with the combination of speed and GPS data, we have explicit information about when a shuttle is in motion or stationary. To determine the dwell time, we define a shuttle as dwelling when it is at a standstill (speed of 0 km/h) and within a predetermined radius of a stop. As soon as the shuttle starts moving, we start the running time calculation for the corresponding route segment. If the shuttle bypasses a stop without stopping, we consider the dwell time as zero.

However, in some cases, e.g. in Linköping, GPS positioning might be the only data at the disposal, and the information about the shuttle's movement is less explicit. In this scenario, we establish a threshold for GPS differences to detect when a shuttle dwells at a stop while accounting for the noise (GPS jitter) in the GPS data. Dwell times are then calculated based on periods of minimal movement, as defined by the threshold. A predefined stop radius remains equally relevant here to identify when the shuttle is at a stop.

There are additional cases that require specific attention in our pre-processing. There are instances when a shuttle is parked between runs or traveling to and from its depot without turning off the GPS device. We manually identify these relevant locations and exclude instances where the shuttle deviates from its designated route from our data set. We also handle situations where two stops on opposite sides of the road have overlapping radii by hard-coding all possible stop orders in the pre-processing to ensure precise calculations. This way, we can precisely determine the relevant shuttle stop for each data point, eliminating ambiguity and potential errors. Other site-dependent adjustments must be considered, such as the change in GPS sampling frequency in Les Mureaux. Here, the GPS sampling rate was reduced mid-pilot, affecting time calculations, which required us to discard the impacted data for consistency. For the GNN, the data must be structured as a graph with nodes and edges linking these nodes based on an adjacency matrix. In the context of dwell time prediction, each

TABLE I: Data Stats

Site	Country	#Routes	#Stops	Dates	Avg. speed	Max speed	#Vehicles	#Observations
Linkoping	Sweden	2	15	12/21-06/23	7.7 km/h	22 km/h	3	48 679
Tampere	Finland	1	7	12/21-03/22	8.2 km/h	64 km/h	2	6 696
Les Mureaux	France	3	7	11/22-02/23	2.2 km/h	6 km/h	2	4 580
Madrid	Spain	1	5	06/22-02/23	4.1 km/h	36 km/h	2	7 096
Graz	Austria	2	4	10/22-06/23	14.3 km/h	57 km/h	1	359

node represents a stop, while for the running time prediction, the nodes correspond to the segments between stops. The connectivity of the graph follows the different routes of the shuttle. We illustrate the two graphs for Linköping in Figure 2.

Finally, we enrich our data set with weather data at an hourly resolution and perform sine and cosine encodings of time-of-day and day-of-week features. For a correct model evaluation, we hold out a part of the data set depending on the date to avoid data leakage. In Linköping, Tampere, Madrid, and Graz, our test set selects one month of data and one week in Les Mureaux.
The data pre-processing step is of utmost importance to make an accurate distinction between dwell time and running time. Since our data originates from pilot tests, careful data cleaning is essential to obtain a data set that reflects regular operation as accurately as possible.

VI. RESULTS

In this section, we perform a series of experiments on the data sets presented in the previous section. First, we investigate whether it is beneficial to create historical lags based on all vehicles or per vehicle. Then we compare the different model's performance in dwell time and running time predictions. Lastly, we aggregate the predictions from the models to calculate AT predictions along observed journeys from the test data to evaluate the aggregation error.

As baselines for our prediction models, we compare the performance of the Lag model, which predicts the most recent observation, and the Mean regressor, also often referred to as a historical average model, which predicts the mean of each segment. The hyperparameters and training scripts for all evaluated models are available. [1] As performance metrics, we report the Root Mean Square Error (RMSE) as well as the Mean Absolute Error (MAE):

$$\text{RMSE}(y, \hat{y}) = \sqrt{\frac{\sum_{i=0}^{N-1}(y_i - \hat{y}_i)^2}{N}} \tag{4}$$

$$\text{MAE}(y, \hat{y}) = \frac{\sum_{i=0}^{N-1}|y_i - \hat{y}_i|}{N} \tag{5}$$

where y is the true labels and $\hat{y}$ is the predictions for the test set with N observations.

Prior to assessing the final prediction results, we need to determine whether lags are taken from the most recent observation of the individual vehicle or the whole fleet. Hence,

---

[1] https://github.com/carolinssc/Arrival-time-prediction-autonomous-shuttles

in Table II, we compare the predictive performance of the Lag Model and XGBoost on running time depending on the lags. Pilot testing occasionally results in sparse data, making recent lags from the whole fleet potentially advantageous. If, on the other hand, the fleet is a mix of shuttles from different manufacturers, observations from different vehicles may not be as informative. As seen in Table II, the results are not consistent across sites. Les Mureaux and Tampere deploy the same type of vehicles, while the remaining sites test different manufacturers. As a consequence, we can see a clear improvement in Linköping, with the highest number of vehicles deployed, when providing lags from the respective vehicle. For the other pilot sites, the difference is less striking. In Les Mureaux, the vehicles are identical, which leads to a better lag model for the lags based on the whole fleet. Yet, XGBoost performs better at the individual vehicle level, indicating subtle differences in the data of the vehicles. Based on this analysis, we create the lags per vehicle in Linköping, Tampere, and Les Mureaux and for the whole fleet in Madrid.

In Table III, we present the performance for dwell time predictions of the different models. Interestingly, there is no universal best model across all sites. In Linköping and Tampere, XGBoost outperforms in RMSE, but RF-GCN achieves the lowest MAE, while the difference in MAE is more significant than in RMSE. This implies that integrating a GNN with an upstream classifier could be the best strategy for these locations. The difference in performance between MLP and GCN, especially in Linköping, underlines the benefits of integrating spatial information, specifically for pilot sites with good data availability and longer, frequently-used routes. In Les Mureaux, the situation is reversed. The tree-based models outperform the other models by a large margin. We perform a more detailed analysis of this difference in the subsequent section. Given the very limited data availability in Graz, it is no surprise that our tested models cannot outperform the mean regressor, as there is not enough data for the advanced models to learn the spatial and temporal correlations. In Madrid, the GCN exhibits better performance than RF-GCN. We can attribute this to an observed degradation in the performance of the RF classifier in detecting zeros compared to the standalone GCN. This finding is intriguing, suggesting that spatial information in this pilot not only helps in accurate dwell time prediction when a shuttle stops but also in detecting stop skips.

In Table IV, we analyze if the results from the dwell time predictions translate to running time predictions. Note that the RF-GCN model has not been used for running time predictions, as zero observations are impossible here. Looking
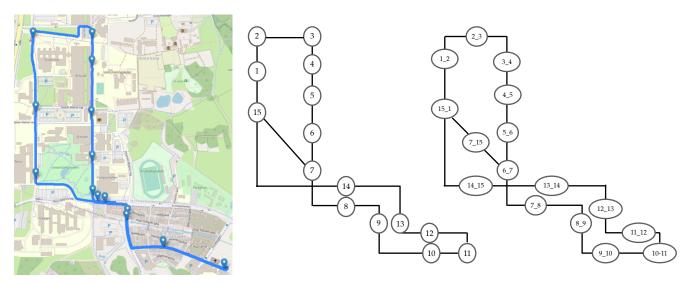
Fig. 2: Construction of the graph for GNN predictions from the original route (left) to the final graph for dwell time (middle) and running time predictions (right)

TABLE II: Results for Running Time Predictions Depending on Lags (Results are (RMSE/MAE)

| | Lag Model | | XGBoost | |
Site	All vehicles	Per vehicle	All vehicles	Per vehicle
Linköping	22.10/ 14.47	**20.22/ 11.98**	15.57/ 10.15	**15.28/ 9.76**
Tampere	16.40/ 11.53	16.40/ **11.28**	12.41/ 8.94	**12.19/ 8.58**
Les Mureaux	**21.60/ 11.23**	21.88/ 11.42	18.73/ 10.68	**18.47/ 10.41**
Madrid	**29.14/ 18.85**	29.47/ 19.40	**21.09/ 14.56**	21.21/ 14.57

TABLE III: Results for Dwell Time Prediction

		Lag	Mean	LR	RF	XGBoost	MLP	GCN	RF-GCN
Linköping	RMSE	15.65	12.28	11.43	11.14	**10.57**	11.63 ± 0.03	10.87 ± 0.03	10.66 ± 0.06
	MAE	9.15	8.78	7.63	7.03	6.59	6.92 ± 0.06	5.92 ± 0.04	$\mathbf{5.66} \pm 0.06$
Tampere	RMSE	17.10	13.28	13.07	12.69	**12.61**	13.73 ± 0.07	13.70 ± 0.10	12.73 ± 0.05
	MAE	10.57	9.73	9.46	8.75	8.55	8.20 ± 0.04	8.13 ± 0.02	$\mathbf{7.65} \pm 0.01$
Les Mureaux	RMSE	12.46	12.26	10.06	**8.06**	8.24	10.08 ± 0.13	9.79 ± 0.18	8.52 ± 0.06
	MAE	5.41	8.51	5.85	**3.30**	3.46	5.45 ± 0.11	5.23 ± 0.15	3.98 ± 0.13
Graz	RMSE	15.84	10.88	11.27	**10.52**	10.66	12.71 ± 0.24	12.92 ± 0.17	11.21 ± 0.09
	MAE	11.59	**7.13**	7.16	7.89	7.94	8.76 ± 0.27	8.77 ± 0.10	7.24 ± 0.08
Madrid	RMSE	22.83	19.41	18.49	18.03	**16.19**	16.90 ± 0.17	17.06 ± 0.2	17.18 ± 0.09
	MAE	6.58	9.71	9.27	7.77	6.20	4.51 ± 0.38	$\mathbf{4.21} \pm 0.01$	4.62 ± 0.10

at Table IV, we can see a quite different picture than in Table III. This is not unexpected, as running time and dwell time have different characteristics. Here we can see that for almost all of the sites, the tree-based methods RF and XGBoost are superior, with XGBoost being the best model for Tampere and Graz, RF being the best for Madrid, and the performance being almost equal for Les Mureaux. Only for Linkoping, the ANN models outperform, with both the MLP and the GCN having comparable performance. For Tampere, the difference between the MLP and XGboost is only half a second, while the difference between Graz, Madrid, and Les Mureaux is around 2 seconds.

It can also be seen that the MLP achieves better results than the GCN for all sites. This is different than what has been observed with using GCNs for other traffic-related tasks, such as traffic flow, where in general, the spatial modeling process of GCNs improves performance [24]. A reason for this result could be the deployment in low-traffic zones or the regulatory speed limit for automated shuttles, leading to less variation in the running times.

A. Assessment of dwell time predictions

The previous section showed that the optimal choice between deep learning and tree-based methods for predicting dwell times is site-dependent, with no universally best model. In light of these findings, a deeper understanding is needed to determine the conditions that favor one approach over the other.

In Figure 3, we depict the dwell time distribution in Linköping. Linköping is unique in having the longest ongoing

TABLE IV: Results for Running Time Prediction

		Lag	Mean	LR	RF	XGBoost	MLP	GCN
Linköping	RMSE	20.22	15.87	15.90	15.34	**15.28**	16.33 ± 0.07	16.33 ± 0.07
	MAE	11.98	10.47	11.04	9.90	9.76	**9.50** ± 0.03	9.53 ± 0.03
Tampere	RMSE	16.40	12.43	12.42	12.34	**12.19**	12.91 ± 0.14	12.97 ± 0.10
	MAE	11.28	8.91	8.87	8.87	**8.58**	9.07 ± 0.16	9.31 ± 0.19
Les Mureaux	RMSE	21.88	29.96	21.42	**18.39**	18.47	23.68 ± 0.37	25.54 ± 0.92
	MAE	11.42	17.61	13.63	10.61	**10.41**	13.64 ± 0.18	14.81 ± 0.57
Graz	RMSE	20.25	15.99	16.28	17.03	**14.76**	17.74 ± 0.31	18.52 ± 0.98
	MAE	15.81	12.97	12.95	13.74	**11.59**	13.38 ± 0.22	14.09 ± 0.59
Madrid	RMSE	29.14	24.01	22.96	**21.07**	21.09	22.71 ± 0.14	22.73 ± 0.07
	MAE	18.85	17.87	16.84	**14.31**	14.56	16.23 ± 0.09	16.47 ± 0.11

pilot and route, collecting nearly five times more observations than the other sites. The dwell time distribution here is distinctly multimodal. A mode at zero seconds indicates instances where a shuttle stop is skipped. The remaining modes suggest times when the shuttle stops and passengers are alighting or boarding. There is a variation of the dwell times between 20 and 40 seconds, indicative of a frequently used service. In Figure 3, we can observe a difference in the performance of the prediction models. Figure 3a illustrates that XGBoost struggles to categorize the majority of zero dwell times and overlooks the third mode at 35 seconds. In contrast, our hybrid RF-GCN (3b) demonstrates a better fit. The Random Forest classifier effectively classifies if a shuttle is stopping or skipping a stop, and the GCN model fits the multimodal dwell time distribution for cases when passengers are alighting and boarding. Remarkably, by incorporating spatial information, this model even differentiates between the two primary dwelling times of approximately 25 and 35 seconds.

On the other hand, the benefits of tree-based methods are evident in the case of the Les Mureaux pilot site (Figure 4). In Les Mureaux, we can notice the automated nature of the service with two main dwell times of 0 and 25 seconds for skipping and stopping, respectively. The data from this site suggests a default dwell time of 25 seconds that extends only when necessary. As prior research has shown [25], neural networks are biased towards overly smooth solutions. In contrast, decision trees, learning piece-wise constant functions, avoid this pitfall. This characteristic is evident in Figure 4, where we can observe the superiority of the Random Forest when fitting to a non-smooth target function.

Therefore, we can conclude that the relative advantage of using a GNN depends on the specific characteristics of the shuttle service. In cases with high data availability and a frequently-used service with a long route, the GNN seems superior. Yet, the random forest remains relevant in detecting the skipped stops. However, in scenarios where the dwell time is largely automated and does not significantly vary with passenger density, the prediction challenge simplifies to a classification task: Is the shuttle stopping or not? Here, the tree-based methods show a clear advantage.

B. Evaluation on journey

So far, our focus has been on evaluating model performance over individual segments. However, in real-world service deployment, the cumulative AT prediction, formed by aggregating individual segment predictions, is more important. Hence, for each pilot site, we sample five journeys from the test set and assess the accumulated error along the journey starting from stop number 1. We illustrate the results for the three sites with the longest routes, Linköping, Les Mureaux, and Tampere, in Figures 5-7. Solid lines depict average accumulated errors, while the shaded regions represent the range between maximum positive and negative errors across the sampled journeys.

Linköping has the longest route of the pilot sites, with a total of 15 stops. In Figure 5, stop number 7 corresponds to a 15-minute ahead, and stop number 15 to a 35-minutes ahead prediction. The depicted models (RF-GCN, XGBoost, and the Mean regressor) overestimate the total travel time, but RF-GCN shows the least average error with a maximum of 30 seconds. In the 15-minutes ahead predictions at stop number 7, we can observe an error of a maximum of 60 seconds and 35 minutes ahead of around 100 seconds. XGBoost still shows a lower average error than the Mean regressor, yet, toward the end of the route, the accumulated errors display worse maximum values.

In Les Mureaux (Figure 6), the full journey is shorter, spanning six stops, which is equivalent to a 20-minute travel time. Here we observe that the accumulated errors do not exceed 70 seconds in over- and 60 seconds in underestimating.

The interesting finding in the Tampere (Figure 7) site is that although XGboost outperforms the GCN on running time prediction, the accumulated predictions by the RF-GCN/GCN combination are more accurate. The average error is close to zero, and the maximum error stays below 100 seconds, while XGBoost is, on average, underestimating the running time by up to 50 seconds on average and 150 seconds in the worst case. These findings suggest that future work needs to rethink how they evaluate their models, as most existing work does not consider aggregation errors.

Since we are aggregating the predictions of the dwell and running time prediction models, the question arises of how the errors accumulate between those two models and which one is driving the difference in performance in Figures 5-7. In Table
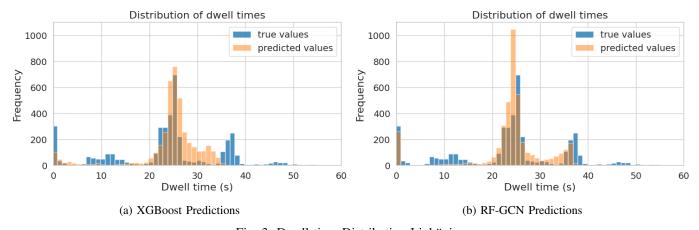
(a) XGBoost Predictions

(b) RF-GCN Predictions

Fig. 3: Dwell time Distribution Linköping

(a) Random Forest Predictions
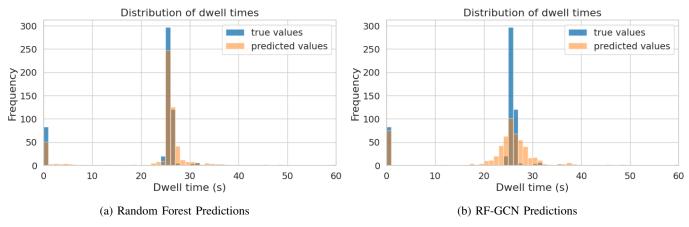
(b) RF-GCN Predictions

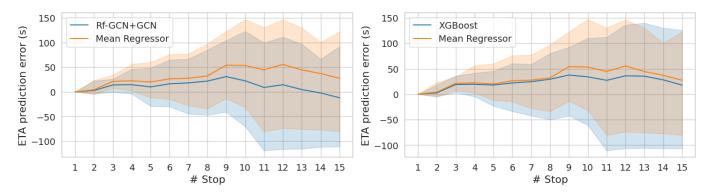Fig. 4: Dwell time Distribution Les Mureaux

Fig. 5: AT Prediction Linköping

V, we calculate the average accumulated absolute error over the entire journey resulting from the dwell and the running time prediction, respectively. The performance of the mean regressor on Linköping and Tampere reveals that the error originating from the dwell time prediction has a larger share of the total travel time error than the one from the running time model. Similarly, we can identify the reason for the superior performance of the GCN in the accumulated travel times, as it decreases the error in dwell time prediction significantly while the error in running time prediction stays comparable. More specifically, if we compare the decrease of errors of

the best model compared to the mean regressor, we observe a decrease of 35%, 26%, and 61% in dwell time error for the three pilot sites and a decrease of 17%, 15%, and 32% for the error originating from the running time predictions. With these results, we stress that dwell time prediction is an important factor for the AT prediction for autonomous shuttles. Especially when they are used in remote, low-traffic areas or are confined to regulatory speed limits, dwell time becomes the most influential factor.
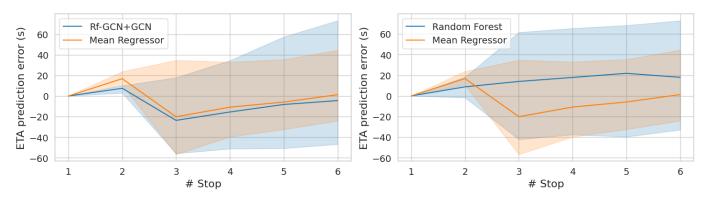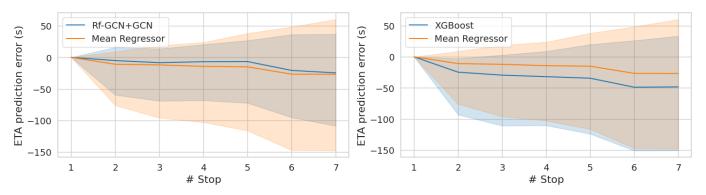
Fig. 6: AT Prediction Les Mureaux

Fig. 7: AT Prediction Tampere

TABLE V: Averaged Accumulated Absolute Error

	Linköping		Tampere		Les Mureaux	
Model	Dwell Time	Running time	Dwell Time	Running time	Dwell Time	Running time
Mean	142	126	76	61	36	72
XGB/RF	106	105	73	52	14	49
GCN	93	105	56	57	16	64

C. Discussion

Since the data is gathered during pilot tests, the data availability is limited. Most sites provide only GPS positioning and speed data. Given this constraint, we cannot rely on many of the input features that have been referenced in prior literature, such as passenger data, that have been proven to be the most impactful for dwell time prediction [26]. Moreover, it is reasonable to assume that demand is highly volatile during pilot phases, as some passengers might be merely curiosity-driven, keen on experiencing technological advancement just once. In some pilot sites, such as Tampere, the shuttle is skipping the majority of the stops due to limited demand. This variability further complicates dwell time predictions. Given that the dwell time of zero emerged as a challenge in our prediction models and the error in dwell time prediction had more impact on the overall AT prediction, passenger monitoring promises a great influence on the final AT prediction.

Regarding running time prediction, we also encounter challenges. As autonomous shuttles operate on both shared public roads and private lanes or shared lanes only with pedestrians and cyclists, we cannot draw from standard road traffic speed estimates available from car data. Instead, we are limited to the most recent running time observations from the shuttle. In some test scenarios where only a single vehicle was operational, or operations were stopped due to technical issues, these recent observations can quickly become outdated and, thus, less reliable. Further, local regulations constrain the speed limit of the shuttle, limiting the variance in running times and, therefore, the potential improvement obtained by predictions.

Our key recommendation for practitioners collecting data during pilots is to be mindful of data consistency. Specifically, if ordinary operations are interrupted, data collection should either be paused, or the collected data should be labeled as 'non-ordinary'. Without this information, we cannot guarantee that test runs to check technical feasibility without the intention of transporting passengers are not included in the data set. Furthermore, any changes to the settings of the shuttle, such as dwelling settings, the sampling rate of the GPS device, or acceleration/deceleration behavior, should be documented. This would greatly improve data pre-processing and the evaluation of predictions for regular operations post-pilot.

VII. Conclusion

This work studied the impact of autonomous vehicles on prediction models within public transport. As autonomous transportation emerges, accurate arrival time predictions are vital to gain public acceptance. Given the early stage of this technology, models typically have to be applied to data originating from pilot tests. Hence, we emphasized the importance of careful data collection to practitioners to obtain an estimate of the prediction for regular operations post-pilot. We tested several established machine learning and deep learning models on dwell and running time prediction in the particular setting of autonomous mobility using data from five diverse pilots across Europe. A key finding is the absence of a universally best model for dwell time prediction and that the performance depends on the specific characteristics of the pilot sites. In cases with high data availability and a long, frequently-used route, the hierarchical approach of a random forest classifier combined with a graph neural network is superior. However, in scenarios where the dwell time is mainly automated and does not significantly vary with passenger density, the prediction challenge simplifies to a classification task, and the tree-based models are preferable. Surprisingly, leveraging deep learning architectures that incorporate spatial information for running time predictions failed to yield clear benefits over the tree-based methods. This observation can be partly attributed to the fact that the shuttles are mainly deployed in low-traffic and low-speed zones or are constrained by a regulatory speed limit for automated shuttles. For the final AT prediction, dwell time prediction proves to be the driving component, with a critical factor of the prediction model being able to detect when shuttles skip stops. Encouragingly, the results for the final AT prediction are promising, with maximum errors not exceeding 60 seconds for predictions up to five stops ahead.

Future research directions are threefold. First, as autonomous public transport systems evolve in complexity, research should monitor performance changes in predictions in response to growing passenger demand and potential increases in speed limits. Secondly, there is room to explore other models, such as combining the hierarchical zero-inflated approach into one model with combined optimization, e.g., with a Gaussian mixture model. Lastly, future research should collect and incorporate additional data features into the prediction models. Specifically, passenger data, along with details on shuttle characteristics or operational settings that affect dwell time or acceleration, declaration, and cruising speed, are potentially important data characteristics that influence the prediction of the arrival time of automated shuttles.

Acknowledgements

References

[1] M. Chen, X. Liu, J. Xia, and S. I.-J. Chien, "A dynamic bus-arrival time prediction model based on apc data," *Computer-Aided Civil and Infrastructure Engineering*, vol. 19, 2004. [Online]. Available: https://api.semanticscholar.org/CorpusID:18929025

[2] Z. R. Wall and D. J. Dailey, "An algorithm for predicting the arrival time of mass transit vehicles using automatic vehicle location data," 1998. [Online]. Available: https://api.semanticscholar.org/CorpusID:15659679

[3] M. Sinn, J. W. Yoon, F. Calabrese, and E. P. Bouillet, "Predicting arrival times of buses using real-time gps measurements," *2012 15th International IEEE Conference on Intelligent Transportation Systems*, pp. 1227–1232, 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:11089217

[4] B. Yu, Z. Yang, and B. Yao, "Bus arrival time prediction using support vector machines," *J. Intell. Transp. Syst.*, vol. 10, pp. 151–158, 2006. [Online]. Available: https://api.semanticscholar.org/CorpusID:29548589

[5] J. Li, "Bus arrival time prediction based on random forest," 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:57943904

[6] Y. Lin, X. T. Yang, N. Zou, and L. Jia, "Real-time bus arrival time prediction: Case study for jinan, china," *Journal of Transportation Engineering-asce*, vol. 139, pp. 1133–1140, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:110345685

[7] A. Agafonov and A. Yumaganov, "Bus arrival time prediction using recurrent neural network with lstm architecture," *Optical Memory and Neural Networks*, vol. 28, pp. 222–230, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:203609512

[8] J. Pang, J. Huang, Y. Du, H. Yu, Q. Huang, and B. Yin, "Learning to predict bus arrival time from heterogeneous measurements via recurrent neural network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, pp. 3283–3293, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:116196627

[9] Z. Lingqiu, H. Guangyan, H. Qing-wen, Y. Lei, L. Fengxi, and C. Lidong, "A lstm based bus arrival time prediction method," *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 544–549, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:215738588

[10] N. C. Petersen, F. Rodrigues, and F. C. Pereira, "Multi-output deep learning for bus arrival time predictions," *Transportation Research Procedia*, vol. 41, pp. 138–145, 2019, urban Mobility – Shaping the Future Together mobil.TUM 2018 – International Scientific Conference on Mobility and Transport Conference Proceedings. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352146519304375

[11] J. Ma, J. Chan, S. Rajasegarar, and C. Leckie, "Multi-attention graph neural networks for city-wide bus travel time estimation using limited data," *Expert Syst. Appl.*, vol. 202, p. 117057, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:248074161

[12] M. Chen, J. Yaw, S. I.-J. Chien, and X. Liu, "Using automatic passenger counter data in bus arrival time prediction," *Journal of Advanced Transportation*, vol. 41, pp. 267–283, 2007. [Online]. Available: https://api.semanticscholar.org/CorpusID:110472924

[13] J. Ma, J. Chan, G. Ristanoski, S. Rajasegarar, and C. Leckie, "Bus travel time prediction with real-time traffic information," *Transportation Research Part C: Emerging Technologies*, vol. 105, pp. 536–549, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0968090X18309082

[14] N. Singh and K. Kumar, "A review of bus arrival time prediction using artificial intelligence," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 12, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:247969151

[15] E. Antypas, G. Spanos, A. Lalas, K. Votis, and D. Tzovaras, "Estimated time of arrival in autonomous vehicles using gradient boosting: Real-life case study in public transportation," in *2022 IEEE International Smart Cities Conference (ISC2)*, 2022, pp. 1–7.

[16] A. B. P, S. R. M, and R. Sumathi, "Bus dwell time forecasting using machine learning models," in *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)*, 2023, pp. 1156–1161.

[17] S. Rashidi, P. Ranjitkar, and Y. Hadas, "Modeling bus dwell time with decision tree-based methods," *Transportation Research Record*, vol. 2418, no. 1, pp. 74–83, 2014. [Online]. Available: https://doi.org/10.3141/2418-09

[18] A. B P, S. Ranganathaiah, and S. H S, "Bus travel time prediction: A comparative study of linear and non-linear machine learning models," *Journal of Physics: Conference Series*, vol. 2161, p. 012053, 01 2022.

[19] L. Zhu, S. Shu, and L. Zou, "Xgboost-based travel time prediction between bus stations and analysis of influencing factors," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1–25, 07 2022.

[20] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 785–794. [Online]. Available: https://doi.org/10.1145/2939672.2939785

[21] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2017.

[22] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 10 2001.

[23] https://show-project.eu/, 2023, accessed: 2023-07-31.

[24] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv: Learning*, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:3508727

[25] L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on typical tabular data?" in *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., vol. 35. Curran Associates, Inc., 2022, pp. 507–520. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/0378c7692da36807bdec87ab043cdadc-Paper-Datasets_and_Benchmarks.pdf

[26] S. Rashidi, S. Ataeian, and P. Ranjitkar, "Estimating bus dwell time: A review of the literature," *Transport Reviews*, vol. 43, no. 1, pp. 32–61, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0144164722004342