Soft Computing Letters 3 (2021) 100029

Contents lists available at ScienceDirect

Soft Computing Letters

journal homepage: www.elsevier.com/locate/socl

Bus journey simulation to develop public transport predictive algorithms

Thilo Reich [a,*], Marcin Budka [a], David Hulbert [b]

[a] Department of Computing and Informatics, Bournemouth University, Fern Barrow, Poole, BH12 5BB, United Kingdom
[b] Passenger Technology Group Ltd., 65 Seamoor Road, Bournemouth, BH4 9AE, United Kingdom

ARTICLE INFO

Keywords:
Public transport
Arrival time prediction
Data simulation
Data quality
Machine learning
Deep learning

ABSTRACT

Encouraging the use of public transport is essential to combat congestion and pollution in an urban environment. To achieve this, the reliability of arrival time prediction should be improved as this is one area of improvement frequently requested by passengers. The development of accurate predictive algorithms requires good quality data, which is often not available. Here we demonstrate a method to synthesise data using a reference curve approach derived from very limited real world data without reliable ground truth. This approach allows the controlled introduction of artefacts and noise to simulate their impact on prediction accuracy. To illustrate these impacts, a recurrent neural network next-step prediction is used to compare different scenarios in two different UK cities. The results show that a realistic data synthesis is possible, allowing for controlled testing of predictive algorithms. It also highlights the importance of reliable data transmission to gain such data from real world sources. Our main contribution is the demonstration of a synthetic data generator for public transport data, which can be used to compensate for low data quality. We further show that this data generator can be used to develop and enhance predictive algorithms in the context of urban bus networks if high-quality data is limited, by mixing synthetic and real data.

1. Introduction

Cities around the world are trying to shift personal traffic to public transport to reduce congestion and environmental impact. A crucial part of such a strategy is to make public transport as convenient as possible. Bus passengers often rely on Real-Time Passenger Information (RTPI) systems at bus stops, online and in mobile apps. These RTPI systems can be unreliable [1] which is inconvenient for passengers. In general, passengers assign different priorities to certain aspects of public transport. Reliability and safety are considered the two most important [2].

The importance of making especially buses as attractive as possible in comparison to private vehicles is highlighted in the historical statistical records. In the UK, 4.8 billion bus trips were made in 2018/19, accounting for 58% of all public transport journeys [3]. These journeys amounted to 27.4 billion km travelled and saved approximately 96 million tonnes of CO_2 [4]. However, since 1985, bus travel has been steadily decreasing by a total of 0.7 billion journeys. As other public transport modes such as trains in most areas cannot be a replacement for local bus services, this suggests that a larger share of passengers opt for private vehicles. This is mirrored in the continuous upward trend of car traffic on British roads [3]. To encourage potential passengers to use public transport, it is crucial to make it as attractive as possible to reverse the above trends, ultimately having a positive impact on the environment as well as congestion levels in urban settings. However, the mentioned data are pre-pandemic, thus the long-term impact of the pandemic on public transport cannot currently be anticipated.

Other studies also highlighted the importance of accurate Estimated Time of Arrival (ETA) predictions to improve customer experience [5]. Many public transport providers have developed mobile apps, which give ‘live’ positions of vehicles. Passengers can use such technology to decide when to leave the house to catch a bus without having long wait times at a bus stop. However, we previously noted the latency of this information caused by delays of wireless network infrastructure and the fact that the data in our operational area passes through a number of 3rd party systems [6]. Therefore, the RTPI system might suggest a vehicle is further away than it is in reality. This could cause a passenger to miss a bus and thus unnecessarily inconvenience them. In Bournemouth, one of the two cities used as an example in this study, the latency of the internet-based ‘live position’ is approximately 30–40 s. To alleviate this issue, we have proposed a short-horizon prediction which will be useful in the further development of ETA and long-term predictions, and in bringing the ‘live’ locations closer to reality. The commonly deployed Automatic Vehicle Location (AVL) systems [7], could supply data for such approaches.

To compare any potential model, the assessment of their performance is of crucial importance, this has to be reported in a way

* Corresponding author.
E-mail address: treich@bournemouth.ac.uk (T. Reich).

https://doi.org/10.1016/j.socl.2021.100029
Received 13 September 2021; Received in revised form 8 November 2021; Accepted 22 November 2021
Available online 5 December 2021

that allows to replicate and compare the results. However, this is not possible in all cases as some authors report relative errors [8–10] and no consistency in the reported parameters can be distinguished. The precondition for all machine learning algorithms should be verifiable, and the RoyalSociety's report highlights this as a central feature [11]. This has also been recognised in the healthcare sector where guidelines for the development and reporting of predictive models exist [12]. The difference in standards might be explained because ETA predictions do not affect the health or safety of a passenger and a spurious algorithm might at most cause inconvenience rather than physical harm. However, for an operating company, this might cause a loss of revenue through a decline in patronage, and the society as a whole might be subjected to more congestion that could simply be reduced by providing accurate ETA predictions. Furthermore, the doctrine of science is replicability. The reproducibility crisis is most prominently known from psychological research [13] however due to its notoriety, it has been actively addressed [14]. It has also been identified as a problem in 'harder' sciences such as biomedicine [15] and also artificial intelligence [16]. Although results gained from machine learning techniques might be considered hard evidence, because the final model is based on mathematical concepts, they often suffer from similar problems as seen in psychology where the research is often subjective to the researcher. The similarities between the two fields are that the findings cannot usually be explained due to the 'black box' effect. The field of psychology has now started to apply lessons from problems seen in machine learning research [14]. A suggested way of addressing such problems is meta-science that could shed light on the true accuracy of findings [17]. However, this relies on comparable measurements of accuracy, which is not found in a large proportion of the public transport literature. Therefore, comprehensive standards of reporting are urgently needed in the field of predictive bus transportation research. This as a consequence poses the issue that high-quality data is required to develop good predictive models. We and other researchers have highlighted that data quality issues need to be considered in the context of public transport research [6,18–20]. Therefore, in this study we demonstrate a method to synthesis bus journeys based on limited and low quality data. This allows on the one hand to generate a hybrid dataset to develop models from. On the other hand it has the potential to be used to generate synthetic datasets that can be used for benchmarking in an attempt to combat the highlighted replicability issues faced by public transport research.

In our data, a notable lack of quality hampers the development of predictive algorithms. The quality issues include the lack of clear journey identification, linkable to a timetable, artefacts such as gaps in recordings, falsely reported line numbers, and direction of travel (inbound vs. outbound). These quality issues make it impossible to develop accurate predictive algorithms. Unfortunately, the simplest solution of recording high-quality historical data is not feasible due to closed source data collection by 3$^{\text{rd}}$ party companies. To address this issue, this study describes a reference curve-based synthetic data generator, which bases its assumptions on limited real-world data. This allows to test algorithms in a controlled environment and enables the injection of user-defined artefacts into the dataset to test their effect on prediction quality. We also show that mixing real and synthetic data improves the prediction accuracy.

2. Background

Methods for ETA prediction can include simple historical averages or be based on statistical models. However, due to the complexity of the ETA prediction, machine learning methods have become increasingly popular [21]. In recent years, artificial Neural Networks (NN) have revolutionised a number of other domains. Therefore, NNs should be expected to have similar potential when applied to bus ETA prediction problems. A comprehensive review specifically investigating NN applications in public transport [22] found that only 16% (12) addressed ETA of buses, whereas the rest of the studies applied the technique to other modes of transport. This suggests that the area of bus ETA prediction using NNs might be underrepresented in the context of public transport research. This relative absence of NNs to predict bus ETA is striking as NNs have revolutionised other areas of data science such as image and speech recognition [23,24].

The challenge of all machine learning approaches is to fine tune the model parameters, one solution is to use genetic algorithms [25] to optimise machine learning algorithms inspired by nature. Several innovative variations have been demonstrated in the recent literature, such as an algorithm inspired by the mating of red deer populations [26], or the simplification of parameter search with a simplified metaheuristic [27]. The same authors also demonstrate methods applicable to supply chain management using the Taguchi method to outperform conventional genetic algorithms [28] as well as the potential use of blockchain algorithms in the management of supply chains [29], additionally they show applications to predict photovoltaic electricity generation [30] as well as bioremediation [31].

Nowadays, the majority of buses have onboard AVL systems, which are equipped with GPS sensors and transmit the location of the bus at frequent intervals, typically ranging between 20 and 60 s. The availability of vehicle locations are the basis for any ETA prediction and are accessible through the AVL system and do not necessarily need any additional investment in static sensors.

The biggest hurdle in developing machine learning solutions generally is the difficulty to acquire enough good-quality data to develop a useful algorithm. In some fields, this has led to the use of simulated data ranging from medicine [32] to geophysics [33]. Regarding public transport journey simulation, the literature is scarce. Some examples related to bus data simulation include bus platooning [34] as well as traffic simulation [35]. However, to the best of our knowledge, no study has investigated the use of simulated data to train a next step prediction model for urban bus networks. In many areas of machine learning research, benchmark datasets are common [36]. These allow researchers to objectively compare algorithms against each other. This is missing in the field of urban bus networks. Therefore, the presented data generator could allow the generation of a standardised benchmark dataset that could lay the foundation for further research in public transport.

3. Real-world data processing

3.1. Data collection

Data is accessible via the infrastructure of our collaborators, and two British cities have been selected with the largest number of vehicles and access to recorded travel data. AVL data was collected from two different bus operators from Reading (UK) line 17 and Bournemouth (UK) line 1 (Fig. 1). Each vehicle transmits its position approximately every 40 s, which is recorded by the company providing the Electronic Ticketing Machines (ETMs) with the integrated AVL-system. Due to data handling by several independent entities, only a limited amount of information is transmitted. The available data are:

- Timestamp
- Position (latitude and longitude)
- Line number
- Direction (outbound or inbound)

For the Bournemouth operator, it became apparent that the transmitted directions are often incorrect and so are the line numbers when a vehicle changes its line during an operational run. The data collected in Reading had a better integrity with reliably transmitted direction, thus simplifying the data processing steps. Based on this limited information, it is not possible to match a vehicle to a timetable corresponding to the journey it is currently serving. A journey is a specific trip found in the

Fig. 1. Location of both example cities and the journey shape used for all experiments. The line 1 in Bournemouth is shown yellow and the line 17 in Reading in blue. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

timetable of a bus line, e.g., the outbound 9 AM service 1. In contrast, a route pattern (also referred to as 'shape') is the route as travelled on the road, which can vary slightly for each journey for the same bus service. In the example of line 1 in Bournemouth, there are several patterns which can include different starting points along the route, resulting in shorter overall journeys or slightly different routes. In both cities, reliably matching a vehicle directly to a specific route pattern is not possible as the unique route pattern identifiers were not accessible to us. Therefore, one route pattern for each city was arbitrarily selected and used to generate synthetic data, which is an acceptable approach as in the selected cities the differences between patterns are negligible.

3.2. Identifying route sections for filtering

The bus route used in Bournemouth is line 1, starting in the town centre towards Christchurch (Fig. 1). The complete route shape includes longer journeys and therefore needs to be truncated. In the second example of Reading line 17 was used, which can have up to 90 different route patterns per direction with different runtimes and minor variations in route shapes (Fig. 1). Additionally, a complicating factor is that the route follows a one-way system in the city centre, meaning that the routes are different depending on the served direction. Therefore, a two-pronged approach was used. To initially filter journeys that were too far away from the shape, all available shapes for both directions were combined to a template shape. Any journey outside a radius of 3 x the mean distance to the template shape was excluded. The final filtering with the ability to enforce the direction was done using an arbitrarily selected route pattern from the many different patterns available for each line covering the entire length of the route. In the case of Reading these route patterns are mostly identical, however, in Bournemouth the patterns can be very different. We have described these issues previously [6].

3.3. Identification of individual journeys

Due to the lack of explicit journey identification, a heuristic approach was used to separate individual journeys that will then be used as a basis to generate synthetic data.

Bournemouth operator does not reliably transmit the direction a vehicle currently serves. However, an observation made was that at the end of a journey vehicles stopped transmitting data for a short period of time. Thus, once it reappears in the data stream, a gap in the timestamps can be detected. A new journey was defined as a time gap of more than 15 min. If such a gap is detected, it is assumed a new journey has started.

Reading operator reliably reports the direction of travel, making the identification of an individual journey easier. Furthermore, vehicles tend to serve the same line and do not change lines between runs, by selecting a single direction, large gaps in transmission timestamps can be observed, making the separation of journeys accurate.

3.4. Trajectory generation

It is assumed that the vehicles follow the identified outbound journey shape. This allows us to represent a journey as a trajectory which is the distance travelled along the route shape. Using such a trajectory, a journey can be represented in two dimensions based on the distance travelled and the run time from the start of the journey.

3.5. Additional processing steps

To ensure a clean dataset, repetitions at the start where the vehicle did not move further than 10 m were removed and a journey is assumed to start once the vehicle has moved further than this threshold. The journey was presumed to have ended as soon as it had reached its maximum trajectory.

4. Synthetic data generation

The data generation process uses a heuristic data-based approach to generate synthetic journeys. This process is broken down into several sub steps:

- The interpolation of the route shape as the reported points are not evenly distributed along the route.
- The identification of the normal run time for a journey is based on historical data, which also allows the identification of delays.
- The probability-based simulation of the delays.

The above steps are described in detail in the following subsections.

4.1. Interpolating the journey based on the route shape

A synthetic journey is generated based on future timetables. To avoid all vehicles starting at the same point, a time offset is added to the start time of the timetable, which is a random number between 0 and 40 s (the transmission interval). This is added to the scheduled start time. The distance that should be offset is then calculated by multiplying the offset by the average speed observed in the real world data 8 m/s (30 km/h). The timestamps are then interpolated to a user-defined interval — 40 s in the presented example. Calculating the time difference between two subsequent stops on the route segment gives the overall runtime. This can be divided by the transmission frequency of 40 s to give the number of transmissions expected on this route section. By assuming the vehicle travels at a constant speed, the progress along the shape can be estimated and the coordinates of the shape at the transmission points can be extracted. However, the coordinates of the reference journey pattern are not equidistant; the distances between consecutive reported locations vary between 6 m and 100 m. Therefore, interpolation solely based on the shape would give very different speeds depending on the road shape. This is avoided by generating an interpolation based on the distance along the route. The closest calculated distance of the shape coordinates is used to calculate the difference between the interpolation coordinate and the shape coordinate. If this distance is greater than 5 m, the two neighbouring points on the shape are used to interpolate the positions between these two coordinates to make the data more realistic. This does not account for variations in the speed or the curvature of the earth, but as the distance is at most 100 m, it is a reasonable omittance. Additionally, it appears that wider gaps are found on straight road sections and the frequency increases in meandering sections, making the proposed approach a good compromise.

4.2. The problem of determining delays

As arrival times at bus stops are not recorded, it cannot be determined whether a vehicle was running on time or was delayed. An additional difficulty is that the journey times vary and depend on the time of the day and weekdays. This variation in timetabled runtime compensates for the expected traffic status. TomTom, a location technology company, records congestion characteristics for different cities based on consumer GPS data. The data for Bournemouth indicates the percentage of delay that needs to be added to a journey at a certain time of day. The maximum in Bournemouth is on a Wednesday afternoon with an expected 71% increase in travel time (pre-pandemic) [37].

Most times of the day, the timetable overestimates the travel time compared to the expected time based on TomTom's data. However, it needs to be kept in mind that the vehicles travel between Bournemouth and Christchurch and the data only accounts for Bournemouth. Furthermore, stops to let passengers board or debark are not considered in the TomTom dataset. This means the timetable accounts for expected variations in traffic conditions and thus cannot be used to simulate vehicle delays.

Another avenue explored was the use of Google services to predict delays based on consumer data, which was not possible as buses travel in bus lanes, making the route very different from a prediction based on Google Maps.

4.2.1. Probability based simulation of delays

By assessing all journeys within the real-world dataset by weekday and hours of day, a reference trajectory can be derived. This reference trajectory is simply the mean trajectory of all observed journeys (Fig. 2(a)). As a result, the outliers are removed and the reference curve represents the baseline of a 'normal' journey (Figs. 2(b) and 2(c)). This allows to calculate the probability that a journey will be delayed or early for every time of each week day. Reference curves were generated using a centred moving 3 h window except for the first and last hour where a truncated window was used. This gives the advantage that the time dependency of delays is simulated, meaning that a vehicle following a delayed bus will most likely also be delayed, thus approximating the delay propagation along a single line.

4.2.2. Journey generation

To generate a journey, the timetables of one week are queried and used as a template. The reason for this approach is that although the timetables for Bournemouth are available until the end of the current calendar year, this is not the case in Reading where only one week is available. As the timetable normally does not drastically change within the same year, this is a justifiable approach. Subsequently, the reference curve queried and the following relevant data points are extracted:

- The mean reference trajectory.
- The standard deviation as well as 95% confidence intervals.
- The probabilities of delayed or early arrival with respect to the reference curve (Fig. 2).

4.2.3. Delays

Based on the reference curve, the probability of a journey being delayed or early can be calculated. Whether a journey is delayed is decided by sampling from a normal distribution for each entry of the reference table, a random number r is generated and stored in a probability list $\{r_o \dots r_n\}$. These parameters double as a modification parameter to generate the delay or time gain. To remove variations of the list of probabilities, a Savitzky–Golay filter is applied with a window of 7 and a polynomial order of 3. A decision whether a vehicle will be on time, early or delayed is made based on the smoothed probability list. A vehicle will arrive early if $r < p_{early}$. If $p_{early} < r < p_{early} + p_{delayed}$ vehicle is delayed. If neither of the conditions is true, the vehicle is assumed to be on time. To simulate the variations in time gained, the initially expected runtime t of the reference curve is calculated as well as the difference of the last position of the reference curve γ. The ratio of expected variation is calculated based on the confidence interval of the reference curve ν. Thus, the progress along the trajectory under the influence of a time gain can be calculated as follows:

$$\nu = (\sigma_i/\gamma_i) * (R = \{^1_0)$$
$$P = P_{i-1} + t - (t \times ((0.9 \times \nu) \times 1.25))$$

Where: ν=volatility, γ=reference, P=position, t =expected time at position

If the next position will be delayed, a random modification factor m is generated by sampling from a beta continuous random distribution (α=1, β=2). This tailed distribution was chosen as it makes large reductions in delay less likely and a vehicle will in most circumstances make up no or very little time. The delay *volatility* is defined as the ratio of the reference curve standard deviation to the reference curve itself multiplied by m. Additionally, the delay of the previous step d_{i-1} is calculated and subtracted from the current delay to prevent an exponential increase in delay. To account for random major changes outside the 'norm' of delay or time gains observed in the real data, GPS *noise* is generated using a uniformly sampled random number R which also acts as a weight of the additional delay. Thus, a position with simulated noise can be described as:

$$\eta = \nu \times (R = \{^1_0 + 1)$$
$$P = P_{i-1} + (t + [(\nu \times m) - d_{i-1} \pm \eta])$$

Where: η= noise to be added, ν=volatility, P=position, t =expected time at next position

If the bus is most likely on time, the probability p of it being on time is used to generate an adjustment towards the reference curve as follows:

$$P = [P_{i-1} + t] - [p \times t]$$

Where: P=position, p = probability a vehicle is on time t =expected time at next position

The generated trajectory is then interpolated to give positions in time intervals of 40 s consistent with the transmission rates of the recorded data.

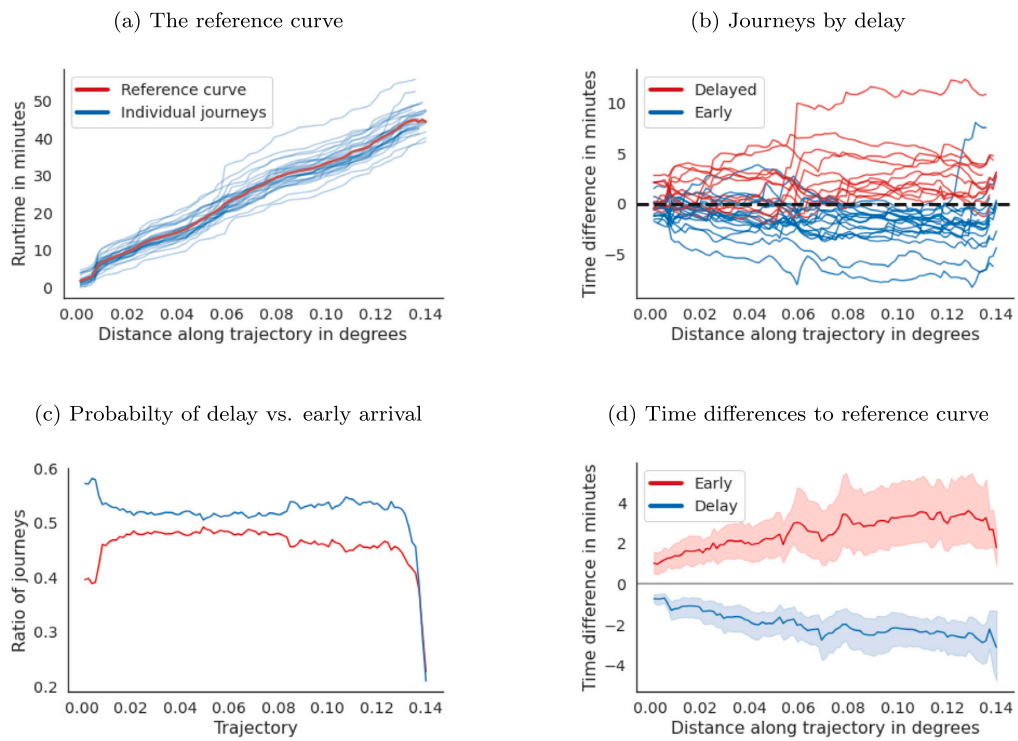

Fig. 2. (a) The historical trajectories of a one day block in Bournemouth (Tuesday 9–12 am). (b) The relative difference from the reference curve along the trajectory. Journeys delayed at more than 60% of the positions are highlighted in red. (c) Probability of travelling early or late on the trajectory. The discrepancy in the sum of the two conditions represents the fraction of vehicles that arrive on time. (c) The average time difference to the reference curve with the uncertainty highlighted. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.3. *Injection of artefacts*

The original data is affected by artefacts caused by the behaviour of vehicles as well as data collection issues. Three noteworthy artefacts have been incorporated into the simulation of the synthetic data and are described below.

4.3.1. *Injection of GPS noise*

GPS recordings are affected by noise which can depend on the surrounding environment, such as high-rise buildings. In the cities used in this study, buildings tend to be low and thus effects due to reflection of the GPS signal are unlikely and have not been observed. To simulate the inaccuracies of the GPS recording, random noise sampled from a normal distribution (mean=0, σ=7) is added to latitude and longitude.

4.3.2. *Injection of repeated locations*

Due to operational reasons, journeys have scheduled buffers to allow vehicles to catch up with the timetable. This means that the vehicle often repeatedly transmits the same location at the start or end of a journey. At the journey start, 83% of the journeys have repeated locations, whereas end-repetitions are seen in 67% of journeys. The number of repeats varies depending on how long a vehicle is stationary. A skew-normal distribution [38] was fitted to both the start and end repetitions and this reference distribution is used to sample the number of repeats at either end of the journey. This artefact is optional and datasets with as well as without have been generated as in theory it is possible to gather journey data only for the journey itself without buffer times at either end.

4.3.3. *Geofencing artefacts*

The original data collected contained characteristic circular patterns. We empirically demonstrated previously [6] that the origin of such characteristic artefacts are the geofencing methods used by some AVL-systems to determine if a vehicle has arrived at a bus stop [6]. Unless the bus has been very close to the stop, the AVL-system ‘snaps’ the real position of the vehicle to a circular geofencing boundary with a radius of 10 m. As this is an unusual artefact, it is generated optionally.

4.4. *Data generation*

For both cities, datasets were generated for 145 days and for three different conditions:

- a journey only with GPS noise,
- a journey with GPS noise and circular artefacts,
- a journey with GPS noise, and start and end repeats.

Additionally, a hybrid dataset was generated for the city of Reading containing 5000 journeys, of which 50% were synthetically generated and the remaining half were taken from the original dataset.

5. Prediction methods

5.1. *Benchmarks*

Two naïve benchmark algorithms were used to compare all models against.

Average speed: This method uses the average speed of a vehicle since the start of its current journey. Thus, it does not reflect any short-term speed variation. The calculated speed is used to interpolate the position of the vehicle from the trajectory of its journey pattern for the next 40 s.

Current speed: This method uses the last three transmitted positions of a vehicle to calculate its current average speed, hence accounting for temporary speed variations. The prediction is made by interpolating the position for the next 40 s from the journey trajectory.

5.2. Target representation

The target was represented as a trajectory, by projecting the coordinates onto the route pattern of a journey. This ensures that inaccuracies locating a vehicle off-route are removed. In practice, this method predicts a number representing the progress along the trajectory with a max of 1, which is the final destination. To illustrate the performance of the model, the trajectory can be decoded into coordinates to allow the calculation of a Haversine distance between the predicted and actual location, which is more intuitive than a loss based on the trajectory. Two variations of this target representation were used: **a.** the unconstrained progress along the trajectory, which could lead to a vehicle appearing to move backwards, **b.** the distance travelled in the next time interval added to the last known position, which enforces a forward prediction.

5.3. Input features

The features included were: coordinates normalised to a bounding box representing the operational area of the bus company, the time delta between consecutive recordings, the elapsed time from the start of the journey, and time embeddings as described below. The input features were min–max normalised.

5.4. Handling of time

The time information was split into its components to make it possible for the algorithms to learn periodic patterns. To achieve this, the timestamp was translated into the minute of the day, the hour of the day, and day of the week. These were embedded in a multidimensional space as detailed in the architecture description 5.6.

5.5. Input windows

A moving window was applied to each journey. The window size was a minimum of 10 data points growing by one time step at a time until the end of the journey. This ensures a realistic simulation of the progress of a journey as would be observed in a real world application.

5.6. Architecture

Two neural networks were used with identical architecture except for the Recurrent Neural Network (RNN) module [39], which was either a Gated Recurrent Unit (GRU) [40] or a Long Short Term Memory (LSTM) network [41]. The time embeddings were learned by the network in a multidimensional space. The dimensions were chosen as half of the possible number of values for each embedded variable. As an example, the hour of the day was embedded in 12 dimensions as the maximum number of hours is 24. These embeddings with a total of 52 dimensions were fed into a linear layer to reduce their dimensions back to the original number of time-based features. The output of the linear layer was concatenated with the remaining input features and fed into either a GRU or LSTM layer followed sequentially by a 1D batchnorm, a linear layer, a leaky ReLU, a second batchnorm and a final linear layer. To ensure the outputs were bounded, a sigmoid function was applied.

5.7. Hyper-parameters

To allow for direct comparison between the models, all training hyper-parameters were kept constant between the two cities. It is appreciated that this might not always yield the best performance but will illustrate the influence of the modifications made on the performance. The variables used were chosen through empirical exploration following the recommendations described by [42]. Each model was trained for 50 epochs using the one-cycle policy [42] with a maximum learning rate of 10^{-1} (Bournemouth) and 10^{-2} (Reading). As a loss function, the mean average error (MAE) was used.

6. Results and discussion

It is crucial to compare predictive algorithms using several different metrics to ensure a balanced interpretation of the results. Furthermore, it has to be kept in mind that in the presented example the two cities are considerably different. The most striking difference is the practice regarding journey shapes. The idea behind a journey shape is that it gives the exact route along the road of a certain journey. This, however, is handled differently by the bus operators. In the example of Reading each journey has an individual shape amounting to 90 shapes a day. These are mostly very similar or identical. In the example of Bournemouth fewer shapes are used, however, the shapes are significantly different in length as well as route, highlighting the need for standardisation of public transport data. As a result, only a subset of the journeys in Bournemouth are similar enough to be simulated in one approach, thus this dataset contains fewer journeys than the dataset generated for Reading (17,115 vs 7839 journeys). These differences have to be kept in mind and are crucial for the interpretation of the results. The median accuracies for mean speed benchmarks in Reading are lower in all datasets compared to the current speed benchmark and are shown in Fig. 3. The current speed benchmark for Bournemouth is comparable to the average speed benchmark. In the example of Reading this is not the case and the current speed benchmark suffers from higher prediction errors compared to the average speed benchmark (Fig. 3). An explanation could be that vehicles in Reading are more likely to stop for brief periods, which is reflected in a 13% increase of standard deviation of the travelling speed compared to Bournemouth. Interestingly, the histogram for the Reading benchmarks shows a peak around 80 m for the dataset with repeated start and ends (Fig. 4). This is explained by the benchmarking method, which uses the last three positions to estimate the average speed. Thus, a vehicle's speed can change from stationary to moving within 120 s or vice versa. Considering this time frame, 80 m/120 s corresponds to an average speed of 24 km/h, which is a realistic prediction for an urban bus network and in accordance with the estimated speed from the mean speed benchmark (Figs. 3 & 4)).

6.1. Perfect journeys

The first set of experiments shows the 'perfect' synthetic journey. These are generated without any of the discussed artefacts and therefore, should represent the simplest prediction problem. Poor performance of both architectures can be observed in the Bournemouth dataset. Both architectures perform virtually identical with a mean error of 63.8 m (σ=55 m) (Fig. 5(a)). This is an accuracy comparable to the benchmarks (current speed: 64.2 m, mean speed: 62.1 m). This underwhelming performance could be explained by the smaller dataset compared to the Reading data, however, a more likely explanation is the variability of the journey shape and routes in Bournemouth, which naturally results in less realistic synthetic data. As a consequence, it is difficult to identify individual journeys from the original data. Furthermore, the data generation suffers from the fact that the vehicles do not follow a consistent route, which would be expected to cause unrealistic synthetic journeys. In contrast, the prediction for Reading performs well with a mean error of 41.5 m (σ=46.5) and 47.5 m (σ=47.2) for the GRU and LSTM respectively (Fig. 5(a)). Both models significantly improve on the error compared to the benchmark (current speed: 68 m, mean speed 50.7 m). As mentioned previously, this dataset contains more journeys per day, however, the most likely explanation of this performance improvement can be attributed to the uniform journey shape, which will reduce errors in the data generation.

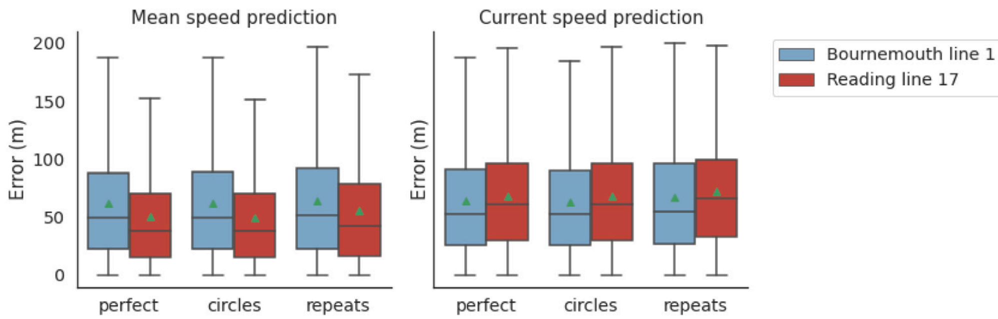

Fig. 3. Boxplot illustrating the prediction errors of the two nïve benchmark algorithms for both cities.

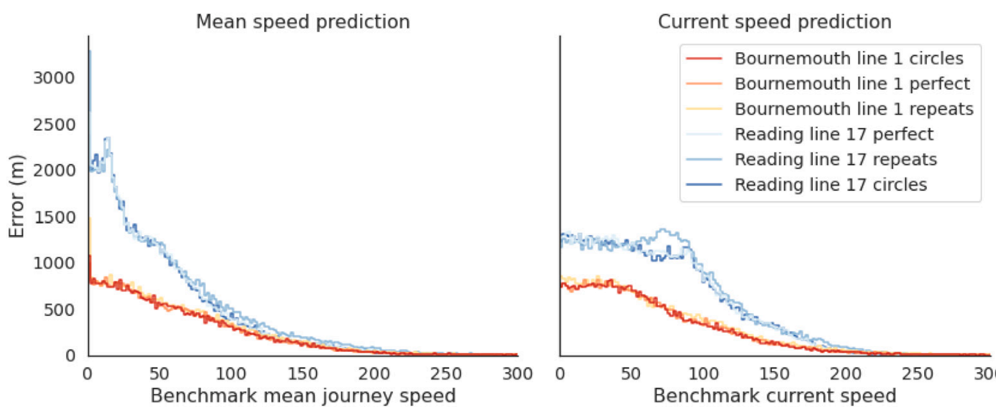

Fig. 4. Boxplot illustrating the prediction errors of the two nïve benchmark algorithms for both cities.

6.2. Ticketing machine artefacts

The introduction of the characteristic circular artefacts into the dataset would be expected to make any prediction more difficult. This is indeed observed in the predictions for Bournemouth. The average GRU performance was reduced by 2.5 m compared to the artefact free journeys. Notably, the performance of the LSTM did not significantly decrease and remained at 63.9 m (Fig. 5(a)). Similar findings were observed in Reading where the mean error of the GRU increased by 5 m. Interestingly, the mean error of the LSTM decreased by 2 m.

6.3. Repeats at start and end

The introduction of repeats at the start and end of the journey did have a strong impact on the prediction performance. The mean prediction error in Bournemouth increased by 5 m and 2 m for the GRU and LSTM, respectively. In Reading, the GRU prediction worsened drastically by 24 m, whereas the LSTM was not affected and remained at 47.8 m (Fig. 5). This is an intuitive response of the LSTM which, due to its ability to forget irrelevant information, is able to focus on the data relevant for the next step prediction.

6.4. Using hybrid data to improve predictions

The described hybrid dataset was used to demonstrate a possible application. As an intuition, it was assumed that the addition of synthetic data, which are cleaner and not affected by uncontrollable artefacts, should improve the overall prediction. When using an unconstrained prediction along the trajectory, this however is not observed and a model trained on purely synthetic or hybrid data performs worse on inference on real data (Fig. 5). This, however, is not the case if the prediction is forced forward as described in Section 4.4. If the prediction space is limited, an improvement in the inference accuracy of networks trained on both the real world dataset can be observed both in the purely synthetic and the hybrid dataset. The largest improvement can be observed if hybrid data were used for training (Fig. 5(b)).

6.5. Discussion of results

The results of this study show that the addition of synthetic data can improve predictive algorithms, which suffer from data quality issues. The use of synthetic data is used in many settings [43], such as healthcare settings to preserve privacy [44] but is also used in the assessment of algorithms such as feature selection methods where the control of features is important [45]. Some authors have also used synthetic data to estimate the upper theoretical limits of predictive algorithms [46]. The generation of hybrid datasets consisting of both real and synthetic data is less common, but examples such as from computer vision exist [47] or for classification problems with heavily unbalanced data [48]. Furthermore, some studies used synthetic data to augment small datasets, for example to improve pandemic datasets and the associated machine learning models [49]. Examples from the field of public transport are rare and mostly focus on optimisation of transport networks and specifically bus routes to minimise delays [50–52]. However, in general, a knowledge gap appears to prevent the combination of simulated data with machine learning algorithms [53], which could be beneficial to improve many areas especially in public transport research. This study demonstrates the use of such hybrid datasets to improve prediction quality. Furthermore, it highlights the lack of framework previously noted by us [54]. A prediction accuracy comparison with the wider literature for this study is not possible as similar research aims to solve different problems. The reason for this is that the research focus regarding short horizon predictions are focused on time frames of >5 min [55,56] or are defined as a distance rather than a time horizon [57]. Shorter prediction horizons are found in the literature but are aimed at predicting different metrics such as speed [58] or the elimination of bus-bunching [59]. As there are, to the author's knowledge, no examples in the literature predicting the position of urban buses in an ultrashort prediction horizon, a comparison with other studies cannot be drawn. Additionally, this study does not claim predictive superiority but demonstrates that the use of hybrid

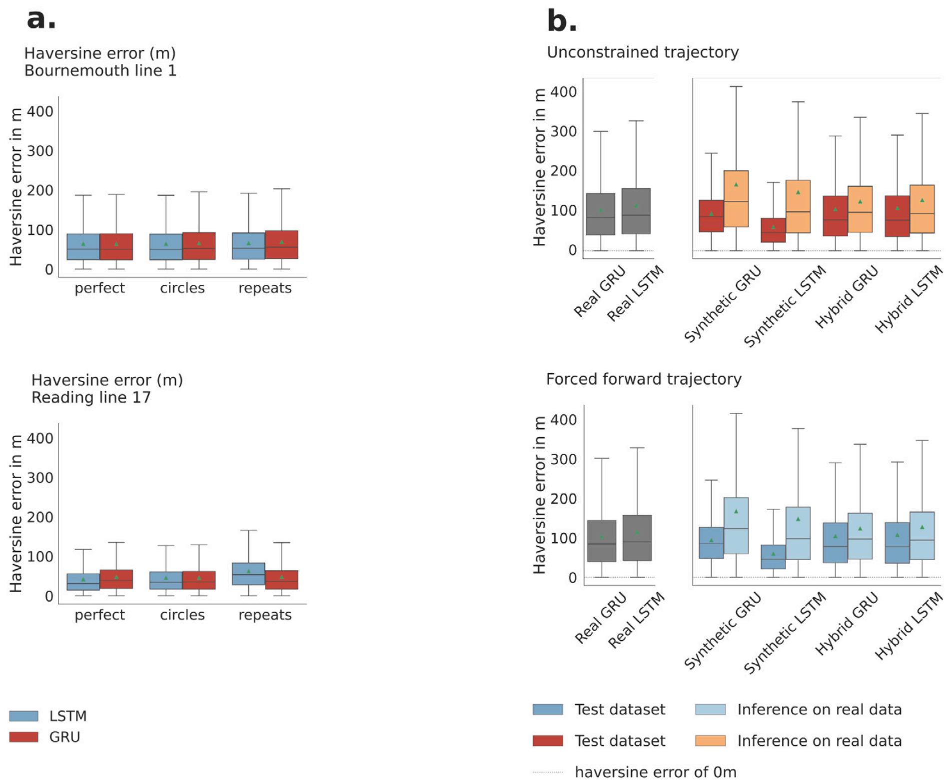

Fig. 5. **(a)** Boxplots for both cities and for each of the dataset and network architecture combinations. It is apparent that the performance in Reading is considerably better and the expected deterioration with the introduction of artefact can be observed. **(b) top:** Boxplots showing the error ranges in meters for the unconstrained networks the grey boxes show a network trained on real data as reference. The red boxes show the error of the holdout portion of the synthetic or hybrid dataset the orange boxes show the inference errors on the real dataset. **(b) bottom:** Boxplots showing the error ranges in meters for the forced forward networks the grey boxes show a network trained on real data as reference. The dark blue boxes show the error of the holdout portion of the synthetic or hybrid dataset the light blue boxes show the inference errors on the real dataset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

data can improve prediction accuracy. This knowledge will be of value to public transport researchers and can be applied to any prediction problem as well as to any model architecture to push the limits of the available data.

7. Conclusion

The importance of making public transport as convenient as possible is self-evident and could help increase passenger numbers and reduce urban congestion and pollution. Reliable predictions of current vehicle position and arrival times play a crucial part in this endeavour. However, this is being inhibited by the lack of reliable data, making any such algorithm development difficult.

Therefore, the described method of generating realistic journeys builds a bridge between the low quality recordable data and the real world. As a result, it is a platform to develop algorithms in a simulated and controlled environment, which can later be deployed in a real world scenario. Additionally, this platform allows simulation of user-specified artefacts as demonstrated by the repetition of positions or geofencing based disturbances. This study has highlighted several areas of improvement for urban bus network data to allow the development of reliable predictive solutions. The most striking observation was that any RNN based predictions in Bournemouth barely outperformed the naïve benchmark. This is due to the varied route shapes and lengths of the same bus line, making generalisation unfeasible. Thus, it can be recommended from a managerial as well as software development point of view that either route shapes should be standardised between the lines or that the lines are subdivided based on their route shapes. This will greatly improve the potential of the data collected and the development of data-based software solutions.

The second observation was that the prediction performance can be improved if the data is as clean as possible. This means that technology providers need to collaborate to ensure the best possible outcome for public transport as a whole. Although geofencing methods to determine the arrival at a stop are useful, the produced artefacts of some systems do have a negative impact on the tested predictive algorithms. Furthermore, an indication whether a vehicle has started or ended a journey will help in the overall prediction accuracy. The differences between the two example cities highlight the need for a national standard if accurate predictions are desired, universally preventing the need to develop a predictive system from the ground up for each city and operational line. This would be a big step forward to an implementation of mobility as a service and would benefit all public transport operators.

The limitations of this study are that the ground truth can only be approximated due to the lack of high-quality data. This, however, is also the driving force behind the demonstrated approach to further advance this research and any other research relying on public transport data, the following key points should be considered for future research:

- Develop a standardised framework to transmit and record public transport data.
- Standardise the use of route patterns to ensure they can be used for data driven applications.
- Develop a benchmarking framework specifically for predictive algorithms in urban bus networks.

In the meantime, until such standardisations become reality, our data generation method described here is a good approximation of reality and a useful tool in simulating effects on urban bus networks.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

No external funding was received for this study.

References

[1] M.M. Salvador, M. Budka, T. Quay, Automatic transport network matching using deep learning, Transp. Res. Proc. 31 (2016) (2018) 67–73, http://dx.doi.org/10.1016/j.trpro.2018.09.053, URL https://linkinghub.elsevier.com/retrieve/pii/S2352146518301273.

[2] G.-J. Peek, M. van Hagen, Creating synergy in and around stations: Three strategies for adding value, Transp. Res. Rec. J. Transp. Res. Board 1793 (1) (2002) 1–6, http://dx.doi.org/10.3141/1793-01, URL http://trrjournalonline.trb.org/doi/10.3141/1793-01.

[3] Department for Transport, Transport Statistics Great Britain 2019 Moving Britain Ahead, Tech. rep., 2019, p. 10.

[4] Department for Business Energy & Industrial Strategy, Greenhouse gas reporting: conversion factors 2019, Research and Analysis (2019) URL https://www.gov.uk/government/publications/greenhouse-gas-reporting-conversion-factors-2019.

[5] R.G. Mishalani, M.M. Mccord, J. Wirtz, Passenger wait time perceptions at bus stops : Empirical results and impact on evaluating real- time bus arrival information, J. Publ. Transp. 9 (2) (2006) 89–106, http://dx.doi.org/10.5038/2375-0901.9.2.5.

[6] T. Reich, M. Budka, D. Hulbert, Impact of data quality and target representation on predictions for urban bus networks, in: 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, IEEE, 2020, pp. 2843–2852, http://dx.doi.org/10.1109/SSCI47803.2020.9308166, URL https://ieeexplore.ieee.org/document/9308166/.

[7] M. Hickman, Bus automatic vehicle location (AVL) systems, in: Assessing the Benefits and Costs of ITS, Kluwer Academic Publishers, Boston, 2006, pp. 59–88, http://dx.doi.org/10.1007/1-4020-7874-9{_}5, URL http://link.springer.com/10.1007/1-4020-7874-9_5.

[8] Z. Junyou, W. Fanyu, W. Shufeng, Application of support vector machine in bus travel time prediction, Int. J. Syst. Eng. 2 (1) (2018) 21–25, http://dx.doi.org/10.11648/j.ijse.20180201.15, URL https://www.tandfonline.com/doi/full/10.1080/21680566.2017.1353449.

[9] J. Li, J. Gao, Y. Yang, H. Wei, Bus arrival time prediction based on mixed model, China Commun. 14 (5) (2017) 38–47, http://dx.doi.org/10.1109/CC.2017.7942193, URL http://ieeexplore.ieee.org/document/7942193/.

[10] L. Meng, P. Li, J. Wang, Z. Zhou, Research on the prediction algorithm of the arrival time of campus bus, in: Advances in Intelligent Systems Research, AISR, Vol. 142, 2017, pp. 31–33.

[11] The Royal Society, Machine learning: the power and promise of computers that learn by example, in: Report By the Royal Society, Vol. 66, The Royal Society, 2017, p. 125, http://dx.doi.org/10.1126/scitranslmed.3002564, URL https://royalsociety.org/~/media/policy/projects/machine-learning/publications/machine-learning-report.pdf.

[12] W. Luo, D. Phung, T. Tran, S. Gupta, S. Rana, C. Karmakar, A. Shilton, J. Yearwood, N. Dimitrova, T.B. Ho, S. Venkatesh, M. Berk, Guidelines for developing and reporting machine learning predictive models in biomedical research: A multidisciplinary view., J. Med. Int. Res. 18 (12) (2016) e323, http://dx.doi.org/10.2196/jmir.5870, http://www.ncbi.nlm.nih.gov/pubmed/27986644.

[13] M. Baker, Over half of psychology studies fail reproducibility test, Nature (2015) http://dx.doi.org/10.1038/nature.2015.18248, http://www.nature.com/doifinder/10.1038/nature.2015.18248 http://www.nature.com/articles/nature.2015.18248.

[14] T. Yarkoni, J. Westfall, Choosing prediction over explanation in psychology: Lessons from machine learning, Perspect. Psychol. Sci. 12 (6) (2017) 1100–1122, http://dx.doi.org/10.1177/1745691617693393.

[15] M. Baker, 1,500 Scientists lift the lid on reproducibility, Nature 533 (7604) (2016) 452–454, http://dx.doi.org/10.1038/533452a, URL http://www.nature.com/doifinder/10.1038/533452a.

[16] M. Hutson, Artificial intelligence faces reproducibility crisis unpublished code and sensitivity to training conditions make many claims hard to verify, Science 359 (6377) (2018) 725–726, http://dx.doi.org/10.1126/science.359.6377.725, URL http://www.ncbi.nlm.nih.gov/pubmed/29449469.

[17] J.W. Schooler, Metascience could rescue the 'replication crisis', Nature 515 (7525) (2014) 9, http://dx.doi.org/10.1038/515009a, URL http://www.nature.com/doifinder/10.1038/515009a.

[18] W. Rasdorf, H. Cai, C. Tilley, S. Brun, H. Karimi, F. Robson, Transportation distance measurement data quality, J. Comput. Civ. Eng. 17 (2) (2003) http://dx.doi.org/10.1061/(ASCE)0887-3801(2003)17:2(75).

[19] S. Robinson, B. Narayanan, N. Toh, F. Pereira, Methods for pre-processing smartcard data to improve data quality, Transp. Res. C 49 (2014) http://dx.doi.org/10.1016/j.trc.2014.10.006.

[20] R. Arbex, C.B. Cunha, Estimating the influence of crowding and travel time variability on accessibility to jobs in a large public transport network using smart card big data, J. Transp. Geogr. 85 (2020) http://dx.doi.org/10.1016/j.jtrangeo.2020.102671.

[21] R. Choudhary, A. Khamparia, A.K. Gahier, Real time prediction of bus arrival time: A review, in: 2016 2nd International Conference on Next Generation Computing Technologies, NGCT, IEEE, 2016, pp. 25–29, http://dx.doi.org/10.1109/NGCT.2016.7877384, URL http://ieeexplore.ieee.org/document/7877384/.

[22] E. Pekel, S.S. Kara, A comprehensive review for artificial neural network application to public transportation, Sigma J. Eng. Nat. Sci. 35 (1) (2017) 157–179.

[23] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, 2014, pp. 1–9, http://dx.doi.org/10.1007/s10107-014-0839-0, URL http://arxiv.org/abs/1409.3215.

[24] R. Yamashita, M. Nishio, R.K.G. Do, K. Togashi, Convolutional neural networks: an overview and application in radiology, Insights Imaging 9 (4) (2018) 611–629, http://dx.doi.org/10.1007/s13244-018-0639-9.

[25] O. Theophilus, M.A. Dulebenets, J. Pasha, Y.y. Lau, A.M. Fathollahi-Fard, A. Mazaheri, Truck scheduling optimization at a cold-chain cross-docking terminal with product perishability considerations, Comput. Ind. Eng. 156 (March) (2021) http://dx.doi.org/10.1016/j.cie.2021.107240.

[26] A.M. Fathollahi-Fard, M. Hajiaghaei-Keshteli, R. Tavakkoli-Moghaddam, Red deer algorithm (RDA): a new nature-inspired meta-heuristic, Soft Comput. 24 (19) (2020) 14637–14665, http://dx.doi.org/10.1007/s00500-020-04812-z.

[27] A.M. Fathollahi-Fard, M. Hajiaghaei-Keshteli, R. Tavakkoli-Moghaddam, The social engineering optimizer (SEO), Eng. Appl. Artif. Intell. 72 (April) (2018) 267–293, http://dx.doi.org/10.1016/j.engappai.2018.04.009.

[28] M.R. Islam, S.M. Ali, A.M. Fathollahi-Fard, G. Kabir, A novel particle swarm optimization-based grey model for the prediction of warehouse performance, J. Comput. Design Eng. 8 (2) (2021) 705–727, http://dx.doi.org/10.1093/jcde/qwab009.

[29] J. Moosavi, L.M. Naeni, A.M. Fathollahi-Fard, U. Fiore, Blockchain in supply chain management: a review, bibliometric, and network analysis, Environ. Sci. Pollut. Res. (July) (2021) http://dx.doi.org/10.1007/s11356-021-13094-3.

[30] N. Ghadami, M. Gheibi, Z. Kian, M.G. Faramarz, R. Naghedi, M. Eftekhari, A.M. Fathollahi-Fard, M.A. Dulebenets, G. Tian, Implementation of solar energy in smart cities using an integration of artificial neural network, photovoltaic system and classical delphi methods, Sustainable Cities Soc. 74 (2021) 103149, http://dx.doi.org/10.1016/j.scs.2021.103149, https://www.sciencedirect.com/science/article/abs/pii/S2210670721004315 https://linkinghub.elsevier.com/retrieve/pii/S2210670721004315.

[31] M. Mohammadi, M. Gheibi, A.M. Fathollahi-Fard, M. Eftekhari, Z. Kian, G. Tian, A hybrid computational intelligence approach for bioremediation of amoxicillin based on fungus activities from soil resources and aflatoxin B1 controls, J. Environ. Manag. 299 (2021) 113594, http://dx.doi.org/10.1016/j.jenvman.2021.113594, URL https://linkinghub.elsevier.com/retrieve/pii/S030147972101656X.

[32] J.M. Huttunen, L. Kärkkäinen, H. Lindholm, Pulse transit time estimation of aortic pulse wave velocity and blood pressure using machine learning and simulated training data, PLoS Comput. Biol. 15 (8) (2019) e1007259, http://dx.doi.org/10.1371/journal.pcbi.1007259.

[33] K.B. Withers, M.P. Moschetti, E.M. Thompson, A machine learning approach to developing ground motion models from simulated ground motions, Geophys. Res. Lett. 47 (6) (2020) http://dx.doi.org/10.1029/2019GL086690, https://agupubs.onlinelibrary.wiley.com/doi/epdf/10.1029/2019GL086690 https://onlinelibrary.wiley.com/doi/abs/10.1029/2019GL086690.

[34] G. Sethuraman, X. Liu, F.R. Bachmann, M. Xie, A. Ongel, F. Busch, Effects of bus platooning in an urban environment, in: 2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 974–980, http://dx.doi.org/10.1109/ITSC.2019.8917041.

[35] Y. Ding, S.I.-J. Chien, N.A. Zayas, Simulating bus operations with enhanced corridor simulator: Case study of new jersey transit bus route 39, Transp. Res. Rec. (1731) (2000) 104–111, http://dx.doi.org/10.3141/1731-13.

[36] P. Ristoski, G.K.D. De Vries, H. Paulheim, A collection of benchmark datasets for systematic evaluations of machine learning on the semantic web, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9982 LNCS, Springer Verlag, 2016, pp. 186–194, http://dx.doi.org/10.1007/978-3-319-46547-0{_}20, URL https://link.springer.com/chapter/10.1007/978-3-319-46547-0_20.

[37] TomTom, Bournemouth traffic report | TomTom traffic index, 2020, Bournemouth Traffic URL https://www.tomtom.com/en_gb/traffic-index/bournemouth-traffic/.

[38] A. Azzalini, A. Capitanio, Statistical applications of the multivariate skew normal distribution, J. R. Stat. Soc. Ser. B Stat. Methodol. 61 (3) (1999) 579–602, http://dx.doi.org/10.1111/1467-9868.00194.

[39] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning internal representations by error propagation, Read. Cogn. Sci. A Perspect. Psychol. Artif. Intell. (V) (2013) 399–421, http://dx.doi.org/10.1016/B978-1-4832-1446-7.50035-2.

[40] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder–decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, Vol. 4, (January) Association for Computational Linguistics, Stroudsburg, PA, USA, 2014, pp. 1724–1734, http://dx.doi.org/10.3115/v1/D14-1179, URL http://arxiv.org/abs/1409.3215 http://aclweb.org/anthology/D14-1179.

[41] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780, http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[42] L.N. Smith, A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay, 2018, pp. 1–21, arXiv CoRR abs/1803.09820 URL http://arxiv.org/abs/1803.09820.

[43] M. Hittmeir, A. Ekelhart, R. Mayer, On the utility of synthetic data: An empirical evaluation on machine learning tasks, PervasiveHealth Pervasive Comput. Technol. Healthc. (2019) http://dx.doi.org/10.1145/3339252.3339281.

[44] D. Rankin, M. Black, R. Bond, J. Wallace, M. Mulvenna, G. Epelde, Reliability of supervised machine learning using synthetic data in health care: Model to preserve privacy for data sharing, JMIR Med. Inform. 8 (7) (2020) http://dx.doi.org/10.2196/18910.

[45] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, A review of feature selection methods on synthetic data, Knowl. Inf. Syst. 34 (3) (2013) 483–519, http://dx.doi.org/10.1007/s10115-012-0487-8.

[46] Y.I. Kuchin, R.I. Mukhamediev, K.O. Yakunin, One method of generating synthetic data to assess the upper limit of machine learning algorithms performance, Cogent Eng. 7 (1) (2020) http://dx.doi.org/10.1080/23311916.2020.1718821.

[47] D. Dai, C. Sakaridis, S. Hecker, L. Van Gool, Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding, Int. J. Comput. Vis. 128 (5) (2020) 1182–1204, http://dx.doi.org/10.1007/s11263-019-01182-4.

[48] H. Kaur, H.S. Pannu, A.K. Malhi, A systematic review on imbalanced data challenges in machine learning: Applications and solutions, ACM Comput. Surv. 52 (4) (2019) http://dx.doi.org/10.1145/3343440.

[49] H.P. Das, R. Tran, J. Singh, X. Yue, G. Tison, A. Sangiovanni-Vincentelli, C.J. Spanos, Conditional synthetic data generation for robust machine learning applications with limited pandemic data, 19, 2021, URL http://arxiv.org/abs/2109.06486.

[50] R.K. Kalle, P. Kumar, S. Mohan, M. Sakata, Simulation-driven optimization of urban bus transport, WIT Trans. Built Environ. 186 (2019) 97–108, http://dx.doi.org/10.2495/UT190091.

[51] S.M.H. Moosavi, A. Ismail, C.W. Yuen, Using simulation model as a tool for analyzing bus service reliability and implementing improvement strategies, PLoS One 15 (5) (2020) 1–26, http://dx.doi.org/10.1371/journal.pone.0232799.

[52] S.R. Pells, An approach to the simulation of bus passenger journey times for the journey to work, Transp. Plan. Technol. 14 (1) (1989) 19–35, http://dx.doi.org/10.1080/03081068908717411, https://www.witpress.com/Secure/elibrary/papers/UT19/UT19009FU1.pdf http://www.tandfonline.com/doi/abs/10.1080/03081068908717411.

[53] L. von Rueden, S. Mayer, R. Sifa, C. Bauckhage, J. Garcke, Combining Machine Learning and Simulation to a Hybrid Modelling Approach: Current and Future Directions, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 12080 LNCS, Springer International Publishing, 2020, pp. 548–560, http://dx.doi.org/10.1007/978-3-030-44584-3{_}43.

[54] T. Reich, M. Budka, D. Robbins, D. Hulbert, Survey of ETA prediction methods in public transport networks, 2019, arXiv.

[55] T. Zhou, W. Wu, L. Peng, M. Zhang, Z. Li, Y. Xiong, Y. Bai, Evaluation of urban bus service reliability on variable time horizons using a hybrid deep learning method, Reliab. Eng. Syst. Saf. 217 (May 2021) (2021) 108090, http://dx.doi.org/10.1016/j.ress.2021.108090.

[56] E. Hans, N. Chiabaut, L. Leclercq, R.L. Bertini, Real-time bus route state forecasting using particle filter and mesoscopic modeling, Transp. Res. C 61 (2015) 121–140, http://dx.doi.org/10.1016/j.trc.2015.10.017.

[57] C. Coffey, A. Pozdnoukhov, F. Calabrese, Time of arrival predictability horizons for public bus routes, in: 4th ACM SIGSPATIAL International Workshop on Computational Transportation Science 2011, CTS'11, in Conjunction with ACM SIGSPATIAL GIS 2011, 2011, pp. 1–5, http://dx.doi.org/10.1145/2068984.2068985.

[58] Q. Ye, W.Y. Szeto, S.C. Wong, Short-term traffic speed forecasting based on data recorded at irregular intervals, IEEE Trans. Intell. Transp. Syst. 13 (4) (2012) 1727–1737, http://dx.doi.org/10.1109/TITS.2012.2203122.

[59] B. Varga, T. Tettamanti, B. Kulcsár, Energy-aware predictive control for electrified bus networks, Appl. Energy 252 (August) (2019) 113477, http://dx.doi.org/10.1016/j.apenergy.2019.113477.