A Predictive Model of Arrival Times for Smart Shuttle Buses in Astana, Kazakhstan

Darya Taratynova[1], Assel Kassenova[2], Bissenbay Dauletbayev[3], Muammar Al-Shedivat[4], and Eugene Pinsky[2](✉)

[1] Department of Mathematics, Nazarbayev University, 53 Kabanbay Batyr Avenue, Astana 010000, Kazakhstan
darya.taratynova@nu.edu.kz
[2] Department of Computer Science, Metropolitan College, Boston University, 1010 Commonwealth Avenue, Boston, MA 02215, USA
{kassel,epinsky}@bu.edu
[3] Department of Computer Science, SDU University, 1/1 Abylai Khan Street, Kaskelen, Almaty 040900, Kazakhstan
b.dauletbayev@sdu.edu.kz
[4] UvU Shuttle, Astana, Kazakhstan
muammar@uvu.kz

Abstract. Accurate prediction of the estimated time of arrival (ETA) for buses is crucial for shuttle companies aiming to enhance profitability and minimize costs. This paper proposes leveraging historical bus route data to predict estimated bus arrivals at specific stations, employing K-Nearest Neighbors (KNN) supervised machine learning (ML) techniques. We develop a simple and interpretable solution that abstains from complexity, as bus routes dynamically adapt to passenger demands.

Keywords: ETA Prediction · KNN · Shuttle bus · Transportation · Clustering

1 Introduction

In today's bustling urban landscapes, individuals find themselves navigating a whirlwind of commitments, shuttling between home, work, schools, and various engagements amidst the chaos of morning rush hours and evening commutes. This surge in activity inevitably leads to significant traffic congestion during peak hours, with roads clogged by overcrowded vehicles and buses packed with passengers, exacerbating stress and disrupting schedules [16].

T. Zlateva and G. Tuparov (Eds.): CSECS 2024, LNICST 609, pp. 29–49, 2025.
https://doi.org/10.1007/978-3-031-84312-9_2

In response to these challenges, innovative transportation services like UvU have emerged, offering solutions to the limitations of traditional public transit and taxis. Operational in Astana, Kazakhstan, UvU provides a convenient, comfortable, and cost-effective shuttle experience by allowing passengers to pre-book seats, guaranteeing seating, and relieving rush hour congestion. Since passengers pre-book seats, routes are created dynamically based on demand. Despite these advancements, a critical need persists for accurate ETA predictions to further enhance commuters' experiences (Fig. 1).

Fig. 1. UvU Shuttles

The primary aim of this paper is to address this need by predicting the ETA for UvU's shuttle buses, thereby facilitating better scheduling of commitments and reducing stress for passengers. Specifically, we aim to:

– Develop a model for ETA prediction tailored to the unique operational context of UvU's shuttle service in Astana.
– Evaluate the performance of our model using real-world data collected from shuttle drivers and integrate the Open Source Routing Machine (OSRM) for enhanced route planning and navigation.

Building upon existing research such as [17], which focused on navigation applications in Kuala Lumpur, Malaysia, our study distinguishes itself by leveraging UvU's data and integrating OSRM. While previous studies have highlighted the suitability of certain navigation APIs for ETA prediction, our approach offers a novel perspective by directly utilizing data from the transportation service itself.

Astana, UvU's operational hub, presents a dynamic urban landscape divided into two zones: the right and left sectors. Understanding this context is crucial as commuters navigate between the bustling city center and the residential areas during peak hours. Visual representations of Astana's intricate road network

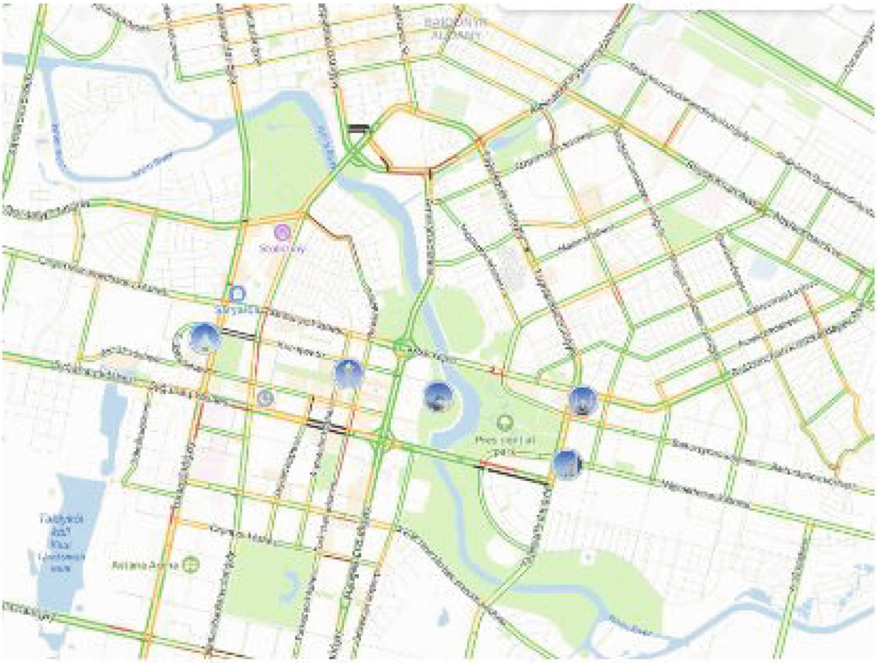


Fig. 2. Astana's Road Network

and traffic patterns provide insight into the complexities of travel logistics in this environment (see Figs. 2, 3, 4, and 5).

The subsequent sections of this paper are organized as follows: Sect. 2 delves into a comprehensive literature review of studies similar to ours; Sect. 3 discusses our data collection and analysis methodology; Sect. 4 outlines the development of our ETA prediction model. Finally, in Sect. 5, we draw conclusions, discuss implications, and suggest avenues for future research.

2 Literature Review

[10] present a methodology centered on route prediction and time-of-arrival (ToA) estimation within a multiple-station car-sharing system. Their implementation leverages automatic vehicle locator hardware and demonstrates promising results in improving service reliability. However, concerns arise regarding the generalizability of their findings beyond the specific system tested and the potential for refinement in the estimation procedure by increasing data frequency.

In a different vein, [1] propose a novel data dissemination scheme in Intelligent Transportation Systems, utilizing non-intelligent vehicles with passive radio frequency identification tags. While their approach shows superiority over existing methods in simulation, questions linger regarding real-world implementation

Fig. 3. Satellite View of Astana's Road Network

and scalability, as well as the exclusion of non-intelligent vehicles without active communication modules.

Meanwhile, [8] delve into the intricacies of ToA error in ultra-wide bandwidth ranging systems, providing valuable insights into lower bounds in realistic multipath environments. Their work underscores the challenges posed by dense multipath environments and highlights discrepancies in classical lower-bound models.

Moving to bus-arrival-time (BAT) prediction, [6] presents a model emphasizing historical travel time data, with the multi-section travel time-based approach exhibiting superiority over existing methods. Despite its success, limitations such as the influence of weather conditions and the need for further validation underscore the complexity of real-world implementation.

Similarly, [9] compare link-based and section-based models, with the latter demonstrating superior performance. Their findings shed light on the importance of model choice in predicting BATs, yet questions remain regarding the broader applicability of these models across diverse contexts.

In the realm of BAT prediction models, [18] explores various forecasting methods, highlighting the benefits of electronic indicators at bus stations. While their approach offers valuable insights, further validation and an extension of the model are deemed necessary for robust implementation.

Fig. 4. Light Traffic in Astana

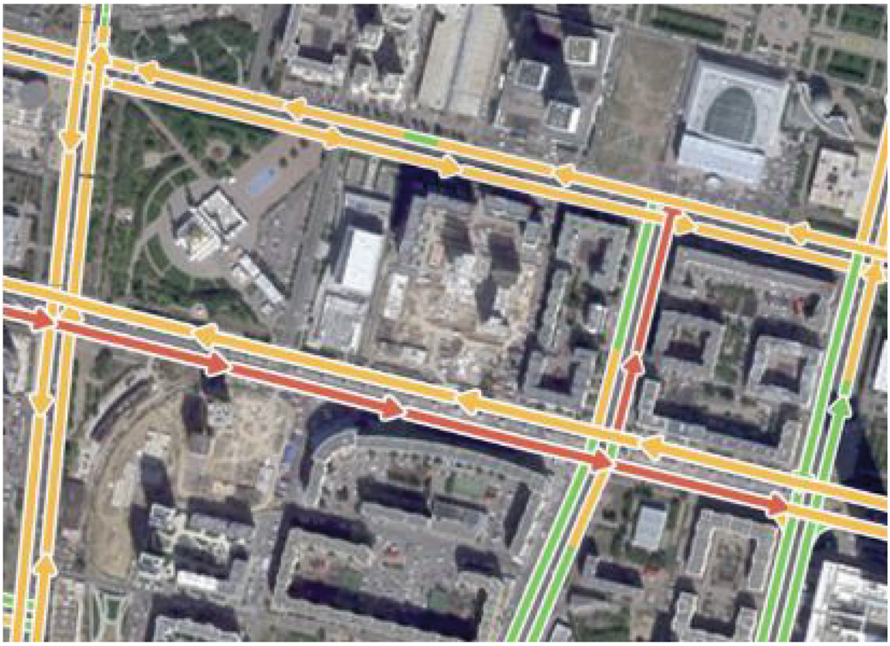


Fig. 5. Heavy Traffic in Astana

Transitioning to urban transportation dynamics, [21] propose a methodology to estimate urban link travel times using taxicab origin-destination trip data. Their approach showcases the potential of taxicab data as a complementary data source, yet the study's reliance on inferred paths and model assumptions warrants careful consideration.

On the topic of Origin-Destination (O-D) matrix construction, [13] introduces a data-driven method using spatial clustering, offering a promising approach to incrementally building O-D matrices. However, the study's reliance on clustering techniques and sample attribute discretization underscores the need for further validation and refinement.

In the realm of speed management strategies, [5] explores the effects of road features on speeding proportions, advocating for a grouped random parameter model for enhanced performance. Their findings underscore the importance of nuanced modeling approaches in understanding and mitigating speeding behaviors.

Transitioning back to BAT prediction, [12] present Artificial Neural Network (ANN) models for real-time prediction, with the hierarchical ANN model showcasing superior performance. Despite its success, the limitations in prediction window and scenario coverage highlight the complexities inherent in modeling bus arrival times.

Meanwhile, [20] introduces a hybrid model combining support vector machine and Kalman filtering for BAT prediction, demonstrating its feasibility and performance advantages. Their work underscores the potential of hybrid modeling approaches in enhancing prediction accuracy.

In the realm of taxi dispatch and destination prediction, [22] propose a novel taxi dispatch system and destination prediction method, showcasing significant improvements in global success rate and user experience. However, the study's reliance on A/B testing and limited dataset size warrants further investigation into broader applicability.

Transitioning to taxi trajectory inference, [23] presents a method for inferring taxi statuses based on Global Positioning System (GPS) trajectories, showcasing advantages over baseline approaches. Despite its success, the study's limited dataset size raises questions about the method's robustness across diverse scenarios.

Finally, [11] introduces a data-driven approach for taxi destination and trip time prediction, demonstrating robust performance in real-world scenarios. Despite its success, challenges related to missing GPS updates and erroneous information underscore the complexities of real-world data integration.

3 Data Collection and Analysis

The data were collected from September 1, 2023, to October 31, 2023, along the routes of Astana. The main source of data is the database of the startup UvU, which collected the drivers' geo-positions including latitude, longitude, and the time at which each coordinate was collected-while executing the tour with the consent of the drivers to do so. Each tour includes bus stops along with planned arrival times and coordinates. During the data refinement phase, records with missing values were excluded to preserve the integrity of our analysis, as incomplete data could skew the ETA predictions. However, we opted to retain outliers within the duration data, under the premise that atypically long or short travel times, while rare, are a genuine aspect of traffic conditions and thus valuable for understanding the full spectrum of ETA variability. An initial analysis of the dataset shows notable patterns in trip frequency across different timeframes. The weekly distribution in Fig. 6 shows a pronounced variability, with the number of trips peaking on Monday and significantly tapering off by Wednesday. This trend may be indicative of the urban commute patterns, which often see a surge at the start of the workweek. Furthermore, September's data significantly outnumbers that of October in Fig. 7, suggesting seasonal or operational influences on the data capture process. This discrepancy might be caused by our data quality, where data contributed by drivers rated below a 3-point threshold were systematically excluded from the October dataset to maintain the integrity of the analysis.

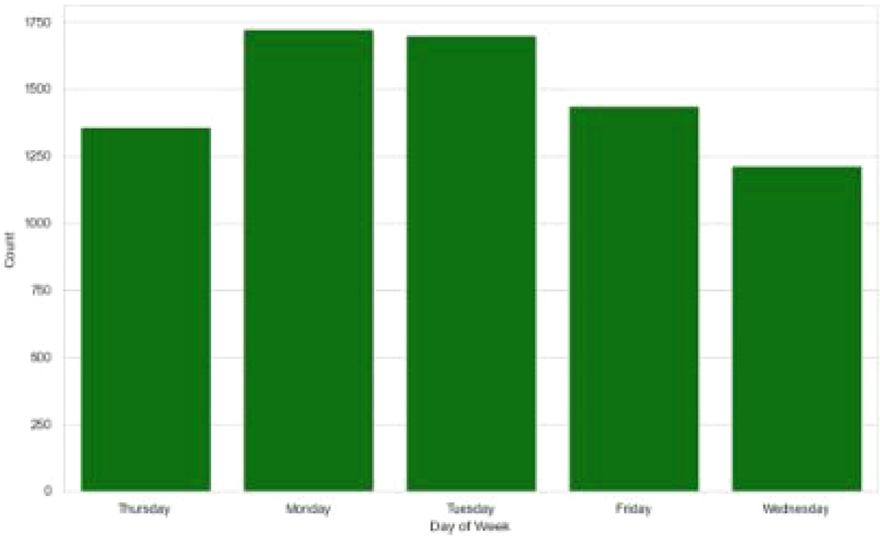


Fig. 6. Trips frequency by weekdays

The quality of the coordinates provided by drivers is evaluated on a scale from 0 to 4, where 0 represents low quality and 4 represents the highest quality of coordinates, indicating that they were sent almost without any gaps and can be observed as an almost continuous path. An example of the 4-scaled tour is provided in Fig. 9, illustrating a continuous path with high-quality coordinates. The selected tours comprised only those rated 3 and 4 on the quality scale, ensuring a focus on routes characterized by higher-quality geo position data. Despite

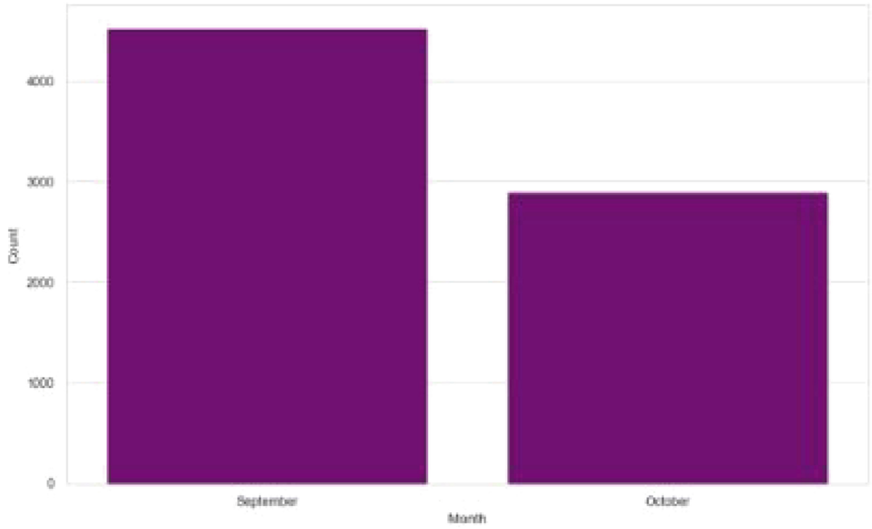


Fig. 7. Trips frequency by month

the provision of routes via the UvU app, leveraging the integrated OSRM route engine, these selections were imperative due to the inherent limitations of automated route generation. Specifically, the OSRM engine does not account for real-time traffic conditions or road maintenance activities. Given the prevalent utilization of OSRM within the application framework, our analysis delved into the ratio of actual time spent between stops compared to the estimated time provided by OSRM. Figure 10 elucidates this ratio spanning from September 10th to November 5th, focusing on a single selected route. A higher ratio indicates a discrepancy between the actual time spent and the accuracy of the ETA provided by OSRM. While analyzing the ratio of actual versus estimated travel times provided by OSRM, as shown in Fig. 4, we anticipated observing discernible patterns of discrepancies that could inform the reliability of the OSRM's ETA. Surprisingly, Fig. 8, which presents the correlation of these ratios across different days, does not reveal a clear pattern. The correlation values between days are relatively high, suggesting some level of consistency in the data, but this consistency does not necessarily equate to accuracy.

Given our concerns about the precision of the OSRM data, these correlations should be interpreted with caution. They may not fully capture the real-world complexities of daily traffic flows or the impact of unpredictable events such as roadworks or accidents. The high correlation might also stem from systematic biases within the OSRM algorithm that persist across different days, rather than true similarities in traffic conditions.

Therefore, seasoned drivers often rely on their expertise or consult alternate route engine paths between two bus stops, as shown in Fig. 11a 11b to navigate efficiently differently. These two paths may take significantly different times,

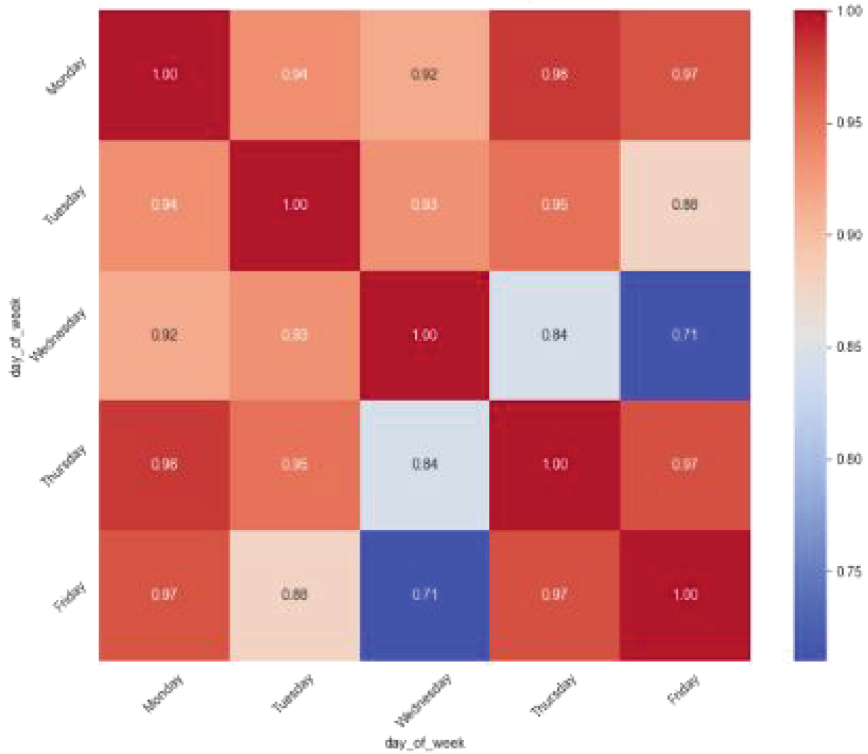


Fig. 8. Correlation of actual time and OS between weekdays

which could create inconsistencies. The figure illustrates the situation between two close points, but as the distance increases, more options for different routes may appear. For this reason, we first split the driver's path between two bus stops into a sequence of crossroads. To identify the crossroads between bus stops, we use OSRM, which returns the maneuvers and their types as illustrated in Fig. 12.

After splitting the route into a sequence of crossroads, we match the positions of drivers to determine the actual time taken between the crossroads. As a result, we have divided the positions of drivers into intervals and then merged them with the actual times taken. However, for some intervals, we may not have data because coordinates are not always shared perfectly. The following collection of intervals is shown in Fig. 13.

In the final dataset, we have the following features organized in a table: In the final dataset, we have the following features, each with a brief explanation:

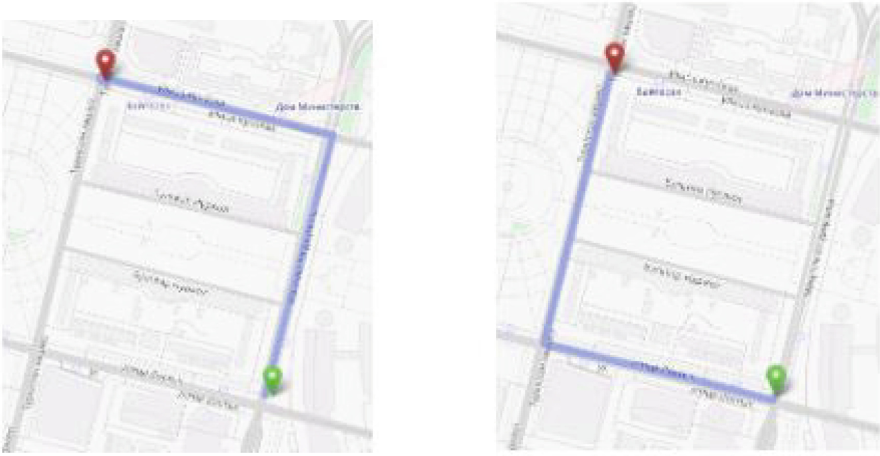
1. tour_id: Unique identifier for the tour.
2. initial_lat: Latitude at the start of the interval.
3. initial_lng: Longitude at the start of the interval.
4. final_lat: Latitude at the end of the interval.
5. final_lng: Longitude at the end of the interval.

Fig. 9. Example of Driver's Geoposition Data with Rating Scale: 4

Fig. 10. Actual to OSRM time Ratio within one route

6. distance: Distance between the start and end of the interval.
7. date: Date of the tour.
8. actual_time_taken: Actual time taken by the driver between start and end of the interval.
9. day_of_the_week: Day of the week of the tour.
10. month: Month of the tour.

The final dataset contains data for 609 tour instances for our predictions. These tours were part of routes operating from 6 AM to 9 AM on weekdays, from Monday to Friday. Figure 14 presents the distribution of distances covered



(a) Option 1 (b) Option 2

Fig. 11. Options of routes between two points

during these tours. It shows a right-skewed distribution, with the majority of the tours covering shorter distances and a few extending to longer distances, highlighting the prevalence of short to medium-length trips within the city during the morning hours. This characteristic is crucial for our ETA prediction model, as most commutes fall within a typical urban range, allowing the model to be calibrated specifically for these common scenarios.

Figure 15 shows the distribution of trip durations. Similar to the distance distribution, trip duration also follows a right-skewed pattern, with most trips being completed within a shorter timeframe and fewer trips taking significantly longer. This indicates that despite the variability of distances, the travel times remain relatively consistent, likely due to the condensed operating window during peak morning hours. This consistency in durations lends itself to a more predictable model output, giving credence to the use of historical data for forecasting ETAs.

These patterns in the distributions of distance and duration underscore the importance of factoring in urban traffic behavior when designing predictive models for ETA. They provide insight into the typical patterns of morning commutes, which can be instrumental in adjusting and improving the accuracy of such models. In synthesizing the data collection and processing steps, this study constructed a dataset for developing a reliable predictive model for ETA. The deliberate curation of data and thoughtful consideration of its limitations establish a solid foundation for the subsequent modeling phase.

As we transition to model development, it is important to reflect on the significance of our data analysis. The insights garnered from assessing trip durations, distributions, and correlations are invaluable. They not only inform the model's design but also shed light on the underlying urban transport dynamics. This

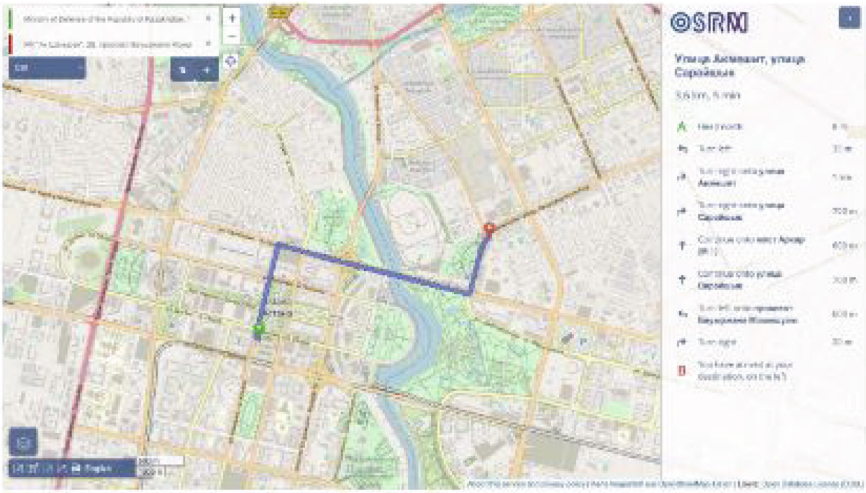


Fig. 12. Example of Maneuvers from OSRM

understanding is imperative as we aim to enhance the predictive accuracy of our ETA model, thus facilitating a more efficient and dependable shuttle service.

4 Model Development

In this section, we discuss how we use machine learning to predict travel times with our transportation data. We combine clustering and the KNN algorithm to tackle the complex variations in our data. Our goal is to create a model that not only predicts travel times accurately but also helps us understand what affects these times. By using clustering to group similar data and KNN to make predictions, our model adapts well to different traffic conditions.

Clustering is a machine-learning technique used primarily in unsupervised learning to group a set of objects into clusters based on their similarities, without prior knowledge of the group definitions. This approach is useful in many applications such as identifying patterns or structures in data, anomaly detection, customer segmentation, and organizing large datasets into meaningful categories.

In our analysis of transportation data, we chose the KNN algorithm for predicting actual time taken based on several key considerations that align with the characteristics of our dataset. The dataset comprises attributes such as the number of crossroads, the distance between bus stops, actual time taken, day of week, time category, and month. These features capture spatial-temporal dynamics that significantly influence travel time outcomes. We selected the KNN method for its non-parametric nature, meaning it does not assume a fixed pattern in the data, which is crucial given that driving conditions and traffic can vary significantly at different times and on different days. KNN's ability to adapt to these

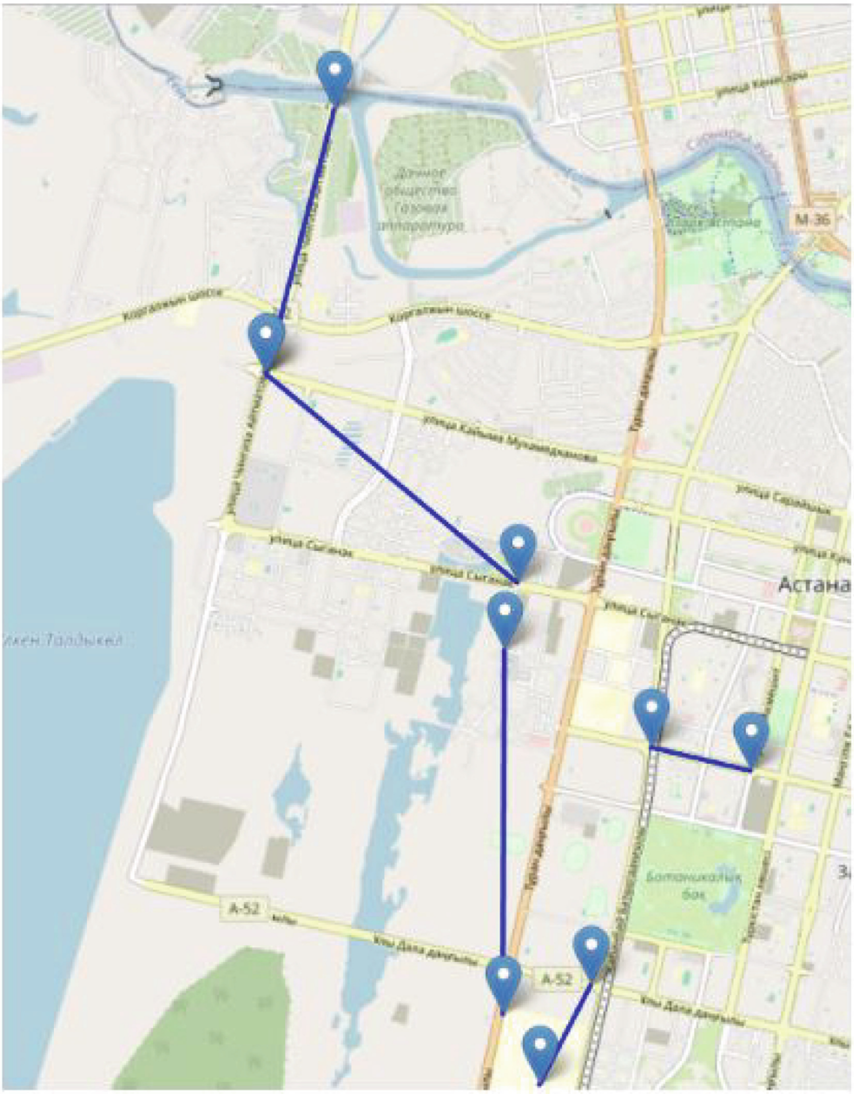


Fig. 13. Final collection

variations by comparing new situations with the most similar past examples makes it especially reliable, even when data trends change or when there are anomalies in the data.

The fundamental principle behind KNN is to predict the label of a data point by examining the k closest labeled data points and taking an average of their values for regression; the output is the predicted value for the object. The choice of 'k' affects the algorithm's sensitivity and accuracy: a smaller 'k' makes

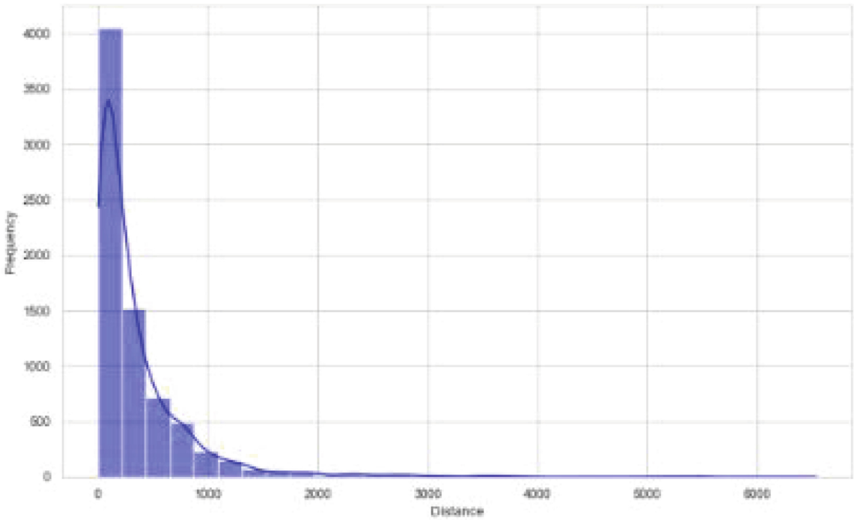


Fig. 14. Distance distribution

the algorithm sensitive to noise, while a larger 'k' can make it computationally expensive and potentially less accurate due to overgeneralizing from the training set. To make predictions, KNN calculates the distance between the new data point and all others in the dataset. Common distance metrics used are Euclidean, Manhattan, and Hamming distance, with the choice depending on the type of data and the specific application. In our case, we use the Manhattan distance, also known as the L1 distance. This metric calculates the distance between two points by only allowing horizontal and vertical movements. The formula for Manhattan distance between two points x and y in an n-dimensional space is given as:

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

Here, $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ are two points in n-dimensional space, and $|x_i - y_i|$ represents the absolute difference between the corresponding components of x and y [15].

The algorithm computes the distance from the test point to all points in the training set, selects the k-closest points, and then predicts by averaging the values of these selected points. [7].

Figure 16 provides a schematic representation of our clustering model.

Initially, the model extracts clustering features from the dataset, which include the number of crossroads, total distance, intervals between bus stops, days of the week, time categories, and months. Notably, we exclude the 'actual time taken' from this feature set to prevent its influence on the clustering process.

To ensure no single feature disproportionately influences the clustering, we implement a normalization step. Here, the importance of each feature is assessed,

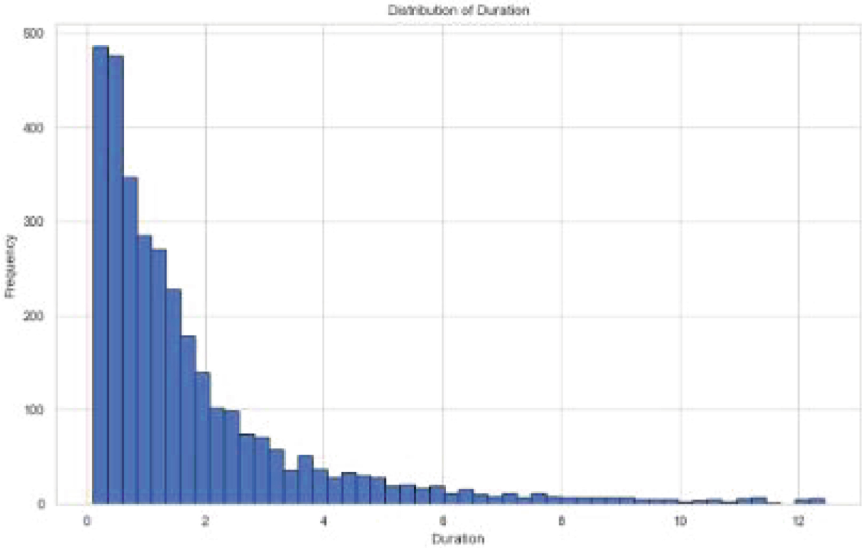


Fig. 15. Duration distribution

and weights are assigned such that their total is one, balancing their impact on the distance calculations within the K-means algorithm. These normalized weights are then applied to the dataset, now scaled by the StandardScaler.

In the clustering phase, the K-means algorithm, with the kmeans++ initializer, is employed to identify the optimal starting positions for our cluster centers. Upon completion of the clustering, data points are allocated to the nearest cluster based on the Manhattan distance, resulting in 12 distinct groups. These groups are reflective of similar travel conditions and patterns.

Each cluster's mean actual time taken is calculated, serving as a proxy for the expected travel time for conditions akin to those within the cluster.

During the prediction phase, travel times for either new or existing data points are estimated by locating the closest cluster centroids using the Manhattan distance. Specifically, the three nearest centroids to a data point are considered. The 'estimated time taken' is then ascertained by averaging the 'actual time taken' for these centroids, thereby providing a predictive measure based on historical cluster performance. Shifting focus to the assignment of feature importance, our initial strategy incorporated a tree-based analysis. However, it became clear that manual fine-tuning was more effective. Each feature was assigned a weight on a scale from zero to four, with these weights serving as multipliers of influence in our model. Observing the spread of these weights. By placing the distributions of 'actual time taken' beside those of 'predicted time taken,' we evaluated the precision of our model. Discrepancies between the two prompted a recalibration of the weights. This recalibration was iterative, with successive adjustments made until the predicted times aligned more closely

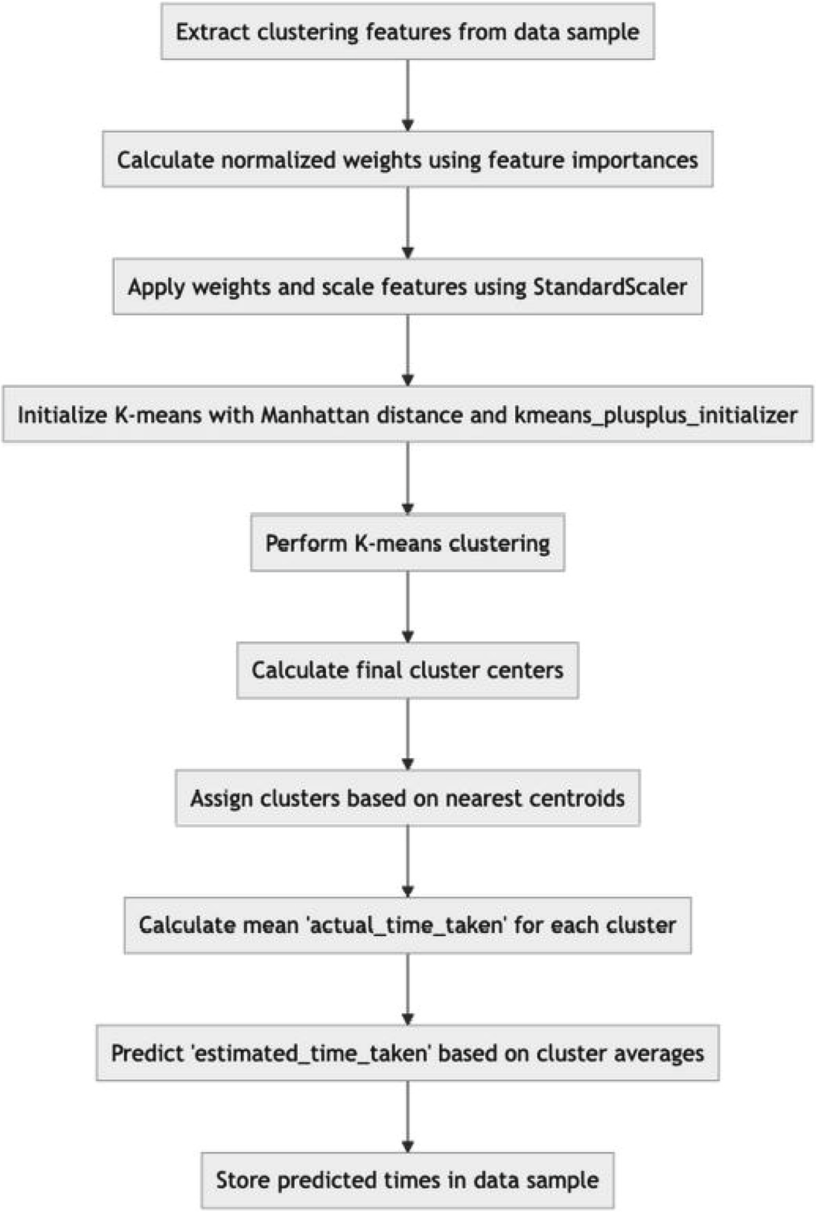


Fig. 16. Cluster-Based Travel Time Prediction Model Workflow

with the empirical data, thereby fine-tuning our model to deliver more accurate forecasts.

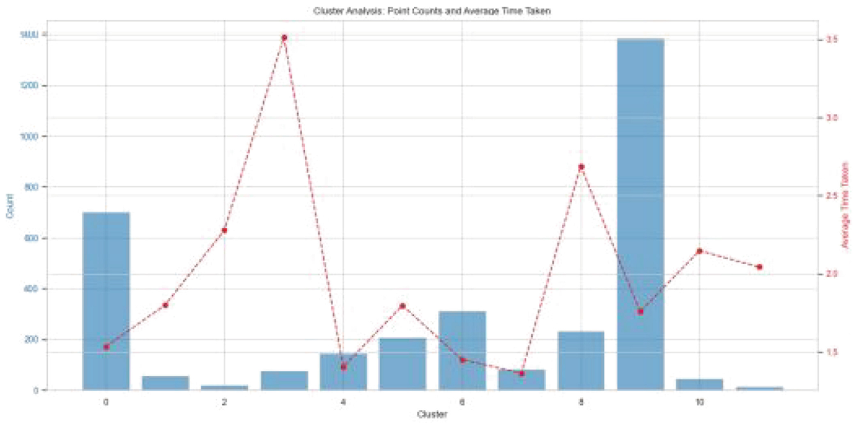
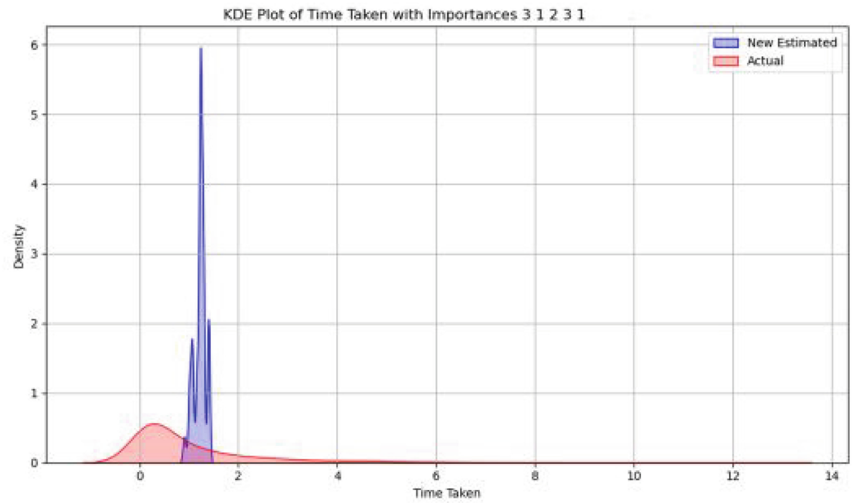


Fig. 17. Average Actual Travel Time Across Clusters

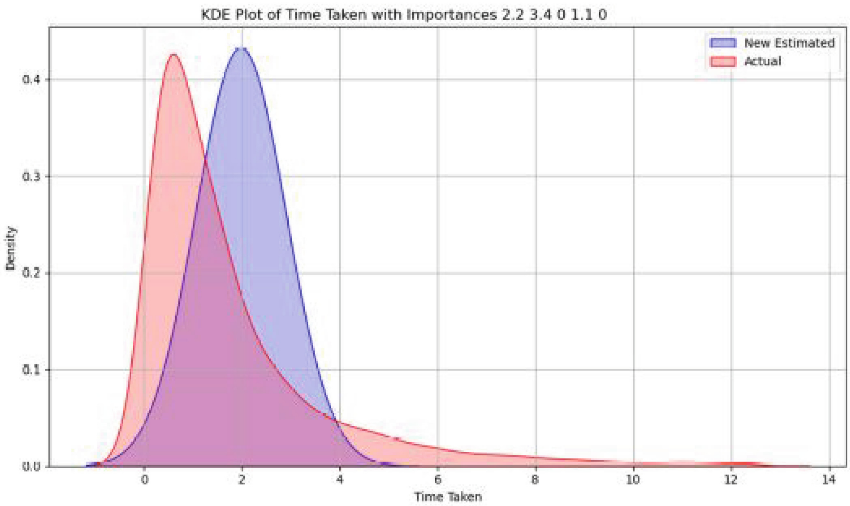
Figure 17 illustrates the average actual times taken by each cluster, with a notable variation across the spectrum. Cluster 3 is distinguished by the highest average time, implying a potential complexity or a set of challenging conditions within this group's context. In stark contrast, Clusters 0 and 4 demonstrate the shortest average times, which could be indicative of fewer complexities. Significantly, Cluster 3 registers the highest average time, suggesting that this group might be experiencing more complex travel scenarios. In contrast, Clusters 0 and 4 are shown to have the smallest travel times. The mid-range travel times exhibited by Clusters 1, 2, 5, 8, 9, 10, and 11 suggest a varying impact of the features on travel conditions.

Figure 18 showcases two Kernel Density Estimation (KDE) plots that capture the model's performance at different stages of feature weight calibration. Specifically, the initial weights assigned were 3 for the number of crossroads, 1 for total distance, 2 for the day of the week, 3 for the time of day, and 1 for the month. The final weights were refined to 2.2 for crossroads, 3.4 for distance, 0 for the day of the week, 1.1 for time of day, and 0 for the month. In Fig. 11a, the KDE plot reveals a comparison between 'actual time taken' and 'estimated time taken' using the initial weighting approach (3-1-2-3-1). The noticeable separation between the 'estimated' and 'actual' distributions suggests a mismatch, indicating that the initial feature weights were not optimal for capturing the travel time dynamics.

Moving to Fig. 18b, we observe the KDE plot with revised feature weights. The resulting distributions now exhibit a greater degree of overlap, signaling an enhanced alignment between the estimated and actual travel times. The convergence of these curves implies that the adjustments made to the feature weights have improved the model's predictive accuracy, bringing the estimated values into closer agreement with the actual observations.



(a) Initial Feature Weighting Impact

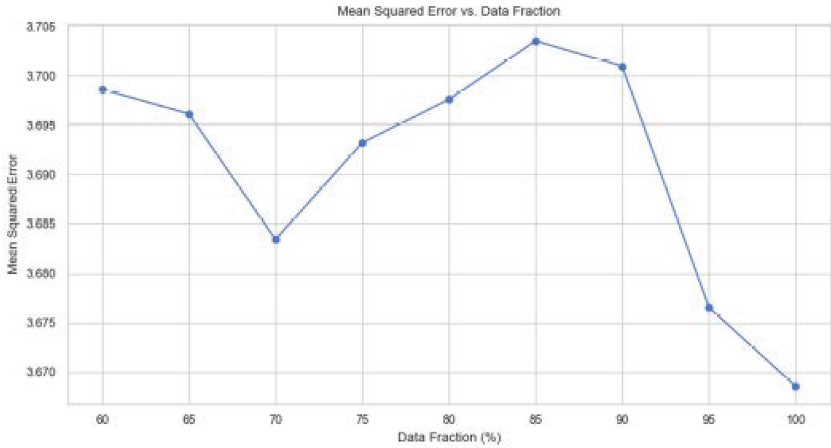


(b) Optimized Feature Weighting Impact

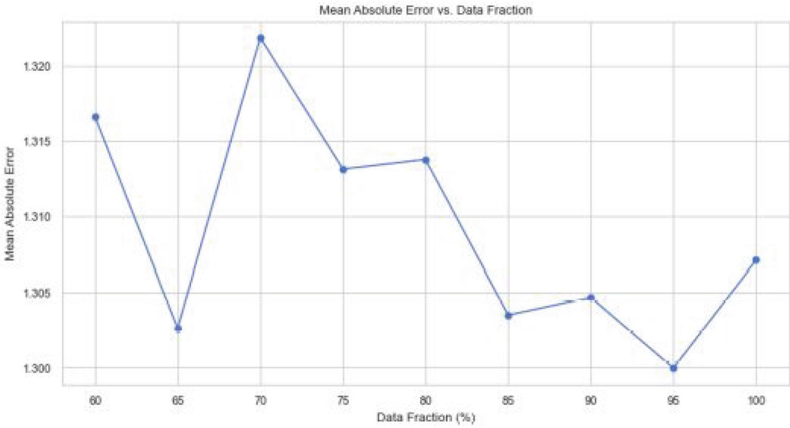
Fig. 18. Feature Weighting Impact on Travel Time Predictions

We also investigated the impact of different size of data for initial clustering and analyzed results of Mean Absolute Error (MAE) and Mean Squared Error (MSE). They are presented in Fig. 19.

In the MSE graph, there's a general increase in error as the data fraction increases from 60% to 85%. However, at 90% data usage, there is a marked decrease, and the error drops significantly as we reach 100% data usage. This suggests that the model's predictions become more accurate as more data is



(a) Mean Squared Error



(b) Mean Absolute Error

Fig. 19. Impact of Data Volume on Model Accuracy

included, with the most substantial improvement occurring when the model uses all available data.

The MAE graph shows a more variable trend, with the error peaking at 70% data usage, then dropping at 75%, rising again slightly at 80% before decreasing steadily. The error increases once more at 95% but then decreases at full data usage. These fluctuations could be due to representativeness at those particular fractions.

Our combined approach of clustering and KNN effectively computes travel time predictions that adapt to varying traffic conditions and temporal factors. As a result, our model stands as an explainable and dynamic method for estimating arrival times.

5 Conclusion

In conclusion, this paper introduces an approach utilizing KNN clustering techniques to minimize passenger waiting times for UvU customers in Astana, Kazakhstan. By harnessing historical data from UvU bus routes, our ML model predicts estimated bus arrivals, thereby enhancing shuttle operations' efficiency and cost-effectiveness. The proposed approach holds the promise of improving transportation efficiency and enhancing overall commuting experiences in Astana.

In the contemporary landscape, there is considerable interest in transformers to address routing problems [19]. We believe that exploring the applications of neural networks, particularly with transformer models, would offer a fruitful avenue for further research. Additionally, another intriguing approach would involve modeling this problem and applying reinforcement learning, which could yield highly promising results [2–4,14].

Acknowledgments. We would like to express our sincere gratitude to UvU for generously providing us with historical data on their bus routes. This data was invaluable for conducting our experiments and analysis, greatly enhancing the quality and reliability of our research. We also acknowledge the support of the Center for International Programs "Bolashak", which played a crucial role in facilitating the academic endeavors necessary for this study. The contributions from both have been instrumental in advancing our collective research goals. Finally, we would like to express our gratitude to Boston University Metropolitan College for their support.

References

1. Ali, K., Hassanein, H.: Using passive RFID tags for vehicle-assisted data dissemination in intelligent transportation systems. In: 2009 IEEE 34th Conference on Local Computer Networks, pp. 688–694. IEEE (2009)
2. Bogyrbayeva, A., Jang, S., Shah, A., Jang, Y.J., Kwon, C.: A reinforcement learning approach for rebalancing electric vehicle sharing systems. IEEE Trans. Intell. Transp. Syst. **23**(7), 8704–8714 (2021)
3. Bogyrbayeva, A., Meraliyev, M., Mustakhov, T., Dauletbayev, B.: Machine learning to solve vehicle routing problems: a survey. IEEE Trans. Intell. Transp. Syst. **25**, 47754–4772 (2024)
4. Bogyrbayeva, A., Yoon, T., Ko, H., Lim, S., Yun, H., Kwon, C.: A deep reinforcement learning approach for solving the traveling salesman problem with drone. Transp. Res. Part C: Emerg. Technol. **148**, 103981 (2023)
5. Cai, Q., Abdel-Aty, M., Mahmoud, N., Ugan, J., Ma'en, M.A.: Developing a grouped random parameter beta model to analyze drivers' speeding behavior on urban and suburban arterials with probe speed data. Accid. Anal. Prevent. **161**, 106386 (2021)
6. Chen, G., Yang, X., Zhang, D., Teng, J.: Historical travel time based bus-arrival-time prediction model. In: ICCTP 2011: Towards Sustainable Transportation Systems, pp. 1493–1504. ICCTP (2011)
7. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Trans. Inf. Theory **13**(1), 21–27 (1967)

8. Dardari, D., Chong, C.-C., Win, M.Z.: Improved lower bounds on time-of-arrival estimation error in realistic UWB channels. In: 2006 IEEE International Conference on Ultra-Wideband, pp. 531–537. IEEE (2006)
9. Guojun, C., Xiaoguang, Y., Jian, A., Dong, Z.: Bus-arrival-time prediction models: link-based and section-based. J. Transp. Eng. **138**(1), 60–66 (2012)
10. Karbassi, A., Barth, M.: Vehicle route prediction and time of arrival estimation techniques for improved transportation system management. In: IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No. 03TH8683), pp. 511–516. IEEE (2003)
11. Hoang Thanh Lam, Ernesto Diaz-Aviles, Alessandra Pascale, Yiannis Gkoufas, and Bei Chen. (blue) taxi destination and trip time prediction from partial trajectories. *arXiv preprint* arXiv:1509.05257, 2015
12. Lin, Y., Yang, X., Zou, N., Jia, L.: Real-time bus arrival time prediction: case study for jinan, china. J. Transp. Eng. **139**(11), 1133–1140 (2013)
13. Moreira-Matias, L., Gama, J., Ferreira, M., Mendes-Moreira, J., Damas, L.: Time-evolving od matrix estimation using high-speed gps data streams. Expert Syst. Appl. **44**, 275–288 (2016)
14. Mustakhov, T., Akhmetbek, Y., Bogyrbayeva, A.: Deep reinforcement learning for stochastic dynamic vehicle routing problem. In: 2023 17th International Conference on Electronics Computer and Computation (ICECCO), pp. 1–5. IEEE (2023)
15. Nayak, S., Bhat, M., Subba Reddy, N.V., Rao, B.A.: Study of distance metrics on k - nearest neighbor algorithm for star categorization. J. Phys. Conf. Ser. **2161**(1), 012004 (2022)
16. Russell, M.L.: Travel time use on public transport: what passengers do and how it affects their wellbeing. Ph.D. thesis, University of Otago (2012)
17. Gan, S.Q., et al.: A comparative study of navigation API eta accuracy for shuttle bus tracking. In: Salvendy, G., Wei, J. (eds.) Design, Operation and Evaluation of Mobile Communications. HCII 2022. LNCS, vol. 13337, pp. 446–461. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-05014-5_37
18. Sun, W., Chen, P., Song, T., Wang, Q.: Bus arrival time prediction model study in apts. In: ICCTP 2010: Integrated Transportation Systems: Green, Intelligent, Reliable, pp. 2597–2605. ICCTP (2010)
19. Vaswani, A., et al.: Attention is all you need. In: Advances In Neural Information Processing Systems, vol. 30 (2017)
20. Bin, Yu., Yang, Z.-Z., Chen, K., Bo, Yu.: Hybrid model for prediction of bus arrival times at next station. J. Adv. Transp. **44**(3), 193–204 (2010)
21. Zhan, X., Hasan, S., Ukkusuri, S.V., Kamga, C.: Urban link travel time estimation using large-scale taxi data with partial information. Transp. Res. Part C: Emerg. Technol. **33**, 37–49 (2013)
22. Zhang, L., et al.: A taxi order dispatch model based on combinatorial optimization. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 2151–2159 (2017)
23. Zhu, Y., Zheng, Y., Zhang, L., Santani, D., Xie, X., Yang, Q.: Inferring taxi status using GPS trajectories. arXiv preprint arXiv:1205.4378 (2012)