Data augmentation for Sparse Graph Traversals

Exploring data augmentation options to enhance deep learning model performance

Mels Lutgerink[1]
Supervisor(s): Elvin Isufi[1] **Ting Gao**[1]
[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 22, 2025

Name of the student: Mels Lutgerink
Final project course: CSE3000 Research Project
Thesis committee: Elvin Isufi, Ting Gao, Jing Sun

Abstract

This research investigates the effectiveness of graph-based data augmentation techniques in improving the performance of DG4b, a deep learning model designed to estimate bicycle travel times in urban environments. Given the limitations of real-world cycling datasets, particularly data scarcity and trip-length imbalance, we propose two augmentation methods: Graph Stitching (GS), which combines segments of existing trips to form new trajectories, and Graphon-Inspired Trip Generation (GITG), which uses an empirically estimated transition kernel to simulate realistic trip patterns through probabilistic sampling. Despite limited improvements, this study establishes a foundation for future research in graph-based trajectory augmentation. Integrating richer trip-level features, such as dynamic environmental conditions or behavioral data, with structural augmentation could lead to more effective training data and improved model generalization.

1 Introduction

Data augmentation is a widely-used technique in machine learning designed to artificially increase the diversity and volume of training datasets. The approach is particularly beneficial in deep learning, where extensive data is typically required to train robust and generalizable models [20]. By generating new training samples through transformations such as rotation, scaling, cropping, and flipping, data augmentation mitigates overfitting, enhances the generalization capabilities of models, and improves model performance, especially in scenarios with limited labeled data [25].

A core challenge addressed by data augmentation is the scarcity and imbalance of datasets, which frequently hampers the effective training of machine learning models [6]. Real-world datasets are often small or skewed, leading to models that poorly generalize to new, unseen data [12]. Data augmentation techniques can partially overcome these issues by synthetically expanding datasets and balancing class distributions, thereby promoting robustness and reducing sensitivity to irrelevant variations in the input data.

A recent study proposing a graph-based deep learning model DG4b (Dual-Graph approach for Bicycle travel time estimation) [10] was trained on the cycling dataset SimRa [2]. This dataset suffers from this imbalance, where longer trips (>20 minutes) are underrepresented, see figure 1. This introduces an aspect to explore, wether or not this missing data negatively impacts model performance.

This imbalance could be caused by multiple variables. [14] found that urban trails primarily support short, utilitarian cycling trips, often replacing short car journeys. [8] supports this, stating that people prefer cars or public transport if the journey is perceived as too long or if the purpose of the journey includes running errands, suggesting that longer cycling trips become unpleasant if carrying luggage.

Another potential issue for the model could be data scarcity. The collection of cycling trajectory data presents significant challenges in terms of spatial and temporal representativeness. Data acquired from fitness tracking applications and GPS-enabled devices often reflects the behavior of a non-representative subset of cyclists, typically skewed toward more affluent, fitness-oriented users who engage with platforms such as Strava or Garmin Connect [23]. As a result, coverage tends to be concentrated along popular recreational routes or commuter corridors, while low-income neighborhoods, rural areas, and less-traveled infrastructure remain underrepresented [15]. This bias limits the generalizability of models trained on such data and may perpetuate inequities in cycling infrastructure planning.

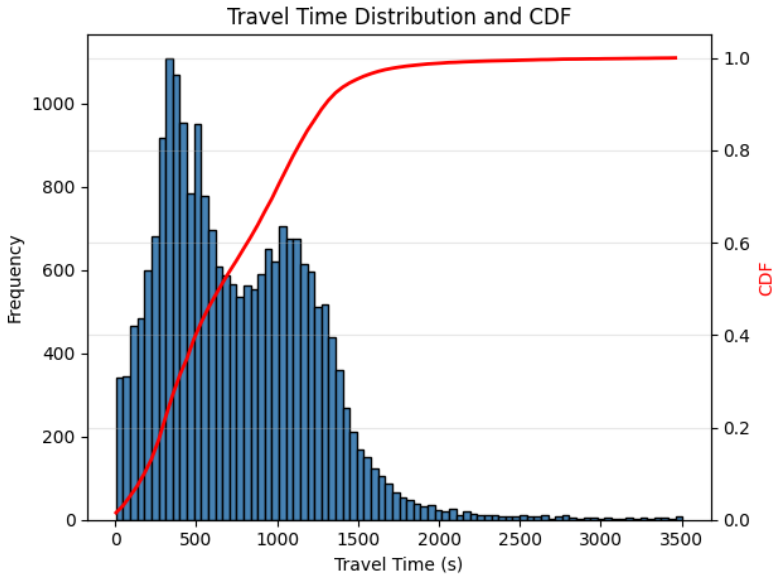


Figure 1: Trip length distribution in seconds

In addition to spatial and temporal limitations, ethical and privacy concerns pose substantial barriers to large-scale cycling data collection and use. GPS traces inherently contain personally identifiable information, particularly when ride start and end points coincide with home or workplace locations [18]. Even when anonymized, trajectory data can often be re-identified with relatively high accuracy [22]. This raises concerns under data protection frameworks such as the General Data Protection Regulation (GDPR) in the European Union, which mandates informed consent and the minimization of identifiable data [9].

Since the DG4b model takes trip specific graphs as input, the focus of this research will be on graph data augmentation. Several graph augmentation, like Graph Mixup [19] [13], apply Mixup [29] to graphs by transforming graphs into graphons [26]. We will discuss why these traditional techniques unfortunately can't be applied to our problem case.

In this research, we explore two different techniques to improve DG4b model performance on the SimRa dataset. We introduce a relatively trivial approach and a more data-oriented approach:

- Graph-stitching, combining two trip graphs creating a new one
- Graphon sampling, sampling new trips from a subset of trips via a graphon aggregate.

2 Background

This section discusses previous work that provides a foundation on which we will expand on.

2.1 DG4b

Dual-Graph approach for Bicycle travel time estimation is a deep learning model proposed in [10]. It seeks to estimate the travel time of bike trips in an urban environment. It has two graph components, a static road line graph and a trip-specific travel graph.

Definition 1 (Road Line Network):
We utilize a line graph representation to emphasize interactions between road segments, which also offers an effective way to model the road network. The road network is defined as an undirected line graph $G = (V, E)$, where V is the set of nodes and E is the set of edges. Each node $v_i \in V$ corresponds to a specific road segment, and each edge $e_{ij} \in E$ represents a connection (i.e., an intersection) between two adjacent road segments v_j and v_i.

Definition 2 (Trip graph): A trip T_i represents a travel route from a starting point to a destination within a network. The initial data for each trip is a series of GPS coordinates paired with timestamps. After applying map-matching[11] techniques, each trip-graph can be characterized by three components: $T_i = \{V_i, t_i, y_i\}$, where:

- $V_i = \{v_i^1, \ldots, v_i^n\}$ represents the sequence of road segments (or nodes) traversed during the trip;
- t_i = wether or not the trip took place during peak hours, $t_i \in [0, 1]$
- y_i = the total duration or the travel time of the trip.

2.2 Dataset

The trip data used in this study is derived from the SimRa dataset[1] [2], a crowdsourced mobility dataset collected via a mobile application installed by voluntary users. The dataset is geographically focused on the Berlin metropolitan area and comprises detailed spatiotemporal traces of bicycle trips. For consistency with prior work, we utilize data from January 2025, aligning with the original training period of the DG4b model. This facilitates direct comparability and ensures that domain-specific patterns present in the original training data are preserved.

The objective of this study is to integrate augmented trajectory data with the real-world SimRa dataset. As the SimRa trajectories have been preprocessed into a graph-based representation, the augmented samples are likewise transformed into graph format to enable seamless data fusion and model compatibility.

2.3 Graphons

Graphons are symmetric, measurable functions $W : [0, 1]^2 \to [0, 1]$ that serve as limit objects for sequences of graphs. Introduced by Lovász and Szegedy[17] in the theory of dense graph limits, graphons enable a nonparametric, probabilistic framework for modeling large networks. They provide a unified representation for various random graph models, including Erdös-Rényi and stochastic block models, and support the analysis of network convergence, sampling, and estimation. See figure 2 for a visualization of graphon.

Graphons are closely linked to the theory of exchangeable random graphs[7], where the graph distribution is invariant under permutations of node labels. This invariance makes

[1] `https://www.digital-future.berlin/en/research/projects/simra/`

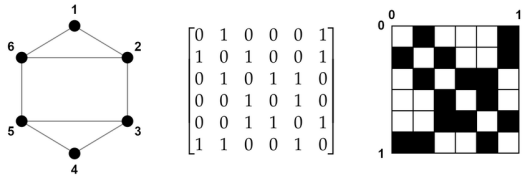


Figure 2: Graph G, adjacency matrix A_G and step graphon W_G, figures from [24]

graphons particularly useful in scenarios lacking node identity, such as population-level modeling or anonymized graph data.

Graphon Estimation. Multiple strategies have been developed to estimate graphons from observed graph data. Block-based methods[26] approximate the graphon using piecewise constant functions, effectively capturing community structure. Spectral techniques, such as Universal Singular Value Thresholding (USVT)[5], estimate graphons via low-rank approximations of the adjacency matrix and are especially effective when the underlying structure is smooth or low-rank. Kernel smoothing approaches[4] apply Gaussian filters to adjacency matrices sorted by empirical degree, assuming monotonic latent position functions.

Bayesian formulations treat the graphon as a latent function with a prior over random arrays[16], enabling uncertainty quantification and hierarchical modeling. More recently, Gromov-Wasserstein alignment has been used to compare and interpolate between graph structures without requiring aligned nodes, inspiring methods for data augmentation and representation learning[28].

Sparse Graphon Models. Real-world road networks are sparse, spatially embedded, and typically constrained by geographic factors. Traditional graphon models, while powerful for modeling exchangeable graphs [17, 3], assume node-permutation invariance and are typically designed for dense networks. This makes them poorly suited for representing road systems directly. However, their probabilistic structure and generative flexibility offer compelling foundations for hybrid modeling.

Graphons have become a foundational tool in modern graph theory and machine learning, supporting applications ranging from synthetic graph generation and data augmentation to structure-aware learning and probabilistic inference on networks.

3 Methodology

In this section, we discuss two methods of graph data augmentation applied to our problem.

3.1 Graph-Stitching

The more simple method we propose is Graph-Stitching (GS). Given a node N, we find two trips that traverse this node.

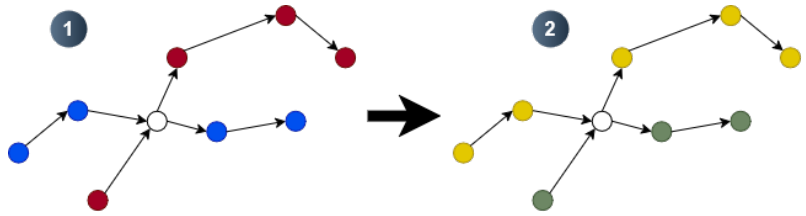


Figure 3: **Graph Stitching** 1. Two directed graph trips red and blue connect at a node. 2. Two new graphs could be sampled, yellow and green.

- For one of the trips T_i, we find the position of N in the sequence V_i and extract the subset $S_i = \{v_i^1, \ldots, v_i^k\}$, where $v_i^k = N$.
- For the other trip T_j, we extract $S_j = \{v_j^l, \ldots, v_j^n\}$ from V_j, where $v_j^l = N$.
- The two sequences are then stitched together, creating $S_{i+j} = \{v_i^1, \ldots, N, \ldots, v_j^n\}$

This gives us a basic form of graph data augmentation, where we combine two trips to create a new one.

3.2 Hybrid Graphon-Inspired Trip Generation via Empirical Estimation

To generate realistic and structurally coherent trip data, we propose a hybrid graphon-inspired model that integrates the generative flexibility of graphons with the spatial and topological constraints of real-world road networks. We will refer to this model as GITG (Graphon-Inspired trip generation). Unlike classical graphon models, which are designed for dense, exchangeable graphs[17], our approach operates on a fixed underlying road network and estimates a graphon-like transition kernel directly from observed trip data.

Let $G = (V, E)$ be a road network, where each node $v \in V$ corresponds to a spatial location (e.g., an road segment) and each edge $(u, v) \in E$ represents an intersection. We are given a set of observed trips $\mathcal{T} = \{T_1, \ldots, T_N\}$, each represented as a path or subgraph within G. To model the likelihood of movement between nodes, we define an empirical graphon $W_{\text{trip}} : V \times V \to [0, 1]$ as a normalized transition matrix based on the observed co-occurrence of consecutive node pairs in trips:

$$W_{\text{trip}}(i, j) = \frac{\text{freq}(i \to j)}{\sum_{k \in \mathcal{N}(i)} \text{freq}(i \to k)},$$

where $\text{freq}(i \to j)$ denotes the number of observed transitions from node i to j, and $\mathcal{N}(i)$ is the neighborhood of node i in G. This formulation yields a data-driven, localized connectivity kernel that captures directional travel preferences inherent in real trips.

New synthetic trips are sampled by initiating a random walk on G, starting from a node drawn according to the empirical distribution of trip origins, and advancing step-by-step using transition probabilities defined by W_{trip}. This walk is terminated based on a either stochastic stopping rule (maximum trip length) or when no further feasible transitions are available (e.g., reaching a dead-end in the network), resulting in a path that reflects empirical movement behavior while allowing for structural variability.

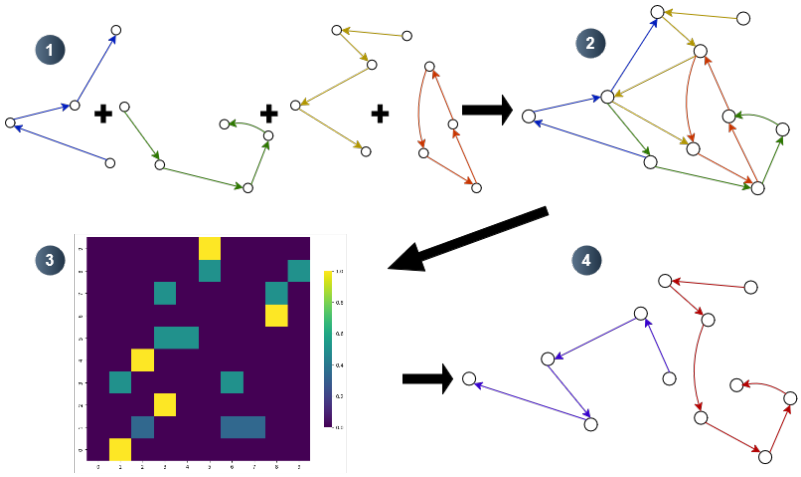


Figure 4: **Hybrid Graphon-Inspired Trip Generation via Empirical Estimation** 1. Select trip graphs 2. Combine into large graph, accounting for topological overlap 3. Compute graphon. This is a visualization of a graphon, where a lighter color indicates a higher probability of an edge being in a trip graph 4. Sample new trip graphs

This hybrid model benefits from the expressiveness of graphons[3] and the realism of spatial networks[1]. It also aligns with recent advances in trajectory modeling that integrate topological priors with data-driven inference[27], and serves as a practical augmentation framework for training and evaluating graph-based learning models on mobility data.

3.3 Setting Node Features

All nodes correspond to a road segment in the static road-line graph, with road segments varying in length. For each node in the artificial trip sequence, there are corresponding node features utilized to determine a trips travel time. The speed for which an the artificial cyclist travels over a road segment, is sampled from a Gaussian Kernel Density Estimate (KDE). KDE produces smooth continuous values and can handle multi-modality[21], proving ideal for our situation. The travel time for a node is set by multiplying the road segment distance, d_i, with the sampled speed, s_i, giving $t_i = d_i * s_i$. The DG4b model also incorporates contextual trip features, specifically the seasonal period (time of year) and whether the trip occurs during peak travel hours. These values are sampledfrom a random uniform distribution, where the intervals are $[0, 3)$ and $[0, 1)$ respectively.

4 Results and Discussion

In this section we will layout the experiments that we have executed to determine the effectiveness of our proposed graph data augmentation techniques. We analyze these results to provide insights into the underlying factors that explain the success or limitations of our proposed methods.

4.1 Evaluation Metrics

To assess the impact of our augmentation techniques, we will compare the model performance with the same evaluation metrics introduced by Gao [10]. Model performance is evaluated using four metrics: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Satisfaction Rate (SR). These metrics are defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{|T_i|}\sum_{i=1}^{|T_i|}(y_i - \hat{y}i)^2}, \tag{1}$$

$$\text{MAE} = \frac{1}{|T_i|}\sum_{i=1}^{|T_i|}|y_i - \hat{y}i|, \tag{2}$$

$$\text{MAPE} = \frac{1}{|T_i|}\sum_{i=1}^{|T_i|}\left|\frac{y_i - \hat{y}i}{y_i}\right| \times 100\%, \tag{3}$$

$$\text{SR} = \frac{1}{|T_i|}\sum_{i=1}^{|T_i|}\left(\left|\frac{y_i - \hat{y}_i}{y_i}\right| \leq 20\%\right) \times 100\%, \tag{4}$$

$$\tag{5}$$

where y^i and $\hat{y}^i$ denote the ground truth and estimated travel time for trip T_i, respectively, and $|T_i|$ is the total number of trips evaluated. The Satisfaction Rate (SR) measures the percentage of trips for which the relative error is within 20%.

RMSE penalizes large errors, indicating if outliers are present. MAE tells the average error, indicating a more general performance of the model. It is important to note that MAPE and SR are particularly sensitive to short trips. In such cases, even minor absolute deviations can lead to large relative errors, potentially skewing the evaluation. Therefore, while these metrics offer insight into relative performance, they should be interpreted with caution when analyzing trips of short duration.

4.2 Experiment set-up

After generated trips are augmented to the original data, we split the augmented data into a train and validation set. The test set remains untouched, to ensure each dataset performance receives fair evaluation. Furthermore, artificial data is only added to train and validation set, meaning that original data retains its original split. Each variation of the mentioned data augmentation techniques will generate a set of 10000 artificial trips. The DG4b model will always run with a learning rate of 0.001, a random seed of 42, and a training batch size of 1024.

4.3 Model performance on augmented data

In ths section we will be performing two different experiments to explore the influence and feasibility of our data augmentation techniques. First we will train the model on the original and augmented data, using 10000 augmented trips. For the second experiment, we generate 10000 trips while attempting to minimize the amount of short trips in the augmented data.

Table 1: Performance comparison for different augmented datasets and the original dataset. The best performing data is marked red and if an augmented dataset outperforms the original dataset it is marked in blue. A score is also marked blue if it equalizes the raw data performance (this is only applicable to SR)

Method	**Total**				**Short (<8 min)**				**Medium (8 â 16 min)**				**Long (>16 min)**			
	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR
DG4b Raw Data	473.82	145.05	18.05	69.0	82.54	56.24	23.5	59.0	154.97	111.32	15.86	72.0	793.39	265.5	14.61	76.0
GS	499.77	153.75	19.52	68.0	81.96	57.82	26.84	58.0	147.44	107.68	15.41	73.0	840.11	292.93	15.97	73.0
GS leave out short trips	474.64	151.06	20.31	67.0	86.43	59.88	29.18	57.0	151.96	110.86	15.88	71.0	794.95	280.02	15.52	74.0
GITG	467.42	148.68	19.45	68.0	89.89	57.22	26.48	59.0	158.22	114.34	16.36	71.0	780.75	272.47	15.26	75.0
GITG exclude short trips	471.85	148.3	19.08	69.0	84.87	56.43	25.75	60.0	153.72	111.4	15.95	72.0	789.91	274.89	15.28	76.0

For GS we do this by leaving out short trips from the sampling pool. In the case for GITG, we do this by setting the minimum amount of edges for an artificial trip to 100. With both techniques, short trips will still be generated but in smaller quantities. This caused by two reasons: 1. GS samples small node sequences from two different trips, resulting in a short end trip. 2. GITG's stopping condition can be met when there are no more edges to be sampled from. For augmented data trip distribution, see the Appendix.

We also compare the model performance on three different trip categories: short (<8 minutes), medium (8-16), and long (>=16 minutes), as defined by [10]. The results are displayed in table 3. Unfortunately, there is no dataset that outperforms the raw data on every metric at the same time.

4.4 Graph Stitching performance

When evaluating the performance metrics associated with the GS-generated trips, we observe a negative overall impact on model performance, with improvements limited primarily to medium-length trips. This improvement may be attributed to the relative underrepresentation of medium-length trips in the original dataset, suggesting that the augmentation process enhances the model's ability to generalize by expanding this subset.

In contrast, performance on longer trips deteriorates significantly. A plausible explanation is that GS-generated trajectories for these longer trips may include implausible road segment sequences or unrealistic route choices, which could introduce noise and reduce the model's ability to learn meaningful patterns. This is consistent with the increased complexity and contextual dependence of long trips, which are not adequately captured by the current augmentation approach.

Furthermore, the GS method does not yield noticeable improvements for short trips. This may be due to the high variability inherent in short cycling trips, which are more susceptible to external factors such as traffic signals, intersections, and stop-and-go behavior. As highlighted by Gao et al. [10], such factors introduce stochasticity that our augmentation method does not currently model, thereby limiting its effectiveness for this segment of the data.

4.5 Graphon-Inspired Trip generation performance

An analysis of model performance using GITG-augmented data reveals that GITG generally underperforms slightly relative to the raw dataset, although it demonstrates marginal improvements in certain cases. Notably, when GITG-generated trips are incorporated into the training set, we observe an improvement in RMSE for long trips, suggesting that the additional data does not introduce substantial outliers or anomalous trajectories that could destabilize the models learning process. This interpretation is further supported by the consistency of SR scores with those observed for the unaugmented data, indicating that the model maintains a reasonable approximation of travel times.

However, GITG consistently exhibits inferior performance on MAE and MAPE metrics, implying that the inclusion of generated trips does not enhance the model's capability for fine-grained or precise time estimations. Moreover, similar to the GS augmentation, GITG performs poorly on short trips. This likely stems from the shared approach used for synthesizing trip-level features-such as average speed and classification into (off-)peak hours-which does not account for external factors (e.g., traffic signals, stop frequency, or urban density) that disproportionately affect short-trip dynamics. Consequently, these limitations may hinder the model's ability to accurately learn from or generalize over short trip data.

4.6 Augmented Data as Sole Training Source

To determine the individual performance of our augmentation techniques, we run an experiment in which we remove the original raw data. For both GS and GITG we generate 20000 trips and train DG4b on these separately. We chose 20000 trips because this approximately resembles the amount of trips the original training set contains. The results can be seen in table 2. According to the results, the model performs better or equal on GITG data for all metrics when compared to training on only GS data. This can be explained by GITG producing data that exhibits greater variability when it comes to route selection, as GS trips consist of only two subtrips, limiting features such as trip length. The marginally better performance of GITG over GS suggests that GITG has greater potential to be a feasible graph data augmentation technique then GS.

Table 2: Performance comparison for training the DG4b model on only augmented data

Method	**Total**				**Short (<8 min)**				**Medium (8 â 16 min)**				**Long (>16 min)**			
	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR
GS Only augmented trips	649.26	246.52	29.18	33.0	111.92	93.67	34.67	23.0	213.04	181.33	26.07	35.0	1087.16	460.65	26.56	40.0
GITG Only augmented trips	646.92	243.69	28.82	33.0	110.66	92.53	34.27	23.0	210.87	179.2	25.77	36.0	1083.57	455.47	26.17	41.0

4.7 Using varying amounts of augmented trips

To further experiment with the augmentation techniques, we generated multiple sets of augmented data varying in size. The increased amount of trips gave similar results as discussed above. All model performance results and generated data distributions are displayed in the Appendix.

5 Responsible Research

he importance of conducting research responsibly and ethically is paramount, particularly in studies involving human mobility data. In this project, we have taken comprehensive measures to ensure that our research adheres to high standards of transparency, reproducibility, and integrity.

To support reproducibility, all experiments were conducted in a controlled and well-documented environment. Each step in the data processing, model training, and evaluation pipeline has been thoroughly recorded to allow for independent verification and replication of results. The data augmentation techniques introduced in this study were applied multiple times under consistent experimental conditions to assess the stability and robustness of their influence on model performance. This repeated experimentation ensures that reported outcomes are not artifacts of random variation or single-run anomalies.

All code, scripts, and generated data associated with this research are archived and can be made available upon request, enabling other researchers to replicate our methods, build upon our findings, or perform independent validations. This commitment to open scientific practice enhances the reliability of our conclusions and fosters collaborative advancement in the field.

In handling real-world cycling data, we recognize the significant ethical responsibility to protect user privacy. The dataset used in this study-the SimRa cycling dataset-has been preprocessed by its custodians to ensure full compliance with privacy regulations. According to the dataset providers, all personally identifiable information has been removed or obfuscated, and the data has been anonymized such that individuals cannot be reidentified [2]. As researchers, we have further refrained from implementing any procedures that could compromise this anonymization or reverse-engineer sensitive details.

By maintaining a strong emphasis on ethical data handling, privacy preservation, and research transparency, this study contributes to a responsible research culture that respects both scientific rigor and individual rights.

6 Conclusions and Future Work

The results of our experiments indicate that the proposed data augmentation techniques do not lead to overall improvements in the performance of the DG4b model for bicycle travel time estimation. While the Graph Stitching (GS) method demonstrates performance gains for medium-length trips, it exhibits reduced effectiveness for short and long trips, suggesting limited generalizability across the full distribution of trip lengths.

The Graphon-Inspired Trip Generation (GITG) approach achieves better average performance compared to GS, likely due to its probabilistic modeling framework, which facilitates the generation of synthetic trips with more realistic spatial patterns. Nevertheless, neither augmentation method results in substantial improvements in prediction accuracy or precision. Although the models trained on the augmented datasets are able to approximate travel times reasonably well, they consistently underperform in terms of fine-grained estimations, as reflected by elevated MAE and MAPE scores.

We attribute this shortfall to the limited treatment of non-routing contextual factors, such as traffic conditions, signal delays, and cyclist behavior, which are particularly influential in short trips and contribute significantly to travel time variability. Our current augmentation methods do not simulate these external influences, which likely constrains the model's performance.

Despite these limitations, the findings of this study open promising directions for future work. Specifically, future research could investigate integrating more realistic trip-level features; such as dynamic traffic data, elevation profiles, or stop frequency into the augmentation process. When combined with graph-based techniques like those explored here, such enhancements could yield synthetic datasets that are not only structurally plausible but also behaviorally rich, thereby improving model accuracy and generalization.

References

[1] Marc Barthélemy. Spatial networks. *Physics Reports*, 499(1-3):1–101, 2011.

[2] Digital-Future Berlin. "SimRa - Safety in Bicycle Traffic", 2024. Last accessed on 12/06/2025. Available: `https://www.digital-future.berlin/en/research/projects/simra/`.

[3] Christian Borgs, Jennifer Chayes, László Lovász, Vera T Sós, and Katalin Vesztergombi. Convergent sequences of dense graphs i: Subgraph frequencies, metric properties and testing. *Advances in Mathematics*, 219(6):1801–1851, 2008.

[4] Stanley Chan and Edoardo Airoldi. A consistent histogram estimator for exchangeable graph models. In *International Conference on Machine Learning*, pages 208–216. PMLR, 2014.

[5] Sourav Chatterjee. Matrix estimation by universal singular value thresholding. 2015.

[6] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[7] Persi Diaconis, Susan Holmes, and Svante Janson. Threshold graph limits and random threshold graphs. *Internet Mathematics*, 5(3):267–320, 2008.

[8] Jennifer Dill and Kim Voros. Factors affecting bicycling demand: initial survey findings from the portland, oregon, region. *Transportation Research Record*, 2031(1):9–17, 2007.

[9] European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation), 2016. Official Journal of the European Union, L 119, pp. 1â88.

[10] Ting Gao, Winnie Daamen, Elvin Isufi, Serge Hoogendoorn, and Senior Member IEE. Bicycle travel time estimation via dual graph-based neural networks. *LATEX CLASS FILES, VOL. 14*, 2021.

[11] Ting Gao, Winnie Daamen, Panchamy Krishnakumari, and Serge Hoogendoorn. Map-matching for cycling travel data in urban area. *IET Intelligent Transport Systems*, 18(11):2178–2203, 2024.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep feedforward networks. *Deep learning*, 1:161–217, 2016.

[13] Xiaotian Han, Zhimeng Jiang, Ninghao Liu, and Xia Hu. G-mixup: Graph data augmentation for graph classification. In *International Conference on Machine Learning*, pages 8230–8248. PMLR, 2022.

[14] Kevin J Krizek, Ahmed El-Geneidy, and Kristin Thompson. A detailed analysis of how an urban trail system affects cyclistsâ travel. *Transportation*, 34:611–624, 2007.

[15] Sven LiÃner and Stefan Huber. Facing the needs for clean bicycle data â a bicycle-specific approach of gps data processing. *European Transport Research Review*, 13(8), 2021. Discusses GPS noise, trip segmentation, and spatio-temporal errors in cycling data.

[16] James Lloyd, Peter Orbanz, Zoubin Ghahramani, and Daniel M Roy. Random function priors for exchangeable arrays with applications to graphs and relational data. *Advances in Neural Information Processing Systems*, 25, 2012.

[17] László Lovász and Balázs Szegedy. Limits of dense graph sequences. *Journal of Combinatorial Theory, Series B*, 96(6):933–957, 2006.

[18] Abhishek K. Mishra, Mathieu Cunche, and Heber H. Arcolezi. Breaking anonymity at scale: Re-identifying the trajectories of 100â¯k real users in japan. *arXiv preprint arXiv:2506.05611*, 2025. Demonstrates that standard mobility anonymization fails at country scale, highlighting re-identification risk.

[19] Madeline Navarro and Santiago Segarra. Graphmad: Graph mixup for data augmentation using data-driven convex clustering. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[20] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

[21] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*, volume 26 of *Monographs on Statistics and Applied Probability*. Chapman and Hall/CRC, 1986.

[22] Benedikt StrÃ¶bl and Alexandra Kapp. Investigating vulnerabilities of gps trip data to trajectoryâuser linking attacks. *arXiv preprint arXiv:2502.08217*, 2025. Shows significant re-identification risk even with anonymized cycling trip datasets.

[23] Tuuli Toivonen, Henrikki Tenkanen, Age Poom, and Maria Salonen. Comparing spatial data sources for cycling studies: a review. In Elias Willberg et al., editors, *Transport in Human Scale Cities*, pages 172–190. Edward Elgar Publishing, 2021. Compares biases in GPS and crowdsourced cycling data, highlighting spatial and temporal coverage gaps.

[24] Renato Vizuete, Paolo Frasca, and Federica Garin. Graphon-based sensitivity analysis of sis epidemics, 12 2019.

[25] Jason Wang, Luis Perez, et al. The effectiveness of data augmentation in image classification using deep learning. *Convolutional Neural Networks Vis. Recognit*, 11(2017):1–8, 2017.

[26] Patrick J. Wolfe and Sofia C. Olhede. Nonparametric graphon estimation, 2013.

[27] Yuxuan Yuan, Yu Zheng, and Xing Xie. Hetero-conv: Context-aware heterogeneous convolution for trajectory forecasting. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1526–1535. ACM, 2020.

[28] Zhichen Zeng, Ruizhong Qiu, Zhe Xu, Zhining Liu, Yuchen Yan, Tianxin Wei, Lei Ying, Jingrui He, and Hanghang Tong. Graph mixup on approximate gromov–wasserstein geodesics. In *Forty-first International Conference on Machine Learning*, 2024.

[29] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization, 2018.

A Appendix

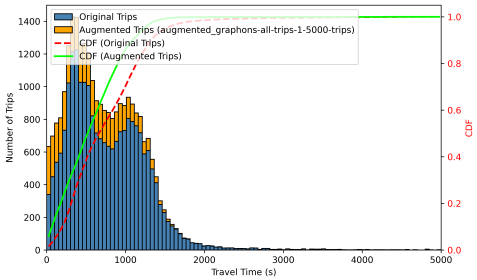
Table 3: Performance comparison for different augmented datasets and the original dataset. The best performing data is marked red and if an augmented dataset outperforms the original dataset it is marked in blue. A score is also marked blue if it equalizes the raw data performance (this is only applicable to SR)

Method	**Total**				**Short (<8 min)**				**Medium (8 â 16 min)**				**Long (>16 min)**			
	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR	RMSE	MAE	MAPE	SR
DG4b Raw Data	473.82	145.05	18.05	69.0	82.54	56.24	23.5	59.0	154.97	111.32	15.86	72.0	793.39	265.5	14.61	76.0
GS 5000 trips	468.05	147.71	19.34	70.0	87.53	57.4	26.81	61.0	152.24	109.24	15.66	73.0	783.23	274.17	15.25	75.0
GS 10000 trips	499.77	153.75	19.52	68.0	81.96	57.82	26.84	58.0	147.44	107.68	15.41	73.0	840.11	292.93	15.97	73.0
GS 20000 trips	470.87	158.79	22.16	65.0	100.81	66.08	32.58	55.0	166.73	119.31	17.02	70.0	783.84	288.61	16.48	70.0
GITG 5000 trips	472.39	149.71	19.85	68.0	89.44	58.99	27.46	59.0	159.86	117.54	16.84	69.0	789.22	270.74	14.98	77.0
GITG 10000 trips	467.42	148.68	19.45	68.0	89.89	57.22	26.48	59.0	158.22	114.34	16.36	71.0	780.75	272.47	15.26	75.0
GITG 20000 trips	458.01	154.26	21.96	66.0	105.73	67.64	33.18	54.0	166.5	117.42	16.78	71.0	760.63	275.49	15.5	74.0
GS Exclude short trips - 5000 trips	473.22	149.96	19.94	68.0	94.23	59.11	28.04	58.0	153.07	110.98	15.97	72.0	791.37	277.44	15.48	75.0
GS Exclude short trips - 10000 trips	474.64	151.06	20.31	67.0	86.43	59.88	29.18	57.0	151.96	110.86	15.88	71.0	794.95	280.02	15.52	74.0
GS Exclude short trips - 20000 trips	476.71	150.21	19.93	68.0	88.7	58.36	28.21	58.0	148.28	109.31	15.62	72.0	798.96	280.49	15.6	74.0
GITG Exclude short trips - 5000	495.47	149.1	20.04	68.0	86.11	60.98	29.59	55.0	146.44	106.5	15.28	74.0	832.36	277.2	14.88	77.0
GITG Exclude short trips - 10000 trips	471.85	148.3	19.08	69.0	84.87	56.43	25.75	60.0	153.72	111.4	15.95	72.0	789.91	274.89	15.28	76.0
GITG Exclude short trips - 20000 trips	502.4	160.45	20.9	65.0	89.62	62.51	29.2	57.0	157.88	114.71	16.41	70.0	842.16	301.32	16.74	70.0

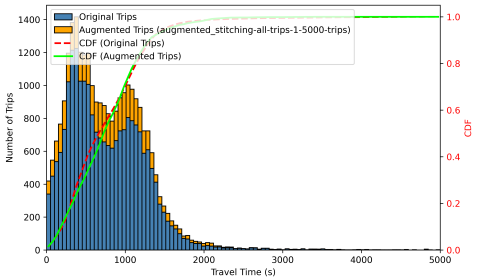
Below are the time travel distributions for the generated datasets.

1. For the first six graphs, we generate 5000, 10000, and 20000 trips for both GITG and GS, without constraints

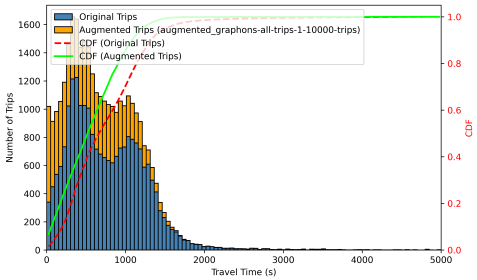
2. For the next six graphs, we attempt to minimize the amount of short trips. For GS, we remove short trips from the GS pool. This means that only stitch medium and long trips together. For GITG, we set the minimum amount of edges to be generated to be 100.



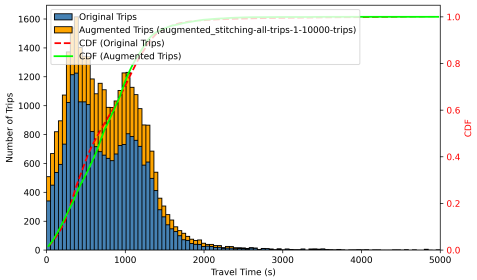
(a) GITG, generating 5000 trips



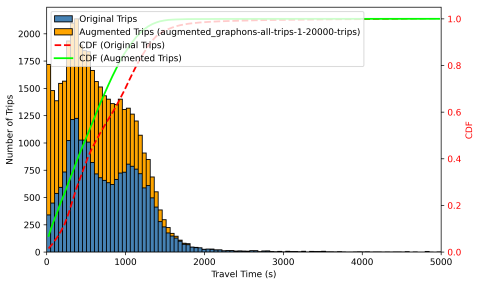
(b) GS, generating 5000 trips



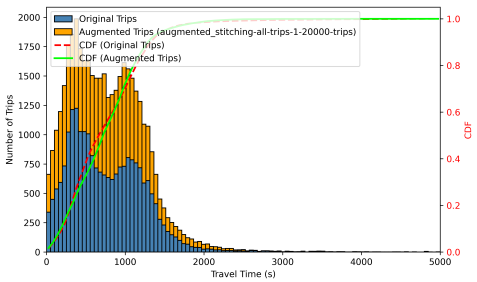
(c) GITG, generating 10000 trips



(d) GS, generating 10000 trips

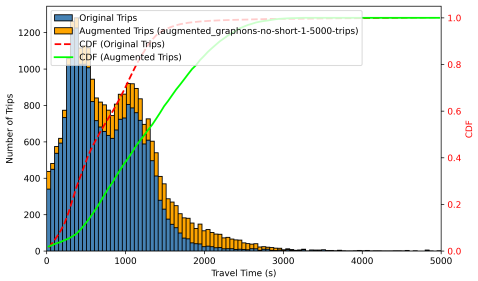


(e) GITG, generating 20000 trips

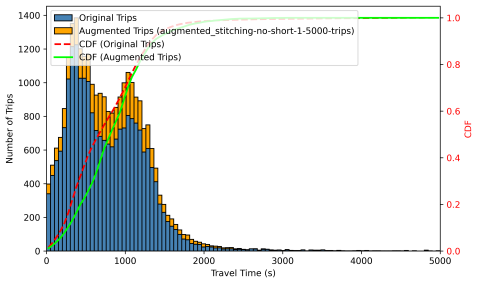


(f) GS, generating 20000 trips

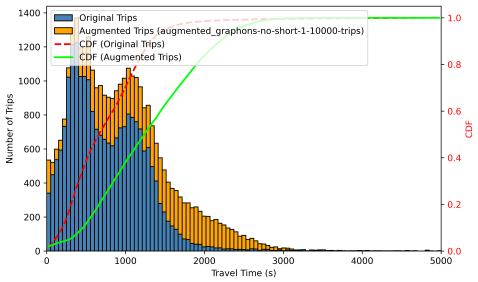
Figure 5: Time travel distributions for generated data



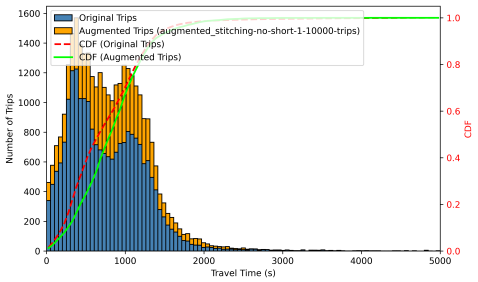
(a) GITG 5000 trips



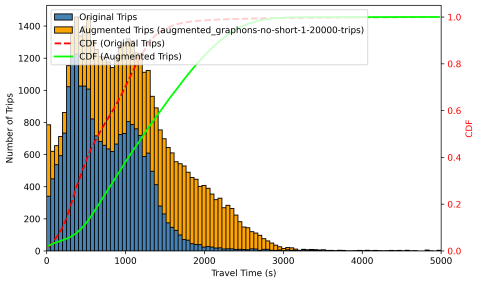
(b) GS 5000 trips



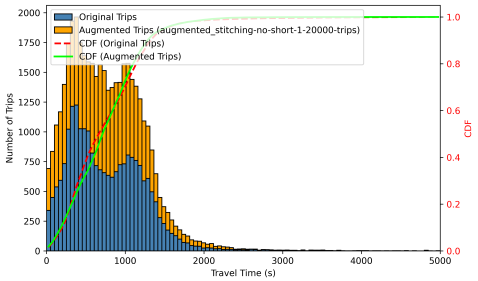
(c) GITG 10000 trips



(d) GS 10000 trips



(e) GITG 20000 trips



(f) GS 20000 trips

Figure 6: Time travel distributions for generated data where short trips are excluded from the sampling pool