Graph2RETA: Graph Neural Networks for Pick-up and Delivery Route Prediction and Arrival Time Estimation

Wilson Sentanoe[1], Sreyashi Saha[1], Yasas Diyanananda[1], Aqsa Manzoor[1], Buddhika Dasanayake[1], and Daniela Thyssens[2]

[1] University of Hildesheim, Germany
{sentanoe,sahas,diyanananda,manzoor,dasanayake}@uni-hildesheim.de
[2] Institute for Computer Science, University of Hildesheim, Germany
thyssens@ismll.uni-hildesheim.de

Keywords: Pick-up and Delivery · Route Prediction · Arrival Time Prediction.

Abstract. This research proposes an effective way to address the issues faced by pick-up and delivery services. The real-world variables that affect delivery routes are frequently overlooked by traditional routing technologies, resulting in differences between intended and actual trajectories. Similarly, the issue of forecasting the Estimated Time of Arrival involves unique challenges due to its high dimensionality. We suggest an integrated predictive modeling methodology that tackles routing prediction in a dynamic environment and ETA prediction at the same time to overcome these difficulties. Our method, Graph2RETA, uses a dynamic spatial-temporal graph-based model to forecast delivery workers' future routing behaviors while integrating route inference into ETA prediction. Graph2RETA leverages rich decision context and spatial-temporal information to improve the prediction accuracy of the concurrent state-of-the-art while capturing dynamic interactions between workers and timesteps by incorporating the underlying graph structure and features.

1 Introduction

Pick-up and delivery (P&D) services, such as food delivery and logistics, have expanded rapidly in recent years, providing exceptional convenience in the face of urbanisation and changing consumer needs. Despite significant investment in routing systems by service providers, practitioners have observed significant differences between planned and real worker routes [5,20]. Studies undertaken in some parts of the world, including the United States and Mexico [11], have highlighted how common this discrepancy is and how crucial correct route prediction is for follow-up activities, such as order dispatching and arrival-time forecasting. These findings emphasise the vital importance of route prediction approaches that effectively reflect workers' dynamic routing behaviours in real-world circumstances.

In the field of logistics, it is imperative to estimate package arrival times with precision in order to maintain both customer satisfaction and operational efficiency. Due to its multi-destination and path-free nature, ETA prediction in logistics presents distinct challenges compared to standard ETA prediction problems seen in other fields. Additionally, the unpredictability of couriers' pick-up routes, which are impacted by variables such as traffic and individual preferences, makes it more difficult to calculate ETA accurately. Meeting the demand for real-time ETA prediction in such complex situations continues to be a challenging and important task.

Our proposed method, Graph2RETA, is an integrated predictive modeling approach designed to tackle these issues in a comprehensive manner. This novel method optimizes routing while also predicting ETA by combining dynamic spatial-temporal graph-based modeling and route inference. Graph2RETA improves the prediction accuracy by utilizing the built-in graph structure and features to capture the dynamic interactions between problem occurrences and to incorporate a broad range of decision context and spatial-temporal information. We show that combining the training procedures of both models and integrating the training losses into both predicting and forecasting tasks is beneficial.

The experiments on a real-world industrial dataset show that Graph2RETA outperforms the prevalent State of the Art (SotA) approaches (Graph2Route and RankETPA). We expect our integrated method to offer the potential for service providers to gain more control over their delivery process, resulting in higher customer satisfaction and operational efficiency in P&D services.

2 Related Work

This section will discuss the current SotA for the problems addressed in this research: the Pick-up and Delivery Route Prediction (PDRP) and Estimated Time of Arrival (ETA) tasks.

2.1 Pick-up and Delivery Route Prediction

The domain of Pick-up and Delivery Route Prediction (PDRP) plays a crucial role in optimizing the logistics and operational efficiency of delivery services. The essence of PDRP lies in predicting the future routes of a worker based on their past behaviors, focusing on the actual route to be taken rather than identifying the most optimal path. This distinction is critical for accurately computing downstream tasks, such as arrival time prediction [19]. Despite its importance, the PDRP problem has yet to be extensively explored within the research community, marking a significant gap in the existing literature [19].

Early attempts to address route prediction in delivery settings include methodologies like OSquare [23] and FDNET [5], alongside recent developments such as Graph2Route [19]. OSquare emerged as one of the pioneering methods, adopting a next-stop prediction approach to sequentially construct routes using conventional machine learning techniques [23]. FDNET further advanced the route prediction field by incorporating the deepFM model [6] for input feature encoding, an LSTM layer for sequential data processing, and attention mechanisms to estimate the probability of a worker visiting subsequent locations [5]. However, both OSquare and FDNET primarily conceptualize unfinished tasks as a sequence, which limits their capacity to encode complex spatial-temporal dynamics between different tasks. It often results in predicting implausible routes, relying solely on information available at the time of request [19].

Graph2Route [19] relies on graph-based information for predicting the routes. It uses a dynamic spatial-temporal graph encoder and a personalized route decoder to enhance prediction quality. This approach models spatial-temporal correlations through a Graph Convolutional Network and Recurrent Neural Networks. The personalized node decoding process employs a masking mechanism to eliminate unlikely solutions, streamlining the decision-making process and enhancing model performance. It leverages massive historical data to learn and model the worker's personalized decision preferences, making it the current SotA model for route prediction and the perfect baseline to improve upon in this work.

2.2 Estimated Time of Arrival

In pick-up and delivery services, predicting the Estimated Time of Arrival (ETA) of couriers plays a crucial role in enhancing service reliability and customer satisfaction [5]. ETA prediction can be defined as forecasting the travel time from an origin to a destination along a specified path. This problem has garnered attention across various domains, including ride-sharing [4,17], logistics [22,1], and food delivery services [25], highlighting its widespread relevance.

ETA prediction methodologies can broadly be categorized into path-free and path-based approaches. Path-free methods, such as the ST-NN proposed by Jindal et al. [7], rely solely on the origin and destination points without incorporating route information. These models predict travel distance and time by also considering additional temporal data. Conversely, path-based methods involve a more detailed analysis by incorporating specific travel routes into the prediction process. For instance, the Wide and Deep Route (WDR) model [18] integrates linear models, deep neural networks, and recurrent neural networks to predict travel times more accurately. Similarly, CompactETA [4] employs a graph attention network and positional encoding to learn the spatio-temporal dependencies of road networks and the sequence of travel routes, respectively.

Path-based models generally surpass their path-free counterparts due to their detailed route information, providing a richer context for more accurate predictions. Our study adopts a path-based approach, aligning with our objective to predict couriers' routes in the initial phase of the research. We utilize the Arrival Time Predictor from RankETPA [21] as a foundation for our work. RankETPA stands out with its spatial-temporal attention-based mechanism. It leverages predicted routes, node information, package details, and courier features to estimate precise arrival time.

The methodology behind RankETPA begins with calculating the first positional encoding of the route sequence derived from the previously predicted routes. This encoding captures the relative positioning of nodes within the route sequence. Subsequently, this data, along with pick-up order encodings, package characteristics, courier details, and other external factors, form the comprehensive input to the model. The RankETPA model then utilizes spatial-temporal attention-based encoder blocks to encapsulate the dynamics between unpicked package pairs and other pertinent information through layers comprising multi-head attention mechanisms and feed-forward networks. The multi-head attention effectively captures the spatial-temporal correlations among packages, while the feed-forward network enhances the model with non-linear transformations, culminating in the accurate prediction of arrival times for each node in the sequence.

3 Problem Formulation

In this section, we formulate the route prediction task from the graph perspective and the arrival time estimation problem.

3.1 Route Prediction

The problem is defined on a graph, where the nodes are each task of the driver and the edges are the spatial and temporal distance between the nodes. The following adopts the notation in [19]. Given the worker w's input graph $\mathcal{G}$ at a given time t

$$\mathcal{G}_t^w = (\mathcal{V}_t, \mathcal{E}_t, X_t, E_t) \tag{1}$$

the main objective is to learn a mapping function F_C to predict the worker's future route $\hat{\pi}$ which should satisfy the given route constraints C, formulated as:

$$\mathcal{F}_C(\mathcal{G}_t^w) = \pi_1, \pi_2, ..., \pi_{|\mathcal{V}_t^U|} \tag{2}$$

here π_i means that the i-th node in the route is node v_{π_i} and $|.|$ denotes the cardinality of a set. Moreover, $\pi_i \in \{1, \dots, |V|\}$ and $\pi_i \neq \pi_j$ if $i \neq j$ where:

$$\mathcal{V}_t^U = \{v | v \in \mathcal{V}_t, \mathrm{X}^{FT} = -1\} \tag{3}$$

are the unfinished nodes and x^{FT} is task finish time. Such that:

$$\min \mathcal{L}_{G2R} = -\sum w \in W \sum_{t \in T} \sum_{i \in \pi_{t:t'}} y_i \log(p(y_i|\theta)) \tag{4}$$

3.2 Arrival Time Estimation

The problem formulation of the arrival time estimation is adopted from [21]. Given the input query $q = \{c, t, V_t^u\}$ and $\hat{\pi} = \{\pi_1, \pi_2, ..., \pi_{|\mathcal{V}_t^U|}\}$ where c is the courier ID and t is the query time, our task is to use a model $\mathcal{F}$ that can map the input query to the pick-up arrival time $\hat{y}$ for the unpicked-up package set:

$$\hat{y} = \mathcal{F}(q, \hat{\pi}) \mapsto Y = \{y_1, ..., y_n\} \tag{5}$$

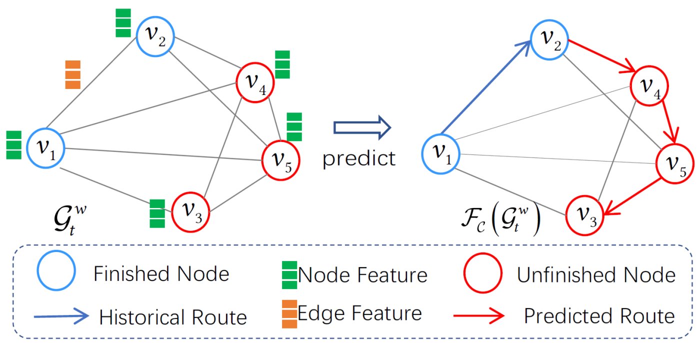

Fig. 1: Problem Illustration [19]. In this case, $V_t^F = \{v_1, v_2\}$ and $V_t^U = \{v_3, v_4, v_5\}$, the output of the model $\hat{\pi} = \{\pi_1, \pi_2, \pi_3\}$ is [4, 5, 3].

such that minimize the loss of the predicted arrival time and the ground truth, which can be formally represented as follows given the Huber loss function

$$\mathcal{L}_{ETPA} = \begin{cases} \frac{1}{2}(y - \hat{y})^2 & |y - \hat{y}| \leq \delta, \\ \delta|y - \hat{y}| - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases} \tag{6}$$

4 Methodology

We propose Graph2RETA, an upgraded version of the vanilla Graph2Route, by extending the model's capabilities to include arrival time prediction for pick-up and delivery tasks. This section outlines how the encoder module of the Graph2Route is upgraded and how the arrival time prediction module is integrated. Additionally, we will describe the dataset used in our experiments, the training procedure, and the evaluation protocol.

4.1 Updating the Route Prediction Module

Our first objective of this research is to refine the route prediction capabilities of the existing Graph2Route model introduced by Wen et al. [19]. Graph2Route is a foundational architecture for our research, particularly its spatial-temporal graph encoder component. It is the key to capturing the complex spatial and temporal context within the graph structure of pick-up and delivery route prediction problems. The original Graph2Route employs graph convolutional layers (GCN) and gated recurrent units (GRU) for encoding spatial and temporal correlations, respectively. These components have been shown to effectively learn the graph's spatial-temporal correlations, underscoring their importance in the architecture's performance [19].

However, GCN and GRU are considered conventional and outdated approaches at the time of the research, which may not fully capture the dynamic complexities of spatial-temporal graph data. Recent literature suggests

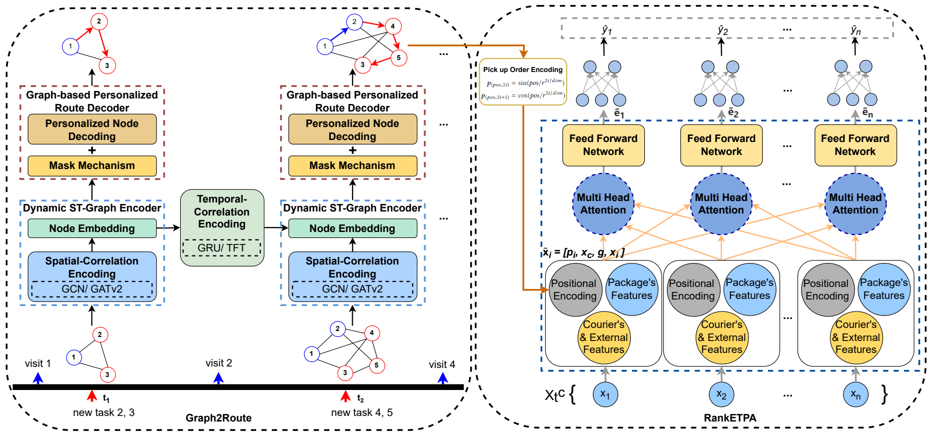

Fig. 2: Graph2RETA Model Architecture [19,21]

that advancements in graph learning techniques offer more sophisticated methods for capturing these relationships [24]. Recognizing this potential issue on the Graph2Route architecture, our research proposes to exchange the spatial encoder module to the well-established attention mechanism module for graphs, namely GAT[1].

Regarding temporal encoding, the GRU module in Graph2Route still encounters challenges with gradient issues and may not effectively capture long-term dependencies [10]. The Transformer architecture, renowned for its attention mechanisms and ability to manage sequences with long-term dependencies, offers promising benefits [16]. We employ the Temporal Fusion Transformer (TFT) model, which combines gating mechanisms, variable selection, and interpretable multi-head attention, for the temporal encoder [12]. However, we will adapt TFT into Graph2RETA by excluding its output layer to maintain compatibility with the original Graph2Route decoder, thereby focusing on the temporal variable interdependence learning.

4.2 Arrival Time Prediction Module

The second part of the work is to take a step forward by introducing an arrival time prediction module, which serves as a downstream task that leverages the output of our route predictor. This approach aims to estimate the time when a courier will come to pick up and deliver packages at each location within the route, commonly referred to as the ETA, and introduces additional feedback on the routing prediction. Drawing upon the methodologies from RankETPA [21],

[1] Our experiments are conducted with a variant of the graph attention network called GATv2 [2], which uses an attention mechanism that dynamically weighs the importance of neighbouring nodes [3] and offers slightly better results than the original GAT architecture.

we seamlessly integrate a spatial-temporal attention-based arrival time prediction with our refined route prediction module.

The underlying principle of Graph2RETA hinges on the sequence of operations where the courier's route is determined initially through the route predictor module. Following this, the arrival time predictor module takes center stage, utilizing the precomputed route to ascertain the arrival times at each designated stop. This sequential processing is grounded in the logic that the ETA is intrinsically linked to the order of stops on the courier's itinerary, underscoring the importance of route prediction as a precursor to time estimation [21,5].

The operational flow begins with the route predictor outlining a sequence of nodes, indicating the courier's path at a specified time step. Subsequently, this predicted route, enriched with additional node and courier data such as the courier's ID, average speed, and maximum load capacity, serves as input to the arrival time predictor module. This module then begins calculating the ETA for each node along the route, providing a comprehensive overview of the delivery timeline. By integrating the arrival time predictor module, our framework now returns two outputs: the predicted delivery route and the ETA for each stop along the way. This improvement not only upgrades the practical use of our model but also supplements it with a layer of temporal insight, offering a more complete view of the courier's journey.

4.3 Dataset

This work utilizes the Food Pick-up and Delivery Data (Food-PD) [19], a dataset composed of online food pick-up and delivery transactions facilitated by Eleme, a prominent online food delivery service in China. This dataset plays a crucial role in our experiment, providing a real-world context for our predictive modeling efforts.

The Food-PD dataset encompasses the pick-up and delivery activities of 916 delivery workers over a span of 28 days in Dalian, China. It offers a comprehensive view of the operational dynamics of food delivery, making it an invaluable resource for our analysis. The primary objective of employing this dataset is to forecast these drivers' future pick-up and delivery routes. We conceptualize each driver as a worker, and each pick-up or delivery task associated with an order is represented as a node.

The dataset encompasses 5039 instances meticulously partitioned into training, testing, and validation sets. The distribution follows a 50:25:25 ratio [19], ensuring a balanced division that facilitates both the development and the rigorous evaluation of the Graph2RETA model.

4.4 Model Training

We explore two strategies for training Graph2RETA, which consists of a route predictor and an arrival time prediction module. The initial approach entails training each module separately. Here, the focus is on first improving the accuracy of the route predictor by training it to minimize the cross entropy loss function (eq. 4). Upon completion, the predicted routes are exported and stored, before being preprocessed to serve as input for the ETA task. This subsequent module is trained to minimize the Huber loss (eq. 6), optimizing it for accurate time estimations. This sequential training process allows each module to refine its predictions independently.

$$\mathcal{L}_{Joint} = \alpha \times \mathcal{L}_{G2R} + (1 - \alpha) \times \mathcal{L}_{ETPA} \tag{7}$$

Conversely, our second strategy adopts an integrated approach, wherein both modules are integrated into a unified architecture and trained concurrently, which we call joint training. Unlike the first approach, the joint training trains both models simultaneously in one training loop, allowing both modules to learn from shared parameters. This integrated approach necessitates formulating a novel loss function, as stated in equation 7, that combines the cross entropy loss from the route prediction with the Huber loss from the time prediction. The fusion of these loss functions is balanced using a weight variable, α, as detailed in [15], which harmonizes the contributions of the two distinct tasks.

Initially, we assign α a value of 0.5, signifying equal importance to both route and time predictions. However, recognizing the potential benefits of temporal prioritization in training, we also experiment with a dynamic α value. This approach starts with an α of 0.9, emphasizing the route predictor's role in the early stages of training, thereby allowing it to establish a solid predictive foundation rapidly. As the training progresses, α is gradually reduced to 0.1, incrementally shifting the focus towards refining the arrival time predictions. This dynamic adjustment effectively uses both modules' strengths, fostering a symbiotic development where initial route accuracy enhances the subsequent time predictions. In the final result, we are showing results from this dynamic adjustment approach.

4.5 Evaluation Metrics

To thoroughly evaluate the performance of Graph2RETA, we have adopted a comprehensive set of metrics tailored to assess both the predicted route and the estimated arrival time of couriers.
For assessing the predicted routes, our evaluation metrics include:

- Kendall Rank Correlation (KRC) [8]: This metric measures the ordinal association between the predicted and actual sequences, which is defined as:

$$\mathrm{KRC} = \frac{N_c - N_d}{N_c + N_d}, \tag{8}$$

  where N_c and N_d are the numbers of concordant and discordant pairs respectively.
- Edit Distance (ED) [14]: It quantifies the dissimilarity between two sequences by counting the minimum number of operations required to transform the predicted sequence into the actual sequence.

- Location Square Deviation (LSD): This measures the extent of deviation between the predicted and actual locations. The LSD is defined as:

$$\mathrm{LSD} = \frac{1}{m}\sum_{i=1}^{m}(O_{\pi}(\pi_i) - O_{\hat{\pi}}(\pi_i))^2 \tag{9}$$

- Hit-Rate@k (HR@k): This evaluates the accuracy of the top-k items in the predicted sequence by determining how many match the actual sequence, formulated as:

$$\mathrm{HR}@k = \frac{\hat{\pi}_{[1:k]} \cap \pi_{[1:k]}}{k} \tag{10}$$

- Accuracy@k (ACC@k): It examines whether the route composed of the top-k predictions is precisely the same as the actual route, formulated as:

$$\mathrm{ACC}@k = \prod_{i=0}^{k}\mathbb{I}(\hat{\pi}_i, \pi_i), \tag{11}$$

  where $\mathbb{I}(\cdot)$ is the indicator function, and $\mathbb{I}(\hat{\pi}_i, \pi_i)$ equals 1 if $\hat{\pi}_i == \pi_i$ else 0.

The KRC, ED, and LSD assess the overall similarity between the predicted and actual routes, while HR@k and ACC@k focus on local similarity, offering an understanding of the model's performance from both global and local perspectives.

For the evaluation of the estimated arrival time, we utilize:

- Root Mean Square Error (RMSE): This metric calculates the square root of the average squared differences between the predicted and actual arrival times. The RMSE is defined as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \tag{12}$$

- Mean Absolute Error (MAE): It measures the average magnitude of the errors in a set of predictions without considering their direction, defined as:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i| \tag{13}$$

5 Experiments

This section outlines the experiment results demonstrating the enhancements achieved with Graph2RETA. We present the outcomes of individually trained models for route and arrival time estimation, along with the results from the joint training of both modules. The code to reproduce the results of this work is available at `https://github.com/wsentanoe/Graph2RETA`.

5.1 Route Prediction

Table 1 shows that replacing the spatial and temporal encoder components of the route prediction model with GAT and TFT would lead to improved results compared to the original GCN and GRU combination. However, we also evaluate each combination of the spatial and temporal components to find the one with the best predictive performance.

We observe that there is only a slight improvement by substituting GCN with GAT, especially while maintaining the GRU component. Although minimal, this variant still shows some improvement compared to the baseline Graph2Route architecture. On the other hand, replacing both components of Graph2Route with GAT and TFT leads to enhanced performance across all evaluation metrics compared to the baseline but remains inferior to the GCN+TFT variant.

In contrast to substituting GCN with GAT, there is a considerable improvement when TFT replaces GRU. Integrating the TFT component with GCN could significantly improve the prediction compared to the baseline Graph2Route results in all the metrics considered. TFT captures complex temporal dependencies and important time steps with its temporal attention mechanism, resulting in a 2.8% improvement in KRC, which is the primary evaluation metric of the route prediction model. Improvement goes to as high as 24.8% in ED metric compared to baseline because of the precise temporal modeling and the ability to handle multi-horizon forecasting of TFT, which predicts routes that are geographically closer to the actual routes, thereby lowering the ED. These results set the GCN+TFT as the best variant for the route prediction. Therefore, we decided to employ them for the final Graph2RETA architecture.

Table 1: Route Prediction Results. †Results are from [19] with the original GCN + GRU modules. The results are averaged over five runs. The best results are highlighted in **bold**. ↑ indicates the higher, the better and ↓ indicates the lower, the better.

Model Variant	HR@1 ↑	ACC@3 ↑	KRC ↑	LSD ↓	ED ↓
Graph2Route†	0.679	0.537	0.786	0.520	1.130
GCN + TFT	0.724 ± 0.009	**0.620 ± 0.012**	**0.808 ± 0.007**	**0.450 ± 0.026**	**0.850 ± 0.034**
GAT + GRU	0.674 ± 0.008	0.582 ± 0.006	0.788 ± 0.004	0.454 ± 0.008	0.902 ± 0.010
GAT + TFT	**0.729 ± 0.013**	0.615 ± 0.017	0.803 ± 0.011	0.466 ± 0.024	0.861 ± 0.041

Overall, the route prediction performance is improved when TFT is integrated in the model directing to the observation that temporal component in the model have more impact in the predictions.

5.2 Arrival Time Estimation with Joint Training

We aimed to assess whether jointly training the model would enhance its outcomes. In Table 2, we have obtained results from comparing the performance of the extended model with arrival time estimation, both trained individually and jointly.

It is evident that both models benefit from the joint training approach. Graph2RETA, when jointly trained, demonstrates an improvement of 8.4% in MAE and 11.2% in MSE for arrival time prediction compared to individual training settings, emphasizing the significance of joint training. The individually trained model excels in route prediction, whereas arrival time estimation significantly benefits and improves when jointly trained. There is an improvement of 8.7% in MAE and 11.3% in MSE when comparing the jointly trained Graph2RETA model to the individually trained Graph2Route model.

We conducted additional experiments on joint training to validate the proposed model. During joint training, we faced the challenge of dealing with two loss functions for route prediction and arrival time prediction, each operating on different scales. To address this issue, we scaled down the loss of arrival time prediction by 10 to ensure that both losses are within the same value range.

Table 2: Graph2RETA Experiment Results. ‡Graph2Route (GCN + GRU) results are from our implementation and are extended with the ETA module. Each module (Route prediction & ETA) is trained separately in the individually trained approach, while the jointly trained model trains both simultaneously. The results are averaged over five runs. The best results are highlighted in **bold**. ↑ indicates the higher, the better, and ↓ indicates the lower, the better.

		Trained Individually		Trained Jointly	
		Graph2Route‡	Graph2RETA	Graph2Route‡	Graph2RETA
Route Prediction	HR@1 ↑	0.674 ± 0.007	**0.724 ± 0.009**	0.650 ± 0.012	0.717 ± 0.008
	ACC@3 ↑	0.581 ± 0.007	**0.620 ± 0.012**	0.550 ± 0.012	0.601 ± 0.007
	KRC ↑	0.784 ± 0.004	**0.808 ± 0.007**	0.767 ± 0.007	0.801 ± 0.004
	LSD ↓	0.465 ± 0.008	**0.450 ± 0.026**	0.499 ± 0.016	0.459 ± 0.007
	ED ↓	0.902 ± 0.015	**0.850 ± 0.034**	0.997 ± 0.026	0.918 ± 0.018
Arrival Time Estimation	RMSE ↓	13.462 ± 0.208	13.446 ± 0.055	12.478 ± 0.616	**11.936 ± 0.422**
	MAE ↓	10.136 ± 0.171	10.110 ± 0.148	9.648 ± 0.595	**9.258 ± 0.294**

6 Discussion

Replacing the spatial component of Graph2Route from GCN to GAT does not significantly impact the route prediction performance, especially when trained individually, which is shown by the minimal improvement in the result. On the other hand, replacing the temporal component with TFT does significantly improve the performance of the route prediction module. The temporal component's contribution to the final outcome surpasses that of the spatial component.

When trained individually, the quality of the arrival time prediction aligns with the route prediction's performance. Improvement in route prediction corresponds to enhancement in arrival time prediction. The GCN+TFT configuration, which establishes consistent results across most route prediction metrics, also yields the best performance on Arrival Time Estimation when jointly trained. Hence, our intuition to integrate in Graph2RETA. This observation emphasizes our rationale for integrating this configuration in Graph2RETA.

Joint training does not necessarily improve the route prediction performance. This phenomenon suggests a nuanced challenge within multi-task learning frameworks, known as negative transfer [9,13]. Negative transfer in multi-task learning refers to a scenario where the knowledge learned from one task interferes with the learning or performance of another task. However, it definitely improves the arrival time prediction, which is the more important downstream task for delivery service or logistics operators.

Arrival time estimation benefits from joint training due to its dependency on route prediction. Also, the quality of the route prediction component's output significantly impacts the accuracy of arrival time estimation. Conversely, the route prediction task may not necessarily benefit from the joint training approach, as evidenced by its sub-optimal performance.

Moreover, it was evident that implementing a scheduling mechanism for alpha in joint training can further enhance the accuracy of the ETA estimation.

7 Conclusion

In this work, we propose Graph2RETA, which seamlessly integrated two complex models for route and arrival time prediction into a unified architecture, revealing the predominant role of temporal over spatial components in route prediction accuracy. Our findings demonstrate that updating the temporal component in the route prediction model contributes significantly to the increase in performance, especially when the route predictor operates independently from the arrival time prediction module. This finding suggests the importance of temporal dynamics in delivery route optimization. Joint training of both models could also improve the accuracy of the arrival time estimation despite the stagnant results in route prediction. This disparity in performance highlights the challenge of negative transfer in multi-task learning. Joint model training may offer specific benefits for arrival time estimation, paving the way for future research in delivery service optimization.

References

1. de Araujo, A.C., Etemad, A.: End-to-end prediction of parcel delivery time with deep learning for smart-city applications. IEEE Internet of Things Journal **8**(23), 17043–17056 (2021). https://doi.org/10.1109/JIOT.2021.3077007
2. Brody, S., Alon, U., Yahav, E.: How attentive are graph attention networks? arXiv preprint arXiv:2105.14491 (2021). https://doi.org/10.48550/arXiv.2105.14491

3. Bui, K.H.N., Cho, J., Yi, H.: Spatial-temporal graph neural network for traffic forecasting: An overview and open research issues. Applied Intelligence **52**(3), 2763–2774 (2022). https://doi.org/10.1007/s10489-021-02587-w
4. Fu, K., Meng, F., Ye, J., Wang, Z.: Compacteta: A fast inference system for travel time prediction. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 3337–3345 (2020). https://doi.org/10.1145/3394486.3403386
5. Gao, C., Zhang, F., Wu, G., Hu, Q., Ru, Q., Hao, J., He, R., Sun, Z.: A deep learning method for route and time prediction in food delivery service. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. pp. 2879–2889 (2021). https://doi.org/10.1145/3447548.3467068
6. Guo, H., Tang, R., Ye, Y., Li, Z., He, X.: Deepfm: a factorization-machine based neural network for ctr prediction. arXiv preprint arXiv:1703.04247 (2017). https://doi.org/10.48550/arXiv.1703.04247
7. Jindal, I., Chen, X., Nokleby, M., Ye, J., et al.: A unified neural network approach for estimating travel time and distance for a taxi trip. arXiv preprint arXiv:1710.04350 (2017). https://doi.org/10.48550/arXiv.1710.04350
8. Kendall, M.G.: A new measure of rank correlation. Biometrika **30**(1/2), 81–93 (1938). https://doi.org/10.1093/biomet/30.1-2.81
9. Lee, G., Yang, E., Hwang, S.: Asymmetric multi-task learning based on task relatedness and loss. In: International conference on machine learning. pp. 230–238. PMLR (2016)
10. Li, S., Jin, X., Xuan, Y., Zhou, X., Chen, W., Wang, Y.X., Yan, X.: Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. Advances in neural information processing systems **32** (2019)
11. Li, Y., Phillips, W.: Learning from route plan deviation in last-mile delivery (2018)
12. Lim, B., Arık, S.Ö., Loeff, N., Pfister, T.: Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting **37**(4), 1748–1764 (2021). https://doi.org/10.1016/j.ijforecast.2021.03.012
13. Malhotra, A., Vatsa, M., Singh, R.: Dropped scheduled task: Mitigating negative transfer in multi-task learning using dynamic task dropping. Transactions on Machine Learning Research (2023)
14. Nerbonne, J., Heeringa, W., Kleiweg, P.: Edit distance and dialect proximity. Time Warps, String Edits and Macromolecules: The theory and practice of sequence comparison **15** (1999)
15. Ott, F., Rügamer, D., Heublein, L., Bischl, B., Mutschler, C.: Joint classification and trajectory regression of online handwriting using a multi-task learning approach. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 266–276 (2022)
16. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
17. Wang, D., Zhang, J., Cao, W., Li, J., Zheng, Y.: When will you arrive? estimating travel time based on deep neural networks. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018). https://doi.org/10.1609/aaai.v32i1.11877
18. Wang, Z., Fu, K., Ye, J.: Learning to estimate the travel time. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 858–866 (2018). https://doi.org/10.1145/3219819.3219900
19. Wen, H., Lin, Y., Mao, X., Wu, F., Zhao, Y., Wang, H., Zheng, J., Wu, L., Hu, H., Wan, H.: Graph2route: A dynamic spatial-temporal graph neural network for pick-up and delivery route prediction. In: Proceedings of the 28th ACM

SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 4143–4152 (2022). https://doi.org/10.1145/3534678.3539084
20. Wen, H., Lin, Y., Wu, F., Wan, H., Guo, S., Wu, L., Song, C., Xu, Y.: Package pick-up route prediction via modeling couriers' spatial-temporal behaviors. In: 2021 IEEE 37th International Conference on Data Engineering (ICDE). pp. 2141–2146. IEEE (2021). https://doi.org/10.1109/ICDE51399.2021.00214
21. Wen, H., Lin, Y., Wu, F., Wan, H., Sun, Z., Cai, T., Liu, H., Guo, S., Zheng, J., Song, C., et al.: Enough waiting for the couriers: Learning to estimate package pick-up arrival time from couriers' spatial-temporal behaviors. ACM Transactions on Intelligent Systems and Technology **14**(3), 1–22 (2023). https://doi.org/10.1145/3582561
22. Wu, F., Wu, L.: Deepeta: A spatial-temporal sequential neural network model for estimating time of arrival in package delivery system. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 774–781 (2019). https://doi.org/10.1609/aaai.v33i01.3301774
23. Zhang, Y., Liu, Y., Li, G., Ding, Y., Chen, N., Zhang, H., He, T., Zhang, D.: Route prediction for instant delivery. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **3**(3), 1–25 (2019). https://doi.org/10.1145/3351282
24. Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., Sun, M.: Graph neural networks: A review of methods and applications. AI open **1**, 57–81 (2020). https://doi.org/10.1016/j.aiopen.2021.01.001
25. Zhu, L., Yu, W., Zhou, K., Wang, X., Feng, W., Wang, P., Chen, N., Lee, P.: Order fulfillment cycle time estimation for on-demand food delivery. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 2571–2580 (2020). https://doi.org/10.1145/3394486.3403307