Generating Practical Last-mile Delivery Routes using a Data-informed Insertion Heuristic

H. Rashidi[a,*], M. Nourinejad[b] and M. Roorda[a]

[a] University of Toronto, Toronto, Canada
hesam.rashidi@mail.utoronto.ca, matt.roorda@utoronto.ca
[b] York University, Toronto, Canada
mehdi.nourinejad@lassonde.yorku.ca
* Corresponding author

Extended abstract submitted for presentation at the Conference in Emerging Technologies in Transportation Systems (TRC-30)
September 02-03, 2024, Crete, Greece

April 28, 2024

Keywords: Last-mile delivery, Data-informed routing, Machine learning, Amazon research challenge

1 INTRODUCTION

Last-mile routing is a combinatorial optimization problem usually formulated as the Traveling Salesman Problem (TSP). The generic TSP finds the optimal route (e.g., route with minimal travel time) that a truck located at a warehouse can take to visit each consumer exactly once and then return to the warehouse. While routes generated by TSPs may be efficient on paper, empirical evidence suggests that they are often impractical, and drivers frequently *deviate* from them (Li & Phillips, 2018, Merchán *et al.*, 2022). Deviations occur as generic TSPs do not consider drivers' preferences and experiences (Özarık *et al.*, 2023). Drivers progressively gain experience from repeated deliveries in the same regions, which they use to create more practical routes than conventional TSPs (Ulmer *et al.*, 2020, Quirion-Blais & Chen, 2021). Experienced drivers, for instance, may possess knowledge of local traffic patterns, recipient availability, or convenient stop locations. This study develops a learning framework that extracts latent driver experiences from historical delivery routes and exploits them within a data-informed heuristic to generate more practical delivery routes.

We present a *human-centred* routing framework for TSPs with soft time windows. Human-centred algorithms leverage the workforce's tacit knowledge and accumulated daily experience to improve productivity and service quality. By valuing the workforce's expertise, these refined algorithms align more closely with the real world and aim to generate solutions that respect both the human element and operational efficiency. We develop a Data-informed Insertion Heuristic (DIIH), which incrementally creates a TSP tour based on a cost function inferred from past delivery routes taken by experienced drivers (Campbell & Savelsbergh, 2004).

We apply our framework to the open-source dataset from Amazon's Last-mile Research Challenge (Merchán *et al.*, 2022). In 2021, Amazon hosted the last-mile routing challenge, where the goal was to understand why generic TSPs generate solutions that differ from the routes executed by experienced Amazon delivery drivers and to ultimately reduce this gap by using data-driven methods. The dataset includes over 6,000 historically realized TSP instances in the United States. For each TSP instance, the actual sequence in which the driver visited the customers is documented and classified into one of three quality classes: high, medium, or low. The route quality labels indicate the satisfaction level of logistics planners at Amazon regarding a given observed route. These labels are based on a route's productivity, the driver's experience, and customer satisfaction levels. Amazon ranked the

competitors' routing algorithms based on a disparity score measuring how closely they matched the sequences in unseen high-quality labelled routes.

Our results indicate that the proposed framework effectively learns to generate high-quality route structures and neighbourhood visit times (i.e., routes with few backtracks, smoother turning angles, and times of visit with a high chance of recipient and parking availability) from historical routes. These learned insights by the framework are transferable to new fleet drivers to assist them in planning their routes and improve overall efficiency and successful delivery handoff. We show that the framework generates high-quality solutions for more problem instances compared to the benchmark. Furthermore, we propose an alternative method (i.e., an energy-based model) to Amazon's disparity score for determining the practical performance of a routing algorithm. We explain how the two metrics differ and provide arguments for why the alternative method may be better suited to assess an algorithm's performance.

2 METHODOLOGY & EXPERIMENTAL SETUP

2.1 OVERVIEW

Figure 1 shows the high-level framework developed by this study. The framework has two phases - offline and online. In the offline phase, the framework trains an ML classifier on the historical delivery data using supervised learning. The classifier predicts the probability of a tour's high or medium/low quality in the field. We use five-fold cross-validation to evaluate the classifier's generalization ability by iterative training and validating the model on different combinations of folds. Throughout the splitting process, stratified random sampling ensures that the proportions of different route quality classes are preserved in all subsets. We engineered two sets of features, differing in whether they change as the delivery sequence changes: instance-related (e.g., number of stops, packages, and day type) and route-related features (e.g., route duration and recipient availability likelihood). In the online phase, the framework exploits the insights from the classifier within the DIIH's cost function. The DIIH dynamically inserts randomly selected unvisited stops into a partial tour. We determine the best insertion position for an unvisited stop by using a cost function considering the added travel time, the increase in the sharpness of turning angles, backtracking, and the change in the quality of the visit times to the neighbourhoods present in the instance. Our framework uses the DIIH to create a pool of solutions (i.e., capped to a one-second runtime) and then selects the highest quality solution in the pool using the ML classifier. Note that the cost function and the classifier have a time complexity of $\mathcal{O}(n)$, where n is the number of stops within the TSP instance.

2.2 BASELINE

Machine Learning Classifier. This study trains and compares three classifiers: Logistic Regression, Random Forest, and Multi-layer Perception.

Data-informed Insertion Heuristic. The complete framework's performance is benchmarked to a solution with near-optimal total travel time and the Amazon competition's winning algorithm by Cook *et al.* (2022). We generate the near-optimal solution using `OR-tool`'s Guided Local Search (GLS) capped to a one-second runtime. This study was implemented using a MacBook Air with an eight-core Apple M2 chip and 16 GB of RAM.

2.3 PERFORMANCE EVALUATION METRICS

Machine Learning Classifier. The classifier's performance is evaluated based on standard binary classification metrics. These metrics include Accuracy, Precision, Recall, Negative Predictive Value (NPV), F1-score, and Area Under the receiver operating characteristic Curve (AUC).

Data-informed Insertion Heuristic. We use two metrics to evaluate the solutions generated by the DIIH: (1) The percentage of the solutions that the classifier predicts is of high quality. Note that

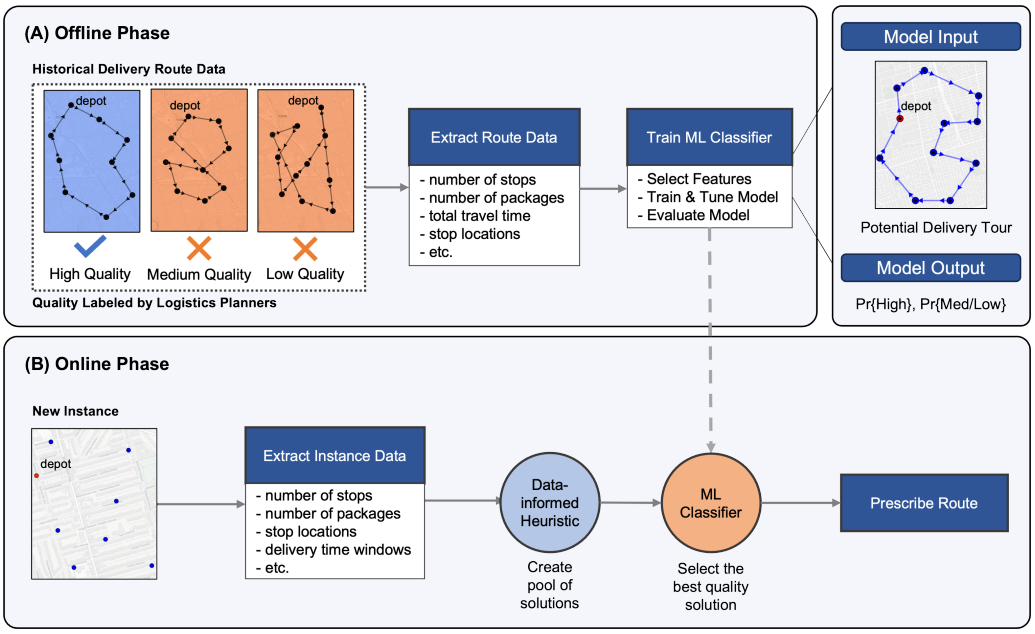

Figure 1 – *Visual abstract of the proposed framework.*

we correct this percentage by accounting for the predictive value of the classifier. (2) we use Amazon's disparity metric to evaluate how well the framework replicates the high-quality labelled routes taken by drivers.

3 RESULTS

Machine Learning Classifier. Table 1 compares the performance of the candidate classifiers. As can be seen, the models generalize well to unseen data since they have consistent performance during training and testing. We chose the Logistic Regression as the primary classifier in this study for the following reasons: (1) it has the best training performance across key metrics such as recall, F1-score, NPV, and the AUC, (2) it offers statistical information about the features' significance and role in predicting the target variable, and, (3) it is particularly well-suited for low-latency applications due to its simplicity and efficiency compared to more complex models.

Table 1 – *Classifier performance during cross-validation and testing. For each metric, the best-performing model during training is shown in bold.*

Model	Dataset	Accuracy	Precision	Recall	F1-score	NPV[a]	AUC[b]
Logistic Regression	CV Mean[c]	0.696	0.665	**0.640**	**0.652**	**0.720**	**0.755**
	Test set	0.698	0.678	0.609	0.642	0.711	0.748
Random Forest	CV Mean	0.685	**0.714**	0.489	0.580	0.673	0.749
	Test set	0.688	0.721	0.487	0.581	0.674	0.744
Multi-layer Perceptron	CV Mean	**0.698**	0.687	0.590	0.634	0.705	**0.755**
	Test set	0.699	0.701	0.562	0.624	0.698	0.747

[a] Negative predictive value, [b] Area under curve, [c] Cross-validation

Our results indicate that both instance-related and route-related features are statistically significant in predicting a route's quality. The instance-related features show that the structure of medium/low-quality routes is statistically different from that of instances with a courier-performed high-quality

route. These features are potentially acting as proxies for route productivity. For instance, while controlling for all variables, routes with more packages per stop are more productive and, thus, have higher quality. The route-related features indicate that high-quality routes have fewer backtracks and smoother turn angles. Additionally, they visit neighbourhoods at times more similar to those of historical high-quality labelled routes.

Data-informed Insertion Heuristic. The proposed framework outperforms the benchmark by generating a higher proportion of high-quality solutions. The classifier's quality assessment of the solutions reveals that our heuristic generates high-quality solutions for 92.0% of the instances. The classifier, while informative, is not infallible and possesses inherent limitations in its predictive power. To adjust for the classifier's predictive performance, we correct the proportions of expected high-quality versus medium/low-quality routes based on the classifier's precision score of 0.665 and an NPV of 0.720, as reported in Table 1. This correction implies that, on average, 66.5% of routes classified as high quality and $100-72=28\%$ of the routes predicted as medium/low quality are indeed high quality.

After the predictive value correction, it should be expected that 63.4% of our framework's solutions have a high quality, increasing the number of high-quality solutions by 18.9%, 19.8%, 12.3% compared to the courier-performed routes, GLS solutions, and the algorithm by Cook *et al.* (2022). Note that the predictive value correction assumes that the classifier's performance generalizes well to unseen data. This assumption is supported by the consistent results observed during cross-validation and testing phases (see Table 1). While our method improves route quality, it does so at the expense of increased travel time. The median travel time for the DIIH is 18.3%, 15.2%, and 12.7% higher than for the GLS, Cook *et al.* (2022), and courier-performed routes, respectively.

Based on Amazon's disparity metric the DIIH performs poorly compared to the benchmark. A disparity score closer to 0 indicates a closer match between two routes. Cook *et al.* (2022) scored an average of 0.019 dissimilarity on unseen data, earning the best score in the competition, whereas our method achieved an average of 0.122. However, we argue that the disparity metric does not fully represent an algorithm's performance and our trained classifier may be a better alternative for this purpose. (1) The disparity metric does not take into account the instance structure while evaluating the similarity of two routes. Our findings suggest that high-quality routes have a significantly different instance structure as compared to medium/low-quality routes. Thus, having a low disparity with high-quality solutions does not indicate how well an algorithm will perform on instances with a similar structure as historical routes with medium/low-quality labels. (2) The disparity metric does not classify TSP solutions; it reports a similarity score that does not specify how good of quality a route is. Our model, predicts a route's quality class based on the instance structure, visit times to neighbourhoods, and the route's overall structure.

References

Campbell, Ann Melissa, & Savelsbergh, Martin. 2004. Efficient Insertion Heuristics for Vehicle Routing and Scheduling Problems. *Transportation Science*, **38**.

Cook, William, Held, Stephan, & Helsgaun, Keld. 2022. Constrained Local Search for Last-Mile Routing. *Transportation Science*, 11.

Li, Yiyao, & Phillips, William. 2018. *Learning from Route Plan Deviation in Last-Mile Delivery*.

Merchán, Daniel, Arora, Jatin, Pachon, Julian, Konduri, Karthik, Winkenbach, Matthias, Parks, Steven, Noszek, Joseph, & Merchán, Merch´. 2022. Amazon Last Mile Routing Research Challenge: Data Set. *Transportation Science*.

Quirion-Blais, Olivier, & Chen, Lu. 2021. A Case-based Reasoning Approach to Solve the Vehicle Routing Problem with Time Windows and Drivers' Experience. *Omega (United Kingdom)*, **102**.

Ulmer, Marlin, Nowak, Maciek, Mattfeld, Dirk, & Kaminski, Bogumił. 2020. Binary Driver-Customer Familiarity in Service Routing. *European Journal of Operational Research*, **286**(10), 477–493.

Özarık, Sami Serkan, da Costa, Paulo, & Florio, Alexandre M. 2023. Machine Learning for Data-Driven Last-Mile Delivery Optimization. *Transportation Science*, **58**(10), 27–44.