Optimizing Automotive Logistics: Enhancing Efficiency in the Vehicle Transportation Services

BOUJIDA Nezar

11/04/2024

Abstract

This paper explores the design and application of a two-tower recommender system to improve the efficiency of a vehicle transportation service. The system aims to enhance the accuracy of matching drivers with vehicle transfer requests, reducing the reliance on manual processes.

Contents

1 Introduction

A leading provider of vehicle transportation services, specializes in moving single vehicles to various destinations. Operating with a network of over 7,000 independent drivers who, while not direct employees, play a crucial role in operations by transferring vehicles for a fee.

2 Problem Statement and Achieved Results

Customers depend on the service to identify suitable drivers for transporting their vehicles from point A to point B. The service facilitates this by listing transfer requests on their platform, where drivers can apply. However, when a transfer date approaches without any driver assignments, the process is handled manually.

Current system prioritizes high precision, recommending a particular group of drivers for each request.
Although this ensures accuracy in matches, it introduces several issues: the system's reliance on a narrowly defined pool of drivers limits its flexibility and the total number of drivers available, leading to increased operational costs and reducing the company's negotiation leverage due to the smaller selection of deemed suitable drivers.

In this paper, we introduce a recommender system designed to address a specific challenge: ranking drivers according to their suitability for a given transfer.

This system narrows the search space to approximately 100 drivers, roughly 2% of the original pool, ensuring that in 93% of cases, unsuitable drivers are ranked beyond the top 100.

For a subset of potential transfers, our existing system initially reached only 20% of drivers. However, with the recommendor system, we doubled our reach, engaging 40% of drivers.

In the existing system, a limited diversity leads to around 20% of drivers making 80% of transfers, granting them considerable choice and bargaining power. By enhancing diversity within the current framework, costs could be reduced by approximately 18%.
Conversely, the recommender system approach not only matches this baseline cost reduction but has the potential to achieve up to a 19% reduction in costs.
Figure 3 demonstrates the margin percentage gains possible with the recommender system, setting a new benchmark for optimizing the existing model.

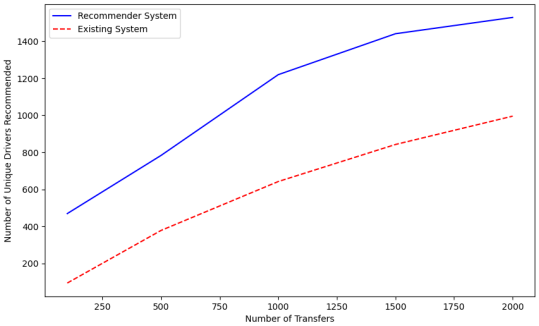

Figure 1: Driver's Diversity vs. Number of Transfers Given

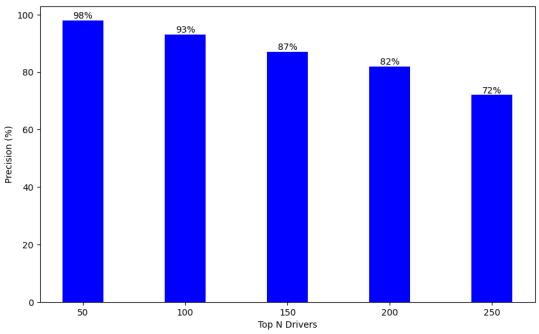

Figure 2: Precision at Different Recommendation Thresholds

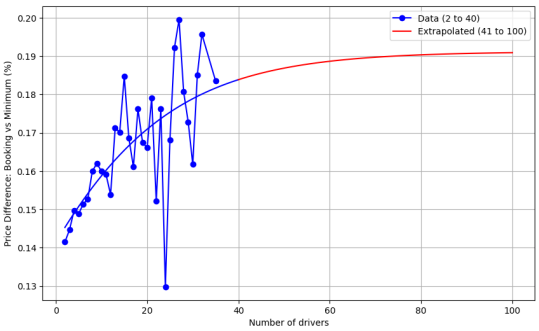

Figure 3: Price Variation Analysis: Booking vs. Minimum (%) by Number of Drivers

3 Approach

In response to the inefficiency in driver-request matching process, our solution is the implementation of a Two-Tower recommendation system.
This system is designed to automate and optimize the matching of drivers with vehicle transfer requests, reducing

the need for manual intervention and mitigating the necessity of last-minute price adjustments.
The system is composed of two key components:

Driver Tower: Processes attributes of each driver, it converts its data into a high-dimensional embedding, the system constructs profiles for drivers that encapsulate their preferences.
Request Tower: Processes attributes of each transfer request, such as pickup and drop-off locations. It transforms these details into an embedding that represents the unique characteristics of the request.

Upon receiving a new transfer request, the system generates its embedding and compares it against the embeddings of available drivers using cosine similarity metrics. This comparison gives a ranked list of drivers, ordered by their compatibility with the request, thereby facilitating an efficient and precise match.

In this model, $\mathbf{W}$, and $\mathbf{W}'$ Matrices denote the neural network weights for the Transfer and Driver Towers, respectively, while $\mathbf{B}$, and $\mathbf{B}'$ represent the corresponding biases. $\mathbf{D}$ captures the driver features, and $\mathbf{T}$ encapsulates the trip features .
Transfer Tower and Driver Tower operations are given by:

$$\mathbf{E} = \left.\begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mn} \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{pmatrix} + \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_m \end{pmatrix} \right\} \text{Transfer Tower}$$

$$\mathbf{E}' = \left.\begin{pmatrix} w'_{11} & w'_{12} & \cdots & w'_{1n} \\ w'_{21} & w'_{22} & \cdots & w'_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w'_{m1} & w'_{m2} & \cdots & w'_{mn} \end{pmatrix} \begin{pmatrix} D_1 \\ D_2 \\ \vdots \\ D_n \end{pmatrix} + \begin{pmatrix} B'_1 \\ B'_2 \\ \vdots \\ B'_m \end{pmatrix} \right\} \text{Driver Tower}$$

Given the vectors $\mathbf{E}$ and $\mathbf{E}'$ from the Transfer Tower and Driver Tower, respectively, the score can be represented as the cosine similarity between these two embeddings, denoted as $\text{cosine}(\mathbf{E}, \mathbf{E}')$. This is given by:

$$\text{score} = \text{cosine}(\mathbf{E}, \mathbf{E}') = \frac{\mathbf{E} \cdot \mathbf{E}'}{\|\mathbf{E}\| \|\mathbf{E}'\|}$$

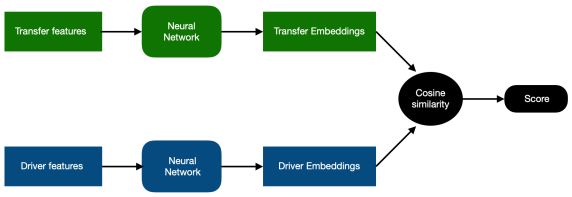

Figure 4: Diagram for two towers model.

4 Challenges/Solutions

4.1 Attendance Bias in Model Development

We encountered an issue with attendance bias in developing the two-tower model system, where it disproportionately favored drivers with more extensive data records. The bias's impact on the model was that these drivers were consistently recommended, overshadowing others who, in some cases, were a better fit.

4.1.1 Analyzing Dataset Discrepancy

The drivers dataset contained approximately 3423 drivers. We noticed that a group of 20% of the drivers were responsible for about 77% of the requests, among that group 23% of the drivers had an unusually high acceptance rate.
Approximately 2765 (80%) of the drivers, were responsible for 23% of the requests.
With the first versions of the recommender system, we noticed that this disparity of records affected the model, in a way that it focused mainly on the drivers with more records.

4.1.2 Resolving the Challenge

Records for 80% of the drivers were synthesized by pairing them with transfers. The match value was determined by evaluating the total round-trip distance, which represents the distance from the driver's location to the transfer departure and the return distance from the transfer conclusion to the driver's location. This cumulative distance is then mapped to a match value using the linear relationship $y = ax+b$, where x represents the round-trip distance and y the corresponding match value.
Through the application of QuantileBinning on the training data, we divide the round-trip distances into 20 equally-sized bins and compute the average match values for each. The coefficients a and b of the linear equation are estimated by correlating these average match values with their respective round-trip distances.

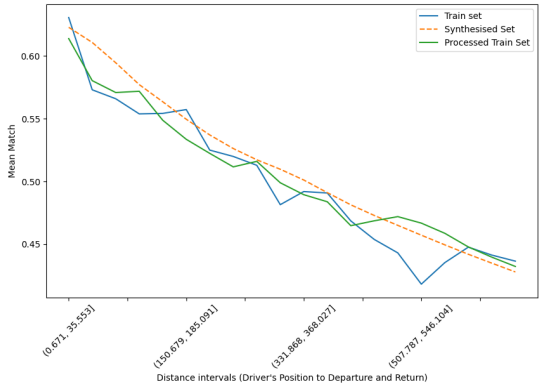

Figure 5: : Average Match Value by Distance intervals for Drivers

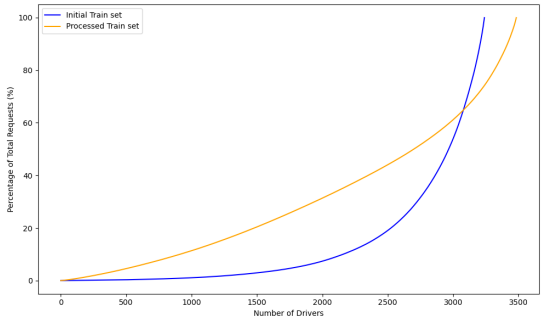

Figure 6: : Request Distribution: Initial vs. Processed Training Data

The figure 5 visually compares pre-sampling, post-sampling training sets, and the synthesized dataset, focusing on the average match value across intervals of a driver's round-trip distance.
This distance is divided into 20 quantiles, each representing an equal 5% of the total round-trip distance distribution.
The blue line represents the initial distribution, while a green line indicates the processed one. The close alignment of the two lines suggests the preservation of inherent data characteristics.

As For the 150 drivers with abnormally high request and acceptance rates, we generated synthetic negative records to neutralize their disproportionate influence.

The Figure 6 illustrates the request distribution among drivers in the training set, pre- and post-processing. The Initial Train set exhibits notable disparity; a small subset of drivers has a disproportionate share of total requests. Conversely, the Processed Train set depicts a more uniform distribution, indicating a reduction in disparity post-processing.

4.2 Selection Bias in the Model Development

We faced a selection bias as most transfer requests were for inter-region transfers.
This bias resulted in the recommender model favoring drivers from these specific regions, often suggesting them for transfers they were not suitable for, especially in the model's early versions.

4.2.1 Analyzing Dataset Discrepancy

In our dataset of transfer requests, we included a column for both the starting region (`region_start`) and the ending region (`region_end`). This setup allowed us to capture all possible combinations of regions, including cases where a transfer originates and concludes within the same region.
With 12 distinct regions, we theoretically could have 144 combinations ($12^2 = 144$). However, our dataset had only 143 combinations, indicating one missing pair.
We discovered that transfers between 10 specific pairs of regions accounted for 54% of all transfer requests. The remaining 133 pairs made up the other 46%, highlighting a significant imbalance in the distribution of transfer requests across different regional pairs.

4.2.2 Resolving the Challenge

We created a balanced training set by assigning weights to the region pairs (`region_all`), allocating higher weights to less frequent region pairs to ensure equitable representation during sampling.
This approach guarantees proportional contribution from each region to the training process of the model.

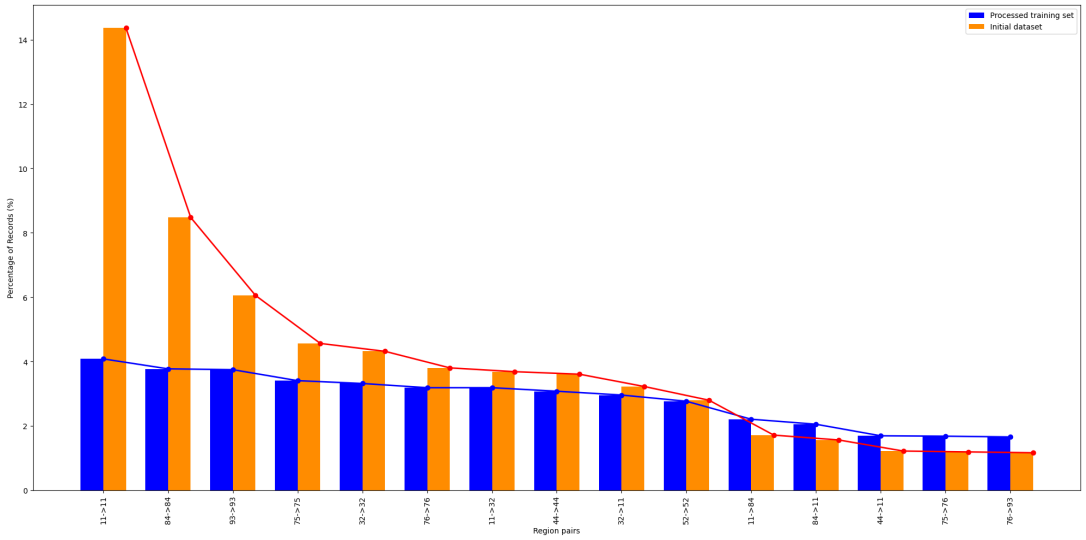

Figure 7: Top 15 Region pairs by Records Percentage: Initial Dataset vs. Processed Set

Figure 7 compares the initial dataset against the processed one, which includes a weighted training set. The initial dataset shows a sharper decline, with a few regional bars dominating a larger share of total requests. On the other hand, the weighted training set demonstrates a smoother decline in requests across regions, indicating a more uniform distribution.

5 Results:

5.1 AUC:

The ROC curve evaluates binary classifiers by plotting the True Positive Rate (TPR) versus the False Positive Rate (FPR) over various thresholds. TPR, defined as $\frac{TP}{TP+FN}$, reflects the model's accuracy in identifying actual positives, where TP are correct positive predictions and FN are positives missed by the model. FPR, calculated as $\frac{FP}{FP+TN}$, measures the rate at which actual negatives, TN, are incorrectly marked as positives, FP.
The AUC measures the area under the ROC curve. Direct integration is challenging due to the empirical nature of the ROC curve, leading to the use of the Trapezoidal Rule for approximation. The Trapezoidal Rule estimates the area under a curve by summing the areas of trapezoids formed between adjacent points on the ROC curve.

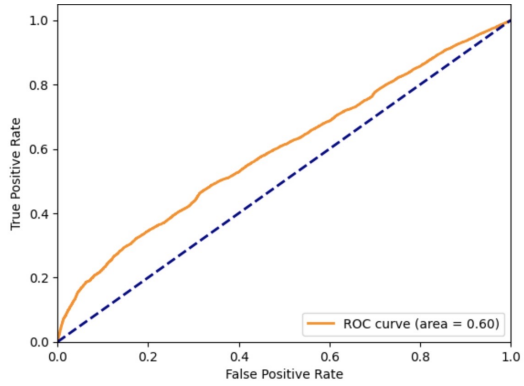

Figure 8: :Receiver Operating Characteristic (ROC)

The formula for the AUC using the Trapezoidal Rule is:

$$AUC = \sum_{i=1}^{n-1} \frac{(x_{i+1} - x_i) \cdot (y_{i+1} + y_i)}{2}$$

where x_i and x_{i+1} are consecutive FPR values, and y_i and y_{i+1} are the corresponding TPR values.

The ROC curve area of 0.60 indicates that the model performs better than random guessing.

An AUC of 0.60 suggests that suitable drivers are more frequently ranked within the top 100, while unsuitable drivers tend to be ranked lower.

5.2 Diversification:

To quantitatively measure the diversification achieved by our refined recommender system model, we introduce a new metric, termed *Driver Recommendation Diversification (DRD)*. Mathematically, the DRD metric is defined as follows:
Let n be the total number of unique transfers considered. For each transfer i, the recommender system generates a list of the top 100 recommended drivers, denoted as R_i. We define the set D as the union of all driver sets R_i across all transfers, i.e.,

$$D = \bigcup_{i=1}^{n} R_i$$

Initially, the system's diversification efforts inadvertently centered around 20% of drivers to achieve coverage across all regions. An early version of the model, biased, targeted merely 9% of the drivers. Subsequent refinements led to an unbiased model, significantly expanding coverage to include 40% of drivers.
This enhancement gives greater diversification while diminishing dependency on a select group of drivers. It aligns with equitable regional representation, providing a wider selection of drivers and facilitating better pricing negotiations.

5.3 No Match Rank:

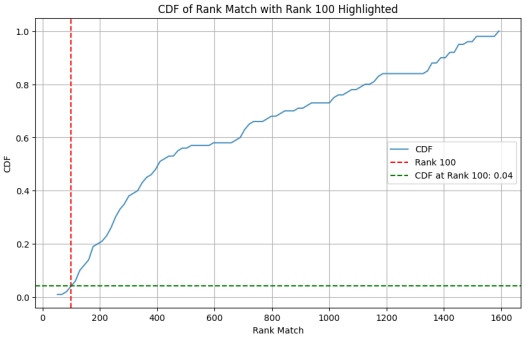

Figure 9: CDF of Rank Match with Rank 100 Highlighted

Figure 9 presents the cumulative distribution plot for the rankings of drivers whose requests were rejected in the testing set, using the recommender system to determine their ranking positions.
The model's proficiency in identifying match relationships is clear, with 96% of cases showing drivers ranked beyond the 100th position. This underscores the model's ability to effectively recognize and categorize mismatches.

6 Appendix

The logistic growth model, used to describe processes with initial rapid growth decelerating near a limiting threshold, is mathematically represented as:

$$\frac{L}{1 + e^{-k(x-x_0)}}$$

Here, L denotes the curve's maximum value, k the growth rate indicating the steepness of initial growth, and x_0 the midpoint where growth shifts from acceleration to deceleration. This model offers a lens for analyzing growth patterns constrained by specific limits.

In our case, the logistic growth model illuminates a pivotal trend: as a transfer garners numerous requests from suitable drivers, the remuneration amount tends to decrease, thereby augmenting the margin per transfer. This phenomenon is attributed to the surge of requests, enhancing the likelihood of identifying the most fitting driver. For instance, a driver already intending to travel from the transfer's departure to its destination perceives this as a mutually beneficial opportunity, often willing to accept lower fees.

In our dataset, each transfer attracts multiple drivers, but the selected driver's bid isn't always the lowest. We identify the lowest bid (*lowest_asked*) and compare it with the actual paid amount (*price_paid*). The profit margin is determined as $\frac{price\_paid - lowest\_asked}{price\_paid}$, highlighting the financial efficiency of the selection process. Through the logistic growth model, we investigate how the abundance of drivers influences finding a suitable one at minimal cost. This analysis sheds light on optimizing driver selection for cost efficiency while ensuring service delivery.

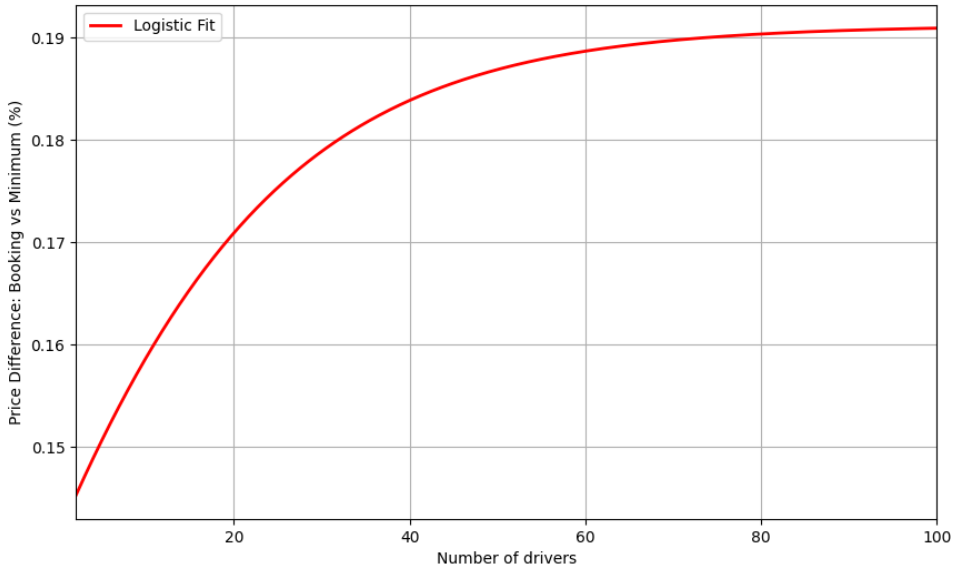

Figure 10: Logistic model fit to data

6.1 Results

Using the existing system with an enhanced driver pool, we can achieve a 16% increase in margin gains solely through increased driver diversity. Implementing the recommender system sets this 16% as a baseline, with projections indicating potential growth up to 19%. This logistic growth trend suggests that strategic driver expansion, coupled with the recommender system, can substantially optimize profit margins.

7 Bibliography

- Innovative Recommendation Applications Using Two Tower Embeddings at Uber. (2023). Retrieved from https://www.uber.com/en-FR/blog/innovative-recommendation-applications-using-two-tower-embeddings/
- Understanding the Two-Tower Model in Personalized Recommendation Systems. (2021). Retrieved from https://hackernoon.com/understanding-the-two-tower-model-in-personalized-recommendation-systems
- Bias Assessment Approaches for Addressing User-Centered Fairness in GNN-Based Recommender Systems. (2022). Retrieved from https://www.mdpi.com/2078-2489/14/2/131
- Logistic growth curve modeling of US energy production and consumption. (2021). Retrieved from https://www.sciencedirect.com/science/article/abs/pii/S1364032118305586