VeLP: Vehicle Loading Plan Learning from Human Behavior in Nationwide Logistics System

Sijing Duan
Central South University
duansijing@csu.edu.cn

Feng Lyu
Central South University
fenglyu@csu.edu.cn

Xin Zhu
JD Logistics
zhuxin8@jd.com

Yi Ding
Massachusetts Institute of Technology
yiding01@mit.edu

Haotian Wang
JD Logistics
wanghaotian18@jd.com

Desheng Zhang
Rutgers University
desheng@cs.rutgers.edu

Xue Liu
McGill University
xueliu@cs.mcgill.ca

Yaoxue Zhang
Tsinghua University
zhangyx@tsinghua.edu.cn

Ju Ren
Tsinghua University
renju@tsinghua.edu.cn

ABSTRACT

For a nationwide logistics transportation system, it is critical to make the vehicle loading plans (i.e., given many packages, deciding vehicle types and numbers) at each sorting and distribution center. This task is currently completed by dispatchers at each center in many logistics companies and consumes a lot of workloads for dispatchers. Existing works formulate such an issue as a cargo loading problem and solve it by combinatorial optimization methods. However, it cannot work in some real-world nationwide applications due to the lack of accurate cargo volume information and effective model design under complicated impact factors as well as temporal correlation. In this paper, we explore a new opportunity to utilize large-scale route and human behavior data (i.e., dispatchers' decision process on planning vehicles) to generate vehicle loading plans (i.e., plans). Specifically, we collect a five-month nationwide operational dataset from JD Logistics in China and comprehensively analyze human behaviors. Based on the data-driven analytics insights, we design a Vehicle Loading Plan learning model, named VeLP, which consists of a pattern mining module and a deep temporal cross neural network, to learn the human behaviors on regular and irregular routes, respectively. Extensive experiments demonstrate the superiority of VeLP, which achieves performance improvement by 35.8% and 50% for trunk and branch routes compared with baselines, respectively. Besides, we deployed VeLP in JDL and applied it in about 400 routes, reducing the time by approximately 20% in creating plans. It saves significant human workload and improves operational efficiency for the logistics company.

PVLDB Reference Format:
Sijing Duan, Feng Lyu, Xin Zhu, Yi Ding, Haotian Wang, Desheng Zhang, Xue Liu, Yaoxue Zhang, and Ju Ren. VeLP: Vehicle Loading Plan Learning from Human Behavior in Nationwide Logistics System. PVLDB, 17(2): 241 - 249, 2023.
doi:10.14778/3626292.3626305

Corresponding author: Feng Lyu

Proceedings of the VLDB Endowment, Vol. 17, No. 2 ISSN 2150-8097.
doi:10.14778/3626292.3626305

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at .

1 INTRODUCTION

With the full-range penetration of online shopping, efficient and fast package delivery like Amazon in the U.S. and JD Logistics (JDL) in China has become crucial to user service experience [10, 27]. By June 2022, JDL built a nationwide logistics network [23], with over 5,000 inter-region and intra-region routes connecting over 160 distribution centers and over 210 sorting centers across roughly 500 Chinese cities. This network facilitates the daily delivery of over 20 million packages. Central to these centers is the frequent task of advance vehicle loading planning, crucial for on-time delivery.

The vehicle loading planning problem involves selecting the types and quantity of vehicles for each transportation route given a certain number of packages. Vehicles of varying capacities, such as vans ($14m^3$-$35m^3$), lorries ($40m^3$-$78m^3$), and trailers ($94m^3$-$120m^3$), can be utilized to meet different loading needs. Previous research models this problem as a pallet stacking or knapsack problem with capacity constraints, employing combinatorial optimization methods or machine learning [2, 4, 7, 8, 13, 14, 20, 22, 34, 36, 42, 44]. However, there are several limitations in the actual industry scenarios. **(1) The volume information of packages is inaccurate.** The reason is that the packages are not accurately measured at courier business stations or stocked from warehouses due to non-standard operations. Although there are some methods to measure the size of the packages accurately (e.g., cameras [17]), the cost is very high for nationwide deployment. **(2)** Even with accurate volume information for the individual packages, **the uneven placement of packages with irregular shapes, soft bags, and deformable items** makes it hard to calculate the accurate loading plan with classic combinatorial optimization solutions. To summarize, the existing solutions cannot effectively address our problem.

In this study, rather than focusing on individual package sizes, we derive loading plans by examining route data and predicting the *volume bound* of packages. Volume bound represents the space a group of packages occupies in a vehicle with efficient placement by on-site workers, calculated from vehicle loading plans (i.e., plans).

The operational nationwide vehicle loading planning behavior data in JDL gives us a new opportunity to achieve this by studying and learning from the dispatchers' decision-making process. However, it's not straightforward because (1) **dispatchers' decisions depend on various complex factors**, including route types, time, and waybill quantity; (2) **their decisions for a specific route are temporally correlated** due to the temporal correlation of cargo volume (e.g., previously unloaded cargo will be loaded in future batches). Hence, inferring volume bound data from dispatchers' decisions is a challenging task.

To tackle these challenges, we take the lead in studying plan learning from human behavior data. Specifically, we conduct a thorough analysis of human planning behaviors and introduce `VeLP`, a model learning from historical human behaviors. `VeLP` comprises two components: a Pattern Mining (PM) module for regular routes and a Deep Temporal Cross Network (DTCNet) for irregular routes. The PM identifies routes based on historical loading plan data and reuses historical plans for regular routes. The DTCNet is tailored for irregular routes, predicting volume bounds by integrating an LSTM-based temporal network and a factorization machines-based deep feature cross network. This captures sequential correlations and cross-feature information. The outputs guide volume bound prediction and plans generated through a cost-aware mapping strategy. The main contributions of this paper are summarized as follows:

- To our knowledge, we are the first to utilize large-scale route and vehicle loading decision data to predict volume bound and generate loading plans by learning human behaviors without leveraging the actual cargo's volume information. This attempt is valuable for the logistics industry to improve dispatchers' working efficiency and save human workload.
- Technically, we design a `VeLP` model with the PM module and DTCNet component for learning human decision behaviors on regular and irregular routes, respectively, addressing the unique challenges in the logistics vehicle planning scenario. It includes inaccurate cargo volume, complicated factors influencing human decisions, and uncertain temporal correlation.
- We implement and deploy `VeLP` in a large-scale logistics system of JDL, applied in about 400 routes with an adoption rate of 80%, which reduces the time by approximately 20% in creating plans. Extensive offline experiments demonstrate the efficacy of `VeLP` in plan volume bound prediction, adoption rate, and coverage rate compared to state-of-the-art baselines. Particularly, `VeLP` can achieve the overall 74.81% and 81.34% adoption rate for respective trunk and branch routes, improving the performance by 35.8% and 50% with baselines.

2 BACKGROUND AND PRILIMILARIES

2.1 System Background and Motivation

Figure 1 overviews the architecture of the logistics transportation system, which supports the delivery of over 20 million packages daily nationwide. The system comprises various nodes (warehouses, sorting centers (SCs), distribution centers (DCs), and courier business stations) and routes (ferry routes, branch routes, trunk routes, transfer routes, and urban delivery routes). Warehouses store cargo, SCs are located in urban areas, larger SCs (DCs) are based in regional capitals, and courier stations directly service users. Ferry

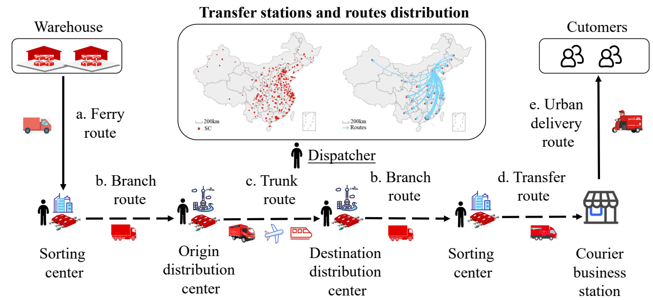


Figure 1: Nationwide logistics delivery process.

routes transport cargo from warehouses to originating DCs. Branch routes are medium-distance trips from SCs to neighboring originating DCs. Trunk routes are long-distance inter-regional trips from an originating DC to a destination DC. Transfer routes deliver packages from SCs to courier stations. Urban delivery routes transport from courier stations to customers. Each type of route and nodes depend on the business type and departure/destination locations.

We focus on branch and trunk routes between sorting and distribution centers in this paper. Figure 1 shows a geographical map representing the nodes' locations and their route links. Red circles represent nodes, and blue lines signify trunk routes (limited to those departing from Beijing for visibility). The extensive transportation infrastructure roughly includes over 6,000 routes across 500 cities.

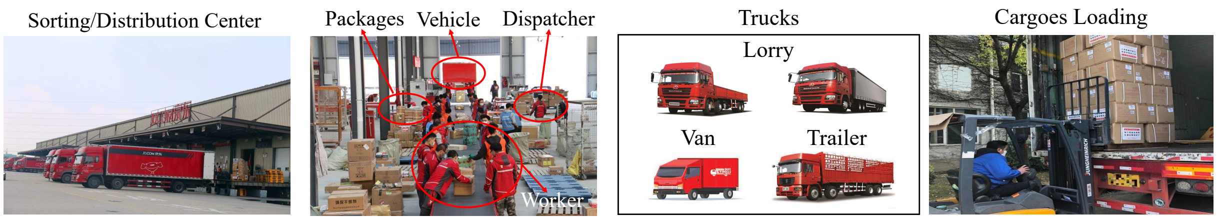
Figure 2(a) displays a transfer station snapshot where dispatchers plan vehicle loads for each departure round, then sort packages by destination and load them onto vehicles. Figure 2(b) shows one example snapshot of different kinds of trucks that can be selected. Note that, in the JDL, there are over 30 different types of vehicles with capacities ranging from $10m^3$ to $120m^3$. The last picture shows the cargo loading process, where cargo is packed in bags/packages and stacked in the compartment of each vehicle. When finishing loading, the dispatcher will use a laser ruler to estimate the cargo volume of vehicles. Each delivery vehicle corresponds to one transportation job, assigned to a specific route for the delivery task.

Motivation. In the logistics system, one of the most crucial and frequent actions is to make plans for each transport task in advance. To reduce the workload and improve the working efficiency of dispatchers, who spend 30% of working time on manual planning every day and have other responsibilities other than vehicle dispatching (e.g., on-site inspection and personnel arrangement). Our motivation is to generate plans by learning human behaviors without considering the cargo's volume information.

2.2 Preliminaries

DEFINITION 1 (**VEHICLE LOADING PLAN <ABBR: A PLAN OR PLANS>**). *We define a tuple $\beta_n(e) = (r^n(e), p_i^n(e), a_i^n(e))$ to represent a plan of each route $e \in \mathbb{U}$ on the n-th day, where r, p_i, a_i, and $\mathbb{U}$ denote carrier name, vehicle type, the number of vehicles of i-th type, and the set of routes, respectively. A plan can include one or multiple vehicle types, depending on the total volume of packages.*

DEFINITION 2 (**ENTROPY OF THE PLAN**). *We define an entropy metric, $E(C_e)$, to measure the plan stability of each route $e \in \mathbb{U}$, calculated by $E(C_e) = \sum_{e\in\mathbb{U}} p(C_e) \log \frac{1}{p(C_e)}$, where C_e is the plan*



(a) Snapshot of transfer station.

(b) Snapshot of trucks and cargoes loading.

Figure 2: Snapshots in an operational system.

of route e, $\mathbb{U}$ is the set of routes, and $p(\cdot)$ is the probability function. A lower $E(C_e)$ indicates a more stable plan pattern.

Definition 3 (**Volume Bound**). *Volume bound is the amount of space occupied by a set of packages within a vehicle, accounting for efficient placement determined by on-site workers. The volume bound can be calculated based on the plans.*

Definition 4 (**Human Behaviors**). *Human behaviors represent dispatchers' decisions when making route plans, influenced by on-site observations and historical experience.*

Definition 5 (**Problem Formulation**). *We define plan learning as a predict-then-decide task. Given a set of the route features $\mathcal{R}^e$ and historical human behaviors $\mathcal{D}^e$ on vehicle loading planning of route e, we firstly predict the volume bound of the target route, then generate the final vehicle plan. The problem can be formally defined as: $f_\theta : (\mathcal{R}^e, \mathcal{D}^e) \rightarrow y$, where f_θ is the model parameterized by θ that we aim to learn, y is the vehicle plan.*

For better understanding, we use the regular route as an example. Given $\mathcal{R}^e$ including origin and destination nodes of route e, business type, departure time, and carrier name. As well as $\mathcal{D}^e$, including vehicle plans on different numbers of waybills for e over the past t days. Our objective is to determine the vehicle plans y on the $t + 1$ day, specified as a pair <number, vehicle type> (e.g., <1, 9.6-meters lorry>). More route patterns can be found in Section 3.2.2.

3 DATA-DRIVEN ANALYSIS AND INSIGHTS

3.1 Data Collection

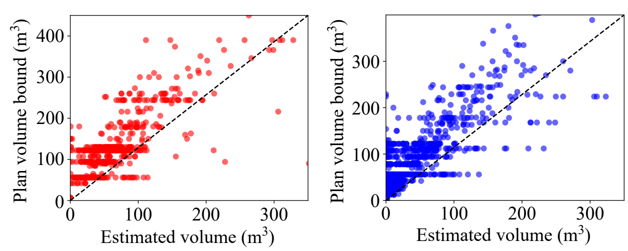
To understand vehicle loading planning behaviors and develop a data-driven model, we gathered a five-month dataset from nationwide logistics transportation. The dataset, as shown in Table 1, includes transportation job records with information such as job code, creation time, business type, departure/destination city, carrier name, estimated volume/weight, driver name/plate number, and the plan. Notably, JDL manages around 80,000 self-owned and 400,000 third-party vehicles from various cooperative carriers, encompassing over 30 vehicle types, each with a volume bound (maximum loading capacity) ranging from $10m^3$ to $120m^3$.

3.2 Human Planning Behaviors

3.2.1 Inexact Plan Decisions. Figure 3 shows the estimated volume (measured by dispatchers with a laser ruler before the vehicle departure) vs. the plan volume bound for trunk and branch routes. Different plans yield multiple volume bounds despite identical cargo volumes, generally exceeding estimated volumes. This reveals dispatchers' inexactitude, leaving room for improvement. To explore the potential reasons, we interviewed 20 dispatchers and learned that the plan decisions account for multiple factors, including historical data, resource availability, route, and temporal features. Moreover, as the plans are made before departure, the dispatchers have to predict the future cargo volume based on the current package counts, further biasing the plan decisions.

Table 1: Transportation job records.

Trans. Job Code	**Create Time**	**Business Type**
TJ21071***74800	2021-07-19 01:00:00	Client Delivery
Departure Node	**Destination Node**	**Departure Time**
Beijing SC	Shanghai SC	2021-07-19 03:00:00
Estimated Arrival Time	**Carrier Name**	**Estimated Volume**
2021-07-19 10:00:00	JLink Company	100 m^3
Estimated Weight	**Package Count**	**Driver Name**
120 Kg	450	Liwei
Plate Number	**Plan Type**	**Vehicle Number**
LAG8**3	17.5-meters Trailer	1
Data Statistics		
Number of DC: 38; Number of SC: 400; Number of records: 0.56 million Number of routes: 2,250; Time span: from 07/15/2021 to 12/09/2021		

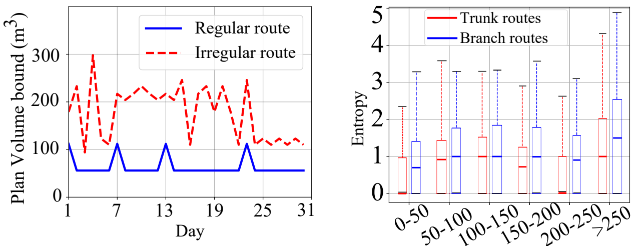


(a) Trunk routes.

(b) Branch routes.

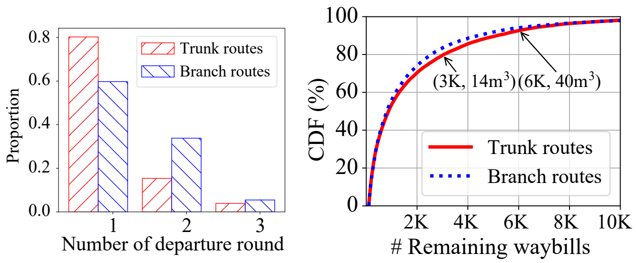
Figure 3: Inexact plan decisions.

3.2.2 Patterns of Regular and Irregular Plans. We investigate two planning patterns: regular and irregular, by randomly selecting representative routes and analyzing their volume bounds in July 2021. As shown in Figure 4(a), regular routes maintain a consistent volume bound, primarily at $96m^3$ throughout the month. In contrast, irregular routes exhibit significant volume bound fluctuations, ranging from $94m^3$ to $298m^3$. To assess plan stability, we utilize $E(C_e)$ as defined in *Definition 3* and categorize volume bounds into six levels, from 0-50 m^3 (Level 1) to > 250 m^3 (Level 6). Figure 4(b) provides box plots showing the entropy distribution of trunk and branch routes, yielding several insights: (i) Branch routes exhibit higher entropy due to cargo volume fluctuations. (ii) Lower entropy is observed for Level 1 and Level 5, indicating more stable loading plans for extremely small or large cargo volumes. (iii) All 25%



(a) Example of regular and irregular routes. (b) Stability of plans.

Figure 4: Patterns of Regular and Irregular Plans.



(a) Departure rounds. (b) CDFs of remaining waybills.

Figure 5: Temporal Patterns.

percentiles reach 0, indicating significant plan stability in certain routes.

3.2.3 Temporal Patterns. We analyze correlations between adjacent rounds for routes with multiple daily departure rounds. Figure 5(a) illustrates route distribution based on the number of departure rounds. While more than 75% of routes have a single departure round, over 20% have multiple rounds. Field studies reveal a strong correlation between plans of adjacent rounds, with previous round's remaining volume impacting the next. During non-holidays, about 25.4% of tasks have remaining waybills, increasing to over 30% during holidays. Figure 5(b) shows cumulative distribution functions (CDFs) of previous round's remaining waybills for trunk and branch routes, indicating that about 50% of tasks in both route types have over 1,000 remaining waybills. According to dispatchers, when waybill numbers reach 3,000 and 6,000, an additional vehicle with volume bounds of 14 m^3 and 40 m^3 is needed. Plans are made 8-12 hours before departure, with a 3-4 hour interval between rounds, making it challenging to establish a direct link between remaining waybills and the next round's plan.

4 DESIGN OF VELP

4.1 System Overview

To solve the vehicle loading planning problem, we propose a predict-then-decide framework VeLP, which is shown in Figure 6. It is composed of four stages and two key components: Pattern Mining (PM) and Deep Temporal Cross Network (DTCNet). PM mines historical patterns to generate vehicle loading plans for regular routes. DTCNet learns historical behaviors for irregular routes using features from experienced dispatchers. The system first preprocesses data from the transportation management system. With historical task records and contextual information (e.g., route, carrier, and departure time), PM identifies regular and irregular routes and matches rules from historical plans for regular routes. These plans are then

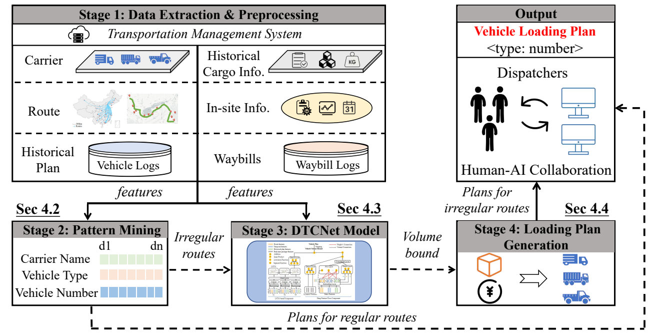


Figure 6: System framework of VeLP.

applied to future contexts. For irregular routes, DTCNet uses additional features to predict volume bound. Based on the volume bound prediction, the cost-aware plan mapping strategy can output the final appropriate vehicle loading plan. Next, we elaborate on each component in detail.

4.2 Pattern Mining Module

Data analysis in Section 3.2 shows that some routes' plans are regular. Thus, the PM module is designed to identify regular and irregular routes. Specifically, plans with similarity scores equal to or greater than θ over c days ($c \in C$) are marked as regular routes, where C is the cycle value set. The threshold θ can be adjusted based on business requirements, and the rules are dynamically updated with the adjustment[1]. We adopt the Jaccard similarity to measure the plan similarity for route e between the n-th and m-th day as $P^e_{simi}(n, m) = \frac{\beta_n(e) \cap \beta_m(e)}{\beta_n(e) \cup \beta_m(e)}$.

Algorithm 1 details the PM module, which utilizes a set C (i.e., $\{1, 7\}$) of cycle values to assess the periodicity of loading plans. For instance, a cycle value of 1 signifies that the plan for a specific route e is identical on consecutive days, while a value of 7 implies the same plan repeats weekly. By categorizing routes based on their cycle values, the PM module extracts regular plans and reuses them in the future context, thus enhancing vehicle loading efficiency.

4.3 Deep Temporal Cross Network

In order to identify the critical factors influencing dispatchers' behaviors on irregular routes, we have undertaken a thorough feature extraction process. This is based on data-driven insights, field studies online interviews and expert knowledge, with some results reported in Section 3. We identify four categories of static and dynamic features (route, temporal, historical plan, and real-time waybill features) that strongly relate to dispatchers' decisions, as detailed in Table 2.

Prior to model input, we preprocess both continuous and discrete features. Specifically, we calculate the detailed embedding v_r for route features (1). For business type (2) and discrete temporal features (3-6), we use one-hot encoding to vectorize them. For historical features (7-9) and package-related continuous features (10), we design a dense layer to extract latent representations.

[1]Based on expert experiences and verified in the experiments, we set θ=0.8 in this paper.

Algorithm 1: Pattern Mining Algorithm

Input: Route set $\mathcal{R}$, plan set $\beta(\mathcal{R})$, cycle set C, similarity threshold θ, day n and m.

Output: Regular route set $\mathcal{R}_s$, irregular route set $\mathcal{R}_q$, regular plan set $\beta(\mathcal{R}_s)$, cycle c.

$\mathcal{R}_s = \emptyset$, $\mathcal{R}_q = \emptyset$, $\beta(\mathcal{R}) = \emptyset$;
for $e \in \mathcal{R}$ **do**
  for $c \in C$ **do**
    Group e based on c ;
    Compute $P^e_{simi}(n, m)$ within groups on n-th and m-th day ;
  Compute the average value $avg(e)$ of all $P^e_{simi}(n, m)$ for route e with cycle c ;
  if $avg(e) \geq \theta$ **then**
    Add e and $\beta(e)$ to $\mathcal{R}_s$ and $\beta(\mathcal{R}_s)$;
  $\mathcal{R}_q = \mathcal{R}$ - $\mathcal{R}_s$;
return $\mathcal{R}_s$, $\beta(\mathcal{R}_s)$, and $\mathcal{R}_q$;

Table 2: The features for DTCNet.

Features	Static
(1) Origin and destination nodes of route e	Y
(2) Business type of route e	Y
(3) Departure time from origin node of route e	Y
(4) The n-th departure round of one day	Y
(5) Day of the week	Y
(6) Month of year	Y
(7) Last departure round plan of route e	Y
(8) Yesterday's plan of route e	Y
(9) Volume bound, volume, weight in last 14 days of route e	Y
(10) The number of waybills in the last 20 hours of route e	N

After identifying and preprocessing the related features, we then design DTCNet[2] to predict vehicle loading volume bound, as shown in Figure 7. This design is based on two considerations: (i) We utilize a Long Short Term Memory (LSTM)-based model [21] to capture temporal correlations among sequential departure rounds within a day for each route. (ii) To allow our model to learn more detailed representations, rather than relying solely on numerical values, we use the DeepFM-based model [12] to encode each category of features, taking into account both low- and high-order feature interactions. We avoid using complex modules like the Transformer to learn temporal correlations, as the sequence length in our time series is relatively short. In our scenario, LSTM is sufficient to learn sequential features among adjacent departure rounds.

4.3.1 The LSTM Component. At departure round T_i, the LSTM cell takes the vector representation of the round T_{i-1}, the memory state and hidden state at round $i-1$ as inputs, i.e., $h_i, c_i = LSTM(v_{T_{i-1}}, h_{i-1}, c_{i-1})$, where h_i and h_{i-1} denote the hidden states at departure rounds T_i and T_{i-1}, and c_i and c_{i-1} represent the corresponding memory states. The vector representation $v_{T_{i-1}}$ of departure round T_{i-1} combines route, temporal, historical plan, and

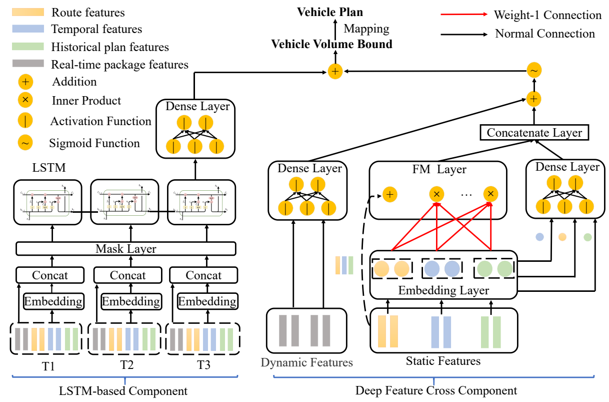


Figure 7: Illustration of DTCNet.

dynamic waybill features. It's worth noting that some routes only have 1 or 2 departure rounds. We use a masking layer to standardize the sequence data so that each route has an identical feature length [3]. To prevent over-fitting, we incorporate a dropout mechanism on LSTM states between consecutive steps. The output of the LSTM cell is then inputted into a dense layer made up of a fully connected network (FCN). Therefore, the output of the LSTM-based component can be represented as $LSTM_{out} = FCN(h_i)$.

4.3.2 The Deep Feature Cross (DFC) Component. To effectively capture the intricacies of human behaviors from the available feature set, we introduce the *DFC* component, which is capable of learning both lower-order and higher-order features. The *DFC* is composed of four primary layers: the embedding layer, dense layer, factorization-machine (FM) layer, and concatenate layer, all receiving the same input. To ensure uniformity in the embedding vectors for varying-sized static features, we compute them. These embeddings are denoted as $e_i = w_i \cdot x_i$, where e_i is the embedding of the i-th feature, w_i denotes the parameters within the embedding layer, and x_i represents the vector of the i-th raw input feature.

Then, e_i is fed into the FM layer to model order-2 feature interactions. The FM layer is designed to learn interaction features, which is applicable for plan prediction with complex feature combinations. Particularly, FM layer models feature pair interactions as the inner products of respective feature latent vectors. Compared to the order-1 linear layer, it can capture order-2 feature interactions much more efficiently, especially for sparse data. Due to this flexible representation learning capability, the FM layer can learn the feature interactions that never or rarely appeared in the training data. The output of the FM layer is the summation of an *Addition* function and several *Inner Product* functions:

$$y_{FM} = < w, x > + \sum_{i=1}^{d} \sum_{j=i+1}^{d} < v_i, v_j > e_i \cdot e_j, \quad (1)$$

where v_i and v_j are latent vectors. The Addition unit $< w, x >$ reflects the importance of order-1 features. The *Inner Product* $e_i \cdot e_j$ represents the impact of order-2 feature interactions.

Afterward, the outputs are fed into the dense layer to model high-order feature interactions. The dense layer is a feed-forward neural network with several hidden layers, which is used to learn high-order feature interaction representations. After going through the dense layer, the output feature vector is denoted as

[2] *Normal Connection* in black refers to the connection with weight to be learned; the *Weight-1 Connection* in red arrow is a connection with weight 1 by default; *Addition* means adding all input together; *Inner Product* means the output is the product of two input vector; *Activation Function* are used for non-linearly transforming; *Sigmoid Function* is used as the output function; Each category of features are colored differently.

$$y_{dense}(x) = W^{|\lambda|+1} \cdot a^{|\lambda|} + b^{|\lambda|+1}, \quad (2)$$

where $|\lambda|$ is the number of hidden layers, W, a, and b are the network parameters.

Finally, the output of the FM layer and dense layer are combined in a concatenate layer. Since the related dynamic amount of waybills is a continuous feature, they are directly fed into dense layers to extract high-order representations. Finally, all features are added and activated with *Sigmoid* function, i.e.,

$$DFC_{out} = Sigmoid(y_{FM}(x) + y_{dense}(x)), \quad (3)$$

where $y_{FM}(x)$ and $y_{dense}(x)$ are the outputs of FM and dense layer.

With the outputs of two components, we can derive the final output $DTCNet_{out}$ (i.e., volume bound) with *Addition* operation, which can be denoted as $DTCNet_{out} = Addition(DFC_{out} + lstm_{out})$. During model training, the mean absolute error loss is calculated by $\mathcal{L} = \sum_{e \in \mathcal{R}} |v_{real} - DTCNet_{out}|$, where v_{real} is the real volume bound calculated by the vehicle loading plan for route $e \in \mathcal{R}$.

4.4 Cost-Aware Plan Generation

To create appropriate plans, we finally match the predicted volume bound to a specific plan from the recent month. We do this in two steps. First, we find plans from the past month with volume bounds that are closest to the predicted volume. However, our analysis shows that one volume bound can have multiple corresponding plans. So, in the second step, we select the plan with the lowest transportation cost. We can denote the final plan as $\mathcal{F}(DTCNet_{out})$, where $\mathcal{F}$ is the cost-aware plan mapping function.

5 PERFORMANCE EVALUATION

5.1 Evaluation Methodology

Experimental Settings. We conduct experiments based on five-month operation data to evaluate the performance. All data samples are divided into five groups equally, among which four groups are used for training while the remaining one is used for testing. In addition, the testing group is alternately adopted to calculate the error bars, i.e., conducting the k-fold cross-validation with $k = 5$. We implement the proposed model and other baselines with Keras, Python 3.6 environment. The experiments are carried out on a CPU server with 8 cores and 20 GB of memory, we set batch size and learning rate as 256 and 0.01, respectively. The Adam optimizer is used for model training.

Baselines. To demonstrate the superiority of our proposed VeLP, we design and implement the following baselines. We choose these three methods because they are commonly used in the literature [24, 38, 41] in human behavior modeling and prediction. The features for all baselines and our proposed method are the same. Note that although these algorithms have been widely adopted in the existing literature, there is no reference implementation for vehicle loading plan generation in logistics transportation systems.

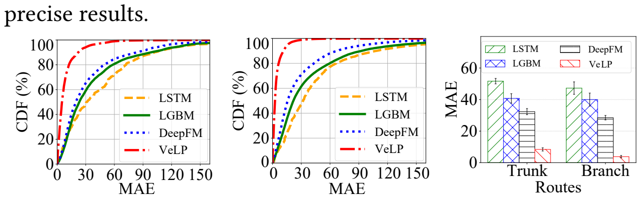
- **LGBM** [18]: is a gradient boosting method that uses tree-based learning algorithms.
- **LSTM** [15]: is a variant of recurrent neural networks (RNN) and widely adopted in capturing the long- and short-term temporal correlations among sequential data.
- **DeepFM** [12]: is a general deep learning framework, which combines the factorization machines and deep neural network for feature learning to conduct item recommendation.

Performance Metrics. Three metrics are adopted to quantify the model performance, including one technical and two business metrics. Denote p^i_{alg} and p^i_{hum} by the i-th plan (i.e., the set of planned vehicles) generated by an algorithm and human decision for a route.

- **Mean Absolute Error (MAE: technical metric)**: refers to the averaged differences between the prediction and ground-truth values, i.e., $\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$, where y_i and $\hat{y}_i$ are the actual value and predicted value of the i-th prediction, and n is the total number of predictions.
- **Adoption rate (business metric)**: refers to the proportion of the same number of vehicles between algorithm-based and human actual plans, divided by the total number of vehicles in algorithm-based plans, i.e., $\frac{\sum_{i=1}^{n}|p^i_{alg} \cap p^i_{hum}|}{\sum_{i=1}^{n}|p^i_{alg}|}$. This metric is utilized to evaluate dispatchers' adoption willingness. Because plans usually require human final approval since some in-site information cannot be obtained in the system in advance. It is also an important problem in other human-AI collaboration systems [19, 31, 35, 43].
- **Coverage rate (business metric)**: refers to the proportion of the same number of vehicles between algorithm-based and human actual plans, divided by the total number of vehicles in human actual plans, i.e., $\frac{\sum_{i=1}^{n}|p^i_{alg} \cap p^i_{hum}|}{\sum_{i=1}^{n}|p^i_{hum}|}$.

5.2 Performance of Volume Bound Prediction

We first examine the performance of plan volume bound prediction. Figure 8(a) and Figure 8(b) display CDFs of MAE scores for trunk and branch routes. VeLP outperforms other models significantly. Figure 8(c) shows average MAEs with error bars, revealing VeLP's smaller MAE and reduced deviation, indicating reliability. VeLP outperforms in capturing critical factors and temporal correlations in the decision-making process, resulting in more accurate predictions. Traditional methods may ignore these nuances, yielding less precise results.

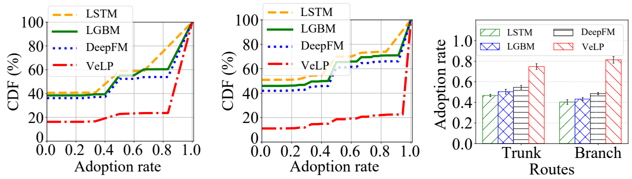


(a) CDFs of MAE on trunk. (b) CDFs of MAE on branch. (c) Average MAE.

Figure 8: Vehicle plan volume bound prediction.

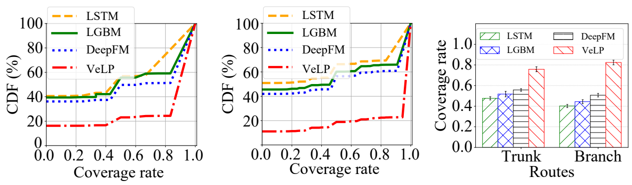
5.3 Performance of Loading Plan Learning

Figure 9(a) and Figure 9(b) display CDFs of adoption rates for trunk and branch routes, and Figure 9(c) shows average performance with error bars. VeLP outperforms other baselines significantly, achieving a higher average adoption rate. This demonstrates VeLP's effectiveness in learning human behaviors through multi-dimensional features. Figure 10(a) and Figure 10(b) exhibit CDFs of coverage rates for trunk and branch routes, with Figure 10(c) illustrating average performance with error bars. VeLP surpasses other baselines significantly in terms of coverage rate, attributed to the separation of regular and irregular routes and the specialized PM module and DTCNet. This approach enables our model to learn from dispatcher behavior patterns, while other baselines may overlook these characteristics, resulting in suboptimal solutions.



(a) CDFs of adoption rate on trunk routes. (b) CDFs of adoption rate on branch routes. (c) Average adoption rate performance.

Figure 9: Performance of overall adoption rate.



(a) CDFs of coverage rate on trunk. (b) CDFs of coverage rate on branch. (c) Average coverage rate performance.

Figure 10: Performance of overall coverage rate.

5.4 Ablation Study

We first conduct ablation studies to verify key feature effectiveness in the design of VeLP. Table 3 shows the results, highlighting the influence of features, particularly yesterday's departure round plan and waybill count in the last 20 hours. We also evaluate component design in VeLP through ablation experiments, as shown in Table 4. We can observe that removing any component has a notable impact on overall performance, underlining the critical role of all components in vehicle loading plan learning.

Table 3: Feature ablation of trunk / branch routes.

Ablation	MAE	Adoption rate
all features	8.39 / 3.81	74.81% / 81.34%
- Yesterday plan	15.33 / 13.41	64.13% / 67.44%
- Waybill feature	10.17 / 7.59	71.21% / 71.87%
- History features	10.03 / 7.01	72.56% / 73.11%
- Day of week	9.95 / 6.84	73.62% / 75.52%

Table 4: Component ablation of trunk / branch routes.

Ablation	MAE	Adoption rate
VeLP	8.39 / 3.81	74.81% / 81.34%
w/o PM	12.85 / 8.52	70.64% / 76.83%
w/o DTCNet	52.34 / 48.50	41.37% / 47.21%

We further conduct two ablation studies to demonstrate the effectiveness of PM and DTCNet components on regular and irregular routes. Specifically, Table 5 presents the adoption and coverage rates of the PM module on regular trunk and branch routes. Likewise, Table 6 provides the results for irregular routes. We observe that PM and DTCNet outperform other baselines on trunk and branch routes. In addition, it shows a smaller performance gap between PM and DTCNet compared to other baselines, mainly because these two modules are tailored for regular and irregular routes.

Table 5: Results for regular routes (Trunk / Branch).

	Adoption rate	Coverage rate
PM	79.41% / 80.33%	81.29% / 80.58%
DeepFM	67.65% / 64.27%	69.68% / 65.51%
LGBM	65.37% / 62.05%	68.23% / 63.94%
LSTM	60.44% / 58.11%	60.35% / 59.23%

Table 6: Results for irregular routes (Trunk / Branch).

	Adoption rate	Coverage rate
DTCNet	72.21% / 79.46%	73.15% / 80.01%
DeepFM	48.29% / 40.21%	50.04% / 40.58%
LGBM	40.56% / 34.12%	41.22% / 35.77%
LSTM	38.23% / 30.21%	36.16% / 31.84%

6 REAL-WORLD DEPLOYMENT

6.1 Deployment Implementation

We deploy VeLP in JDL, one of the largest logistics companies in China, for real-time plan generation to boost dispatcher efficiency. This involves selecting around 400 routes in South and North China as pilot areas and training the model using offline data. We collect usage data from the transportation management system and utilize a data dashboard to monitor online performance. The dashboard in Figure 11 provides detailed information for each route, such as the date, origin and destination nodes, route code, algorithm-generated plans, and actual dispatcher-used plans.

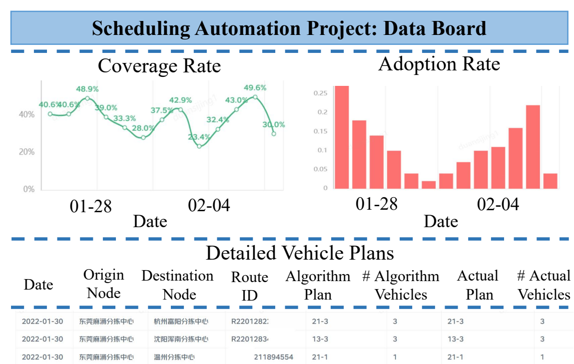


Figure 11: Visualization data dashboard.

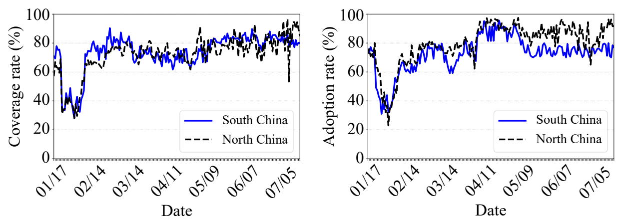
6.2 Deployment Performance

We analyze six months of deployment data in South and North China, spanning from 01/17/2022 to 07/17/2022. Figure 12(a) displays coverage rates in both areas. Scores start around 70% in the first week, dipping during the Chinese Spring Festival, with a significant drop to about 30% around 02/01/2022. Figure 12(b) reveals adoption rates, with North China slightly outperforming South China. Both rates decline markedly as the Chinese Spring Festival approached, reaching about 45% and 42% for North and South China from 01/23/2022 to 02/07/2022. After festival, coverage and adoption rates return to normal levels, approximately 75%.

The reason for the decreased results during the period of the Chinese Spring Festival is that the dispatchers have unique loading plans to handle the increased number of cargoes due to festival promotion. The corresponding behavior data is not included in the training data set (we only have training data from 07/2021 to

12/2021). The coverage and adoption rates are still not 100% in the rest of the time. We have an in-depth discussion on why correctly predicted vehicle plans are not adopted in Section 6.3.

We also analyze the working efficiency improvement. Before implementing VeLP, dispatchers spent 7.8 minutes per task manually creating plans, handling an average of 24 tasks daily and consuming 3.12 hours, 39% of their workload. With VeLP applied to 30% of nationwide routes, auto-generated plans now require only modifications and confirmation from dispatchers, saving up to 35 minutes per day. Furthermore, VeLP can reduce time spent on creating temporal loading plans by around 20%, as it learns from historical route-specific human behaviors, thereby minimizing errors in volume estimation.



(a) Online coverage rate. (b) Online adoption rate.

Figure 12: Temporal Patterns.

6.3 Lessons and Discussions

6.3.1 Human-Data-Driven System Collaboration. While machine learning approaches have been applied to order dispatching and route planning in logistics and delivery[6, 25, 26], the reliance on human decision-making at certain points persists due to factors like data limitations and policies. Therefore, the collaboration between humans and data-driven systems is poised to be a significant area of research in the near future. To facilitate this collaboration effectively, it's crucial to comprehend human decision-making behaviors and the underlying motivations guiding their acceptance or rejection of decisions from data-driven systems.

Our nationwide logistics experiment and a survey of over 300 dispatchers identified two main causes of failure in human-data-driven decision-making collaboration, particularly in plan rejection by dispatchers. Firstly, inconsistencies in human-generated historical data lead to algorithmic failures due to hidden factors influencing human decisions or inherent decision randomness. Secondly, real-time onsite situations unknown to the algorithm, such as resource shortages or capacity limitations, disrupt plan execution. This highlights the need for future collaborations to address human-reported exceptions and develop strategies for their learning.

While the VeLP model might seem less intricate when compared with certain complex models, it is essential to highlight that the true significance of VeLP lies in its effective solution of a significant but often neglected real-world problem within the domain of human-data-driven system collaboration in the industrial logistics. Both offline evaluations and online deployment prove the performance effectiveness. Therefore, our approach can make a meaningful contribution to the field.

6.3.2 Limitations and Enhancements. We identify several limitations of VeLP: (i) It may not fully grasp sudden increases in cargo volume during holiday promotions (e.g., the Chinese New Year Festival). (ii) Relying on historical dispatcher behaviors may yield suboptimal results since not all dispatcher decisions are optimal. (iii) The platform's data lacks comprehensiveness, such as actual cargo volumes, which impacts plan prediction performance. To improve VeLP, we propose collecting more accurate volume data, establishing a plan feedback mechanism, i.e., dispatchers are required to fill in the detailed reasons when they modify or reject the AI-generated plans, and incorporating holiday patterns and common temporary conditions into the algorithm.

7 RELATED WORKS

Data-Driven Delivery/Logistics System. Many studies have proposed to enhance operational efficiency in data-driven delivery and logistics systems, mostly targeting vehicle routing optimization problems [5, 26, 32, 33, 37, 39, 40], and food delivery [6, 9, 11, 16, 28–30]. For example, the authors of [32, 33, 39] proposed route planning methods for shared mobility and ridesharing services, which differs from our scenario in logistics vehicle planning. Pan et al. [28] proposed a UAVs schedule approach optimizing time and package delivery quantity. However, these time and route optimization methods do not apply to our task, which centers on learning vehicle loading plans, not optimizing distance or time.

Cargo Loading Problem. The existing literature typically formulates cargo loading problems as knapsack or pallet stacking issues with capacity constraints, resolved by either combinatorial optimization or machine learning methods [1, 2, 4, 7, 13, 14, 20, 22, 34, 36, 42, 44]. For example, Zhu *et al.* [44] introduced a data-driven tree search algorithm for the 3-dimensional bin packing problem. However, most of them assume accurate advanced knowledge of cargo volume. In reality, accurately calculating package volume is challenging due to the irregular shapes, soft bags, and deformable items that packages can comprise. Therefore, traditional combinatorial optimization solutions are ineffective in our context.

8 CONCLUSION

In this paper, we have studied a vehicle load planning problem based on one operational dataset collected in the nationwide logistics transportation system. With data-driven insights, we have designed and implemented VeLP, which consists of the PM module and DTCNet model. Finally, we have developed VeLP in the nationwide logistic transportation system. Both offline experiments and online deployment performance have demonstrated the efficacy of VeLP. For future works, we will further enhance the performance of VeLP based on the user feedback from dispatchers.

ACKNOWLEDGMENTS

This work was supported by the National K&D Program of China under Grant 2022YFC2009805 and 2022YFF0604504; in part by the National Natural Science Foundation of China under Grant No. 62320106006, 62002389, 62122095, 62341201, 62072472, U19A2067 and 92067206; in part by the 111 Project under Grant B18059; in part by the Central South University Innovation-Driven Research Program under Grant 2023CXQD029; in part by the Key Research and Development Program of Hunan Province of China under Grant 2022GK2013; and by a grant from the Guoqiang Institute, Tsinghua University.

REFERENCES

[1] Sara Ali, António Galrão Ramos, Maria Antónia Carravilla, and José Fernando Oliveira. 2022. On-line three-dimensional packing problems: A review of off-line and on-line solution approaches. *Computers & Industrial Engineering* 168 (2022), 108122.

[2] Batin Latif Aylak, Murat İnce, Okan Oral, Gürsel Süer, Najat Almasarwah, Manjeet Singh, and Bashir Salah. 2021. Application of machine learning methods for pallet loading problem. *Applied Sciences* 11, 18 (2021), 8304.

[3] Hanjie Chen and Yangfeng Ji. 2020. Learning variational word masks to improve the interpretability of neural text classifiers. *arXiv preprint arXiv:2010.00667* (2020).

[4] Michael Cho. 2019. The knapsack problem and its applications to the cargo loading problem. *Analysis of Applied Mathematics* 13 (2019), 48–63.

[5] Dyutimoy Nirupam Das, Rohan Sewani, Junwei Wang, and Manoj Kumar Tiwari. 2020. Synchronized truck and drone routing in package delivery logistics. *IEEE Transactions on Intelligent Transportation Systems* 22, 9 (2020), 5772–5782.

[6] Xuetao Ding, Runfeng Zhang, Zhen Mao, Ke Xing, Fangxiao Du, Xingyu Liu, Guoxing Wei, Feifan Yin, Renqing He, and Zhizhao Sun. 2020. Delivery Scope: A new way of restaurant retrieval for on-demand food delivery service. In *ACM SIGKDD*. 3026–3034.

[7] Jiahui Duan, Xialiang Tong, Fei Ni, Zhenan He, Lei Chen, and Mingxuan Yuan. 2022. A data-driven column generation algorithm for bin packing problem in manufacturing industry. *arXiv preprint arXiv:2202.12466* (2022).

[8] Sijing Duan, Dan Wang, Ju Ren, Feng Lyu, Ye Zhang, Huaqing Wu, and Xuemin Shen. 2022. Distributed artificial intelligence empowered by end-edge-cloud computing: A survey. *IEEE Communications Surveys & Tutorials* (2022).

[9] Chengliang Gao, Fan Zhang, Yue Zhou, Ronggen Feng, Qiang Ru, Kaigui Bian, Renqing He, and Zhizhao Sun. 2022. Applying deep learning based probabilistic forecasting to food preparation time for on-demand delivery service. In *ACM SIGKDD*. 2924–2934.

[10] İlayda Gezer, Hasancan Erduran, Alper Kayıhan, Burak Çetiner, and Pervin Ersoy. 2022. Logistics service quality of online shopping websites during covid-19 pandemic. In *Digitizing Production Systems*. Springer, 725–734.

[11] Baoshen Guo, Shuai Wang, Yi Ding, Guang Wang, Suining He, Desheng Zhang, and Tian He. 2021. Concurrent order dispatch for instant delivery with time-constrained actor-critic reinforcement learning. In *IEEE RTSS*. 176–187.

[12] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247* (2017).

[13] Yakin Hajlaoui, Amel Jaoua, and Safa Bhar Layeb. 2022. Deep reinforcement learning for solving the single container loading problem. *Engineering Optimization* (2022), 1–17.

[14] Xinyun He, Yuan Zhang, Zhu Le, and Yanping Du. 2022. Research on" one-to-many" vehicle and cargo matching optimization problem based on improved genetic algorithm. In *ACM ICIIT*. 7–15.

[15] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[16] Zhiqing Hong, Guang Wang, Wenjun Lyu, Baoshen Guo, Yi Ding, Haotian Wang, Shuai Wang, Yunhuai Liu, and Desheng Zhang. 2022. CoMiner: nationwide behavior-driven unsupervised spatial coordinate mining from uncertain delivery events. In *ACM SIGSPATIAL*. 1–10.

[17] iNEWS. 2023. How to measure the size/volume of express parcel boxes? https://inf.news/en/tech/6d9cb93e0df1a0a882842de880b5bf0e.html.

[18] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems* 30 (2017).

[19] Vivian Lai, Chacha Chen, Q Vera Liao, Alison Smith-Renner, and Chenhao Tan. 2021. Towards a science of human-AI decision making: a survey of empirical studies. *arXiv preprint arXiv:2112.11471* (2021).

[20] Xijun Li, Mingxuan Yuan, Di Chen, Jianguo Yao, and Jia Zeng. 2018. A data-driven three-layer algorithm for split delivery vehicle routing problem with 3D container loading constraint. In *ACM SIGKDD*. 528–536.

[21] Zachary C Lipton, John Berkowitz, and Charles Elkan. 2015. A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019* (2015).

[22] Jiaye Liu, Jiali Mao, Jiajun Liao, Huiqi Hu, Ye Guo, and Aoying Zhou. 2020. SCLPD: Smart cargo loading plan decision framework. In *IEEE ICDE*. 1758–1761.

[23] JD Logistics. 2022. 2022 INTERIM REPORT. https://ir.jdl.com/en/ir_report.php.

[24] Fuyuan Lyu, Xing Tang, Huifeng Guo, Ruiming Tang, Xiuqiang He, Rui Zhang, and Xue Liu. 2022. Memorize, Factorize, or be Naive: Learning Optimal Feature Interaction Methods for CTR Prediction. In *IEEE ICDE*. IEEE, 1450–1462.

[25] Wenjun Lyu, Haotian Wang, Zhiqing Hong, Guang Wang, Yu Yang, Yunhuai Liu, and Desheng Zhang. 2023. REDE: Exploring Relay Transportation for Efficient Last-mile Delivery. In *IEEE ICDE*. IEEE, 3003–3016.

[26] Yi Ma, Xiaotian Hao, Jianye Hao, Jiawen Lu, Xing Liu, Tong Xialiang, Mingxuan Yuan, Zhigang Li, Jie Tang, and Zhaopeng Meng. 2021. A hierarchical reinforcement learning based optimization framework for large-scale dynamic pickup and delivery problems. *Advances in Neural Information Processing Systems* 34 (2021), 23609–23620.

[27] Daniel Merchán, Jatin Arora, Julian Pachon, Karthik Konduri, Matthias Winkenbach, Steven Parks, and Joseph Noszek. 2022. 2021 Amazon last mile routing research challenge: Data set. *Transportation Science* (2022).

[28] Yan Pan, Shining Li, Qianwu Chen, Nan Zhang, Tao Cheng, Zhigang Li, Bin Guo, Qingye Han, and Ting Zhu. 2021. Efficient schedule of energy-constrained UAV using crowdsourced buses in last-mile parcel delivery. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1 (2021), 1–23.

[29] Sijie Ruan, Cheng Long, Xiaodu Yang, Tianfu He, Ruiyuan Li, Jie Bao, Yiheng Chen, Shengnan Wu, Jiangtao Cui, and Yu Zheng. 2022. Discovering actual delivery locations from mis-annotated couriers' trajectories. In *IEEE ICDE*. 3241–3253.

[30] Zeyuan Shang, Emanuel Zgraggen, Benedetto Buratti, Philipp Eichmann, Navid Karimeddiny, Charlie Meyer, Wesley Runnels, and Tim Kraska. 2021. Davos: a system for interactive data-driven decision making. *PVLDB* 14, 12 (2021), 2893–2905.

[31] Sulayman K Sowe, Eric Simmon, Koji Zettsu, Frederic De Vaulx, and Irena Bojanova. 2016. Cyber-physical-human systems: Putting people in the loop. *IT professional* 18, 1 (2016), 10–13.

[32] Yongxin Tong, Yuxiang Zeng, Zimu Zhou, Lei Chen, and Ke Xu. 2022. Unified route planning for shared mobility: An insertion-based framework. *ACM Transactions on Database Systems* 47, 1 (2022), 1–48.

[33] Yongxin Tong, Yuxiang Zeng, Zimu Zhou, Lei Chen, Jieping Ye, and Ke Xu. 2018. A unified approach to route planning for shared mobility. *PVLDB* 11, 11 (2018), 1633.

[34] Giulia Tresca, Graziana Cavone, Raffaele Carli, Antonio Cerviotti, and Mariagrazia Dotoli. 2022. Automating bin packing: A layer building matheuristics for cost effective logistics. *IEEE Transactions on Automation Science and Engineering* (2022).

[35] Dakuo Wang, Justin D Weisz, Michael Muller, Parikshit Ram, Werner Geyer, Casey Dugan, Yla Tausczik, Horst Samulowitz, and Alexander Gray. 2019. Human-AI collaboration in data science: exploring data scientists' perceptions of automated AI. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (2019), 1–24.

[36] Zhenkun Wang, Hui-Ling Zhen, Jingda Deng, Qingfu Zhang, Xijun Li, Mingxuan Yuan, and Jia Zeng. 2021. Multiobjective optimization-aided decision-making system for large-scale manufacturing planning. *IEEE Transactions on Cybernetics* 52, 8 (2021), 8326–8339.

[37] Haomin Wen, Youfang Lin, Xiaowei Mao, Fan Wu, Yiji Zhao, Haochen Wang, Jianbin Zheng, Lixia Wu, Haoyuan Hu, and Huaiyu Wan. 2022. Graph2Route: A dynamic spatial-temporal graph neural network for pick-up and delivery route prediction. In *ACM SIGKDD*. 4143–4152.

[38] Canran Xu and Ming Wu. 2020. Learning feature interactions with lorentzian factorization machine. In *AAAI*, Vol. 34. 6470–6477.

[39] Yi Xu, Yongxin Tong, Yexuan Shi, Qian Tao, Ke Xu, and Wei Li. 2020. An efficient insertion operator in dynamic ridesharing services. *IEEE Transactions on Knowledge and Data Engineering* 34, 8 (2020), 3583–3596.

[40] Yuxiang Zeng, Yongxin Tong, and Lei Chen. 2019. Last-mile delivery made practical: An efficient route planning framework with theoretical guarantees. *PVLDB* 13, 3 (2019), 320–333.

[41] Feng Zhang, Yani Liu, Ningxuan Feng, Cheng Yang, Jidong Zhai, Shuhao Zhang, Bingsheng He, Jiazao Lin, Xiao Zhang, and Xiaoyong Du. 2021. Periodic weather-aware LSTM with event mechanism for parking behavior prediction. *IEEE Transactions on Knowledge and Data Engineering* 34, 12 (2021), 5896–5909.

[42] Yan Zhang, Yanmei Zhou, and Hao Hu. 2021. Research on optimization of knapsack problem in logistics distribution. In *Advanced Multimedia and Ubiquitous Engineering*. Springer, 1–9.

[43] Ji Zhou, Yanhong Zhou, Baicun Wang, and Jiyuan Zang. 2019. Human–cyber–physical systems (HCPSs) in the context of new-generation intelligent manufacturing. *Engineering* 5, 4 (2019), 624–636.

[44] Qianwen Zhu, Xihan Li, Zihan Zhang, Zhixing Luo, Xialiang Tong, Mingxuan Yuan, and Jia Zeng. 2021. Learning to pack: A data-driven tree search algorithm for large-scale 3D bin packing problem. In *CIKM*. 4393–4402.