Proceedings of the 2020 Winter Simulation Conference
K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, eds.

ENHANCING INPUT PARAMETER ESTIMATION BY MACHINE LEARNING FOR THE SIMULATION OF LARGE-SCALE LOGISTICS NETWORKS

Yang Liu
Liang Yan
Sheng Liu
Ting Jiang
Feng Zhang
Yu Wang
Shengnan Wu

JD Logistics
59 Kechuang 11th Street
Daxing District
Beijing, 100176, CHINA

ABSTRACT

The quality of large-scale logistics network simulation highly depends on the estimation of its key input parameters, which are usually influenced by various factors that are difficult to obtain. To tackle this challenge, this paper proposes a framework to estimate these parameters with high precision through machine learning, in which the impacting factors are divided into static and dynamic groups and used as features to train a learning model for estimation. To overcome the obstacle that dynamic factors are hard to obtain in some scenarios, the proposed framework employs unsupervised learning to analyze their patterns and extract time-invariant features for modeling. A validation study is conducted on the estimation of distribution center sorting times. The results proved our approach can generate more accurate estimation of input parameters, even with the shift of operational plans and absence of relevant data.

1 INTRODUCTION

The operation of logistics networks is crucial to corporations and decision makers intend to improve its performance for various purposes, such as reducing the cost of logistics activities, improving the on-time rate and customer satisfaction (Ghiani and Musmanno 2013; Rabe et al. 2018b; Rabe et al. 2018a). However, the maintenance of a large-scale logistics network is usually non-trivial due to the high complexity (Rabe et al. 2018b). This induces the unpredictable nature of the operational actions such that a locally optimal action may have counter effects on the global performance of the logistics network (Rushton and Baker 2006). To resolve this issue, simulation is often employed as a powerful decision support tool to evaluate the impacts of actions on the logistics network's performance (Rabe et al. 2018b; Rabe et al. 2018a). It imitates the real-world in reaction to the actions instead of on-site evaluation, which could be expensive, difficult or dangerous (Poropudas et al. 2011).

As the largest retailer in China, JD.com is running China's largest logistics network that comprises thousands of nodes and transportation routes concatenating them, where a node can be a warehouse, a regional distribution center or a terminal station. To operate this large-scale logistics network, resource allocation is crucial to guarantee the fast, reliable and in-time delivery of goods. The real logistics network is influenced by multiple dynamic, random and complicated factors that cannot be appropriately captured by a mathematical formulation. Hence, the conventional optimization techniques for logistics network

978-1-7281-9499-8/20/$31.00 ©2020 IEEE

planning usually need many assumptions which may not be plausible. In contrast, simulation models can incorporate enough details of the logistics network, and hence the decision makers can achieve a more concrete evaluation of the strategies. Thus, we have built a simulation model that mimics our logistics network, which depicts the whole shipping process, where an item purchased online is packaged and shipped from a warehouse and forwarded by distribution center(s) to the station closest to the customer for final delivery. As a customer may place multiple orders simultaneously, the items purchased at the same time are usually packaged and delivered together and referred to as a "shipping bill". Given a generated plan, the simulation model emulates the processing state of each shipping bill, thereby gaining a global view of the logistics network for multiple purposes such as time efficiency inspection, system bottleneck identification and resource allocation rationality checking.

While simulation techniques demonstrates the advantages on analyzing the complex systems, it relies on the input data models to capture the characteristics of the real world (Barton 2012). In fact, input parameter modeling is a long existing domain in simulation research since an appropriate input data model can effectively complement the simulation system and improve the accuracy (Rabe and Scheidler 2014). A key challenge on the construction of input model is that it is approximated from the finite samples in the real world. This induces uncertainties and errors of estimation, which is subsequently propagated to the simulation output (Barton 2012; Zhou and Xie 2015). Based on our observation from the simulation of JD logistics network, the quality of input data modeling significantly impacts the accuracy of simulation results. Several sampling and approximation approaches have been explored to address this issue (Barton 2012; Xie et al. 2014; Poropudas et al. 2011; Law 2013). These methods basically aim to discover the statistical distributions of the input variables and the reliability cannot be guaranteed without considering the underlying impacting factors (Li and Ji 2019). Furthermore, a decision maker may desire to simulate a future or virtual event, in which scenario no historical data is available for sampling or approximation due to the shift of configuration.

Inspired by the recent success of machine learning applications in various domains, we explore to incorporate machine learning techniques into input parameter modeling for logistics network simulation in this work. In the logistics network operated by JD Logistics, the performance (e.g., processing time) of the nodes and routes are critical inputs of the simulation model. The key challenge we are facing to estimate them is that in such a huge logistics network, the performance of each component is impacted by very complex external factors, thereby posing difficulties on the accurate estimation. Therefore, We intend to investigate the underlying factors that impact the input data distribution and mathematically model the impact using machine learning techniques. In this approach, the components (e.g., nodes and routes in the logistics network) are represented by a list of features extracted from the impacting factors, which are subsequently mapped to the desired input data using a model trained on historical record. The advantage of this approach is reflected by both the accuracy and robustness. For instance, the characteristics of a component may shift if the operation policy is updated, in which scenario the data sampled from historical record can not be used directly for statistical modeling. In contrast, the machine learning based modeling can handle this update appropriately and generate reliable input data for simulation as long as the values of features are clearly given.

Due to the aforementioned advantages, machine learning technique has attracted attentions from the simulation community and is employed to facilitate the simulation systems for multiple purposes (Negahban 2017; Elbattah et al. 2018; Feng et al. 2018; Batata et al. 2018; Giabbanelli 2019; Hoog and Sander 2019; Li and Ji 2019). In (Negahban 2017), simulation model calibration is carried out using deep neural network to reduce the uncertainty in agents' rules. In (Hoog and Sander 2019) and (Feng et al. 2018), deep neural networks have been designed to emulate the simulation system outputs from inputs, which accelerates the evaluation of actions by avoiding iterative execution of the simulation procedure. In (Elbattah et al. 2018), clustering methods are used to gain insights from patients' characteristics to facilitate care pathway design. In (Li and Ji 2019), Bayesian deep neural network is investigated to estimate input data distribution considering the impact of external factors for road paving operation. Despite the success of these recent attempts, there still exists unsolved issues.

Through analyzing the data, we found that the factors impacting input data distribution can be categorized into static factors (e.g., location based factors) and dynamic factors (e.g., recent volume of shipping bills), where dynamic factors can reflect the important time-varying characteristics. While static factors are time-invariant and directly available, dynamic factors are varying over the time horizon. It means that the future values of dynamic factors are not known at the current moment and can be obtained only by simulation or prediction, which usually subject to high cost and non-negligible errors. For instance, in the prediction of distribution sorting time, it is natural to consider the recent volume of shipping bills near the sorting date as features for modeling. However, this information cannot be available if the simulation analyzes far future events, thereby posing difficulties on the accurate prediction. The same problem can also be encountered when considering other dynamic factors.

To address this issue, we investigate the data patterns of dynamic factors using unsupervised learning and transfer them into static features, thereby improving the robustness of the modeling framework. To be specific, clustering techniques are employed to discover the similarity between logistics network components in terms of various dynamic factors. We notice that although dynamic factors are time-varying, they can reflect time-invariant characteristics. For instance, the volume of shipping bills of a distribution center is affected by the factors such as scale, processing capability and the regional level of consumption. The clustering methods can help extract the relevant information by discovering distinguishable patterns and assign a component to the type of pattern it belongs to. The discovered patterns are subsequently used as features of the supervised machine learning model instead of the raw dynamic factors. This effectively overcomes the difficulty induced by the lacking of future data record. Note that there are a variety of input parameters for a complex simulation model. In this work, we take the estimation of sorting times in distribution centers as an example to validate our methodology since it has a crucial impact to the simulation result. However, the proposed methodology can be easily generalized to the estimation of other input parameters such as loading, unloading and transportation times. Thus, the contributions of this work are summarized as follows

- A machine learning framework is formulated to analyze the key input parameters of a large-scale logistics network associated with the underlying impacting factors.
- Unsupervised learning techniques are employed to discover the data pattern of dynamic factors and resolve the difficulty induced by the lacking of future data.
- The proposed method is validated on the estimation of distribution center sorting time extracted from real JD Logistics data. As demonstrated by the experimental results, it outperforms the conventional method on estimation accuracy.

The remainder of this paper is organized as follows. The machine learning framework for input parameter modeling is presented in Section 2. The patterns of dynamic factors are analyzed and transferred using unsupervised learning in Section 3. Experiments are conducted to evaluate the proposed method in Section 4. Finally, we conclude in Section 5.

2 MODELING FRAMEWORK

Given a logistics network, suppose that the simulation input we want to estimate is the distribution center sorting time y_i, where $i \in \{1,2,\ldots,N\}$ represents the index of a shipping bill. For each y_i, we consider m impacting factors represented by $X_i \in \mathbf{R}^m$. In this scenario, we desire to learn a mathematical mapping from X_i to y_i $\forall i \in \{1,2,\ldots,N\}$, denoted by

$$y_i = \phi(g(X_i)) + \varepsilon_i, \tag{1}$$

where $\phi(\cdot)$ is the mapping function, $g(\cdot)$ is a projection that transfers X_i into the format required by $\phi(\cdot)$ and ε_i is a noise term. In the developed framework, XGBoost (Chen and Guestrin 2016) is selected for the modeling of $\phi(\cdot)$ since it has achieved state-of-the-art results on a wide variety of machine learning tasks. Although the developed framework is not limited to a specific machine learning technique, we include the key steps of XGBoost in this section to make the technical details more clear to the readers. Thus, the mathematical mapping $\phi(\cdot)$ is presented in the form (Chen and Guestrin 2016)

$$\hat{y}_i = \phi(g(X_i)) = \sum_{k=1}^{K} f_k(g(X_i)), \tag{2}$$

where each term $f(\cdot)_k$ is constructed by a regression tree and $\hat{y}_i$ is the estimated value of y_i. Given the form in Eqn. (2), XGBoost aims to minimize the estimation error over all the data samples subject to regularization constraints (Chen and Guestrin 2016)

$$L(\phi) = \sum_i l(y_i, \hat{y}_i) + \sum_k \Omega(f_k(\cdot)), \tag{3}$$

where $l(\cdot)$ is mean-squared error for regression tasks and the regularization term penalizes the complexity of the trees to avoid overfitting, denoted by (Chen and Guestrin 2016)

$$\Omega(f(\cdot)) = \gamma T + \frac{1}{2}\lambda \|\omega\|^2. \tag{4}$$

T and ω are the total numbers and scores of leaves in the trees, respectively. γ and λ are hyper-parameters. Due to the indifferentiable nature of a tree ensemble model, the objective depicted in Eqn. (3) and Eqn. (4) is optimized in a greedy manner, such that at iteration k a regression tree $f_k(\cdot)$ aims to minimize (Chen and Guestrin 2016)

$$L^k = \sum_i l(y_i, \hat{y}_i^{k-1} + f_k(g(X_i))) + \Omega(f_k(\cdot)), \tag{5}$$

where $\hat{y}_i^{k-1}$ is the estimated value of data sample i obtained at iteration $k-1$. It is further approximated using second-order gradient as (Chen and Guestrin 2016)

$$L^k = \sum_i \left[l(y_i, \hat{y}_i^{k-1}) + \frac{\partial l(y_i, \hat{y}_i^{k-1})}{\partial \hat{y}_i^{k-1}} f_k(g(X_i)) + \frac{1}{2}\frac{\partial^2 l(y_i, \hat{y}_i^{k-1})}{\partial^2 \hat{y}_i^{k-1}} f_k(g(X_i))^2 \right] + \Omega(f_k(\cdot)), \tag{6}$$

which is subsequently solved using the bags of tricks described in (Chen and Guestrin 2016).

For our estimation task, it is extremely important to identify those dominant factors that have the most impact on the sorting time from the X_i. Based on the domain knowledge from logistics network management, we focus on the time that the sorting activity happens and the characteristics of distribution centers, which can be divided into static factors and dynamic factors. For static factors, we consider geographical and scale factors of the distribution centers. For dynamic factors, we consider the recent volume and distribution of shipping bills processed by the distribution centers. For a distribution center, the volume of shipping bills refers to the average number of shipping bills processed each day and the distribution of shipping bills refers to the number of shipping bills processed at each time slot of the day. The dynamic factors contain rich information about the scale, processing capability and schedule of the distribution centers, thereby having important impact to the sorting time. However, as discussed in Section 1, the relevant data cannot be available when considering a far future event. This motivates us to transfer the dynamic factors into static features via the projection $g(\cdot)$, which is studied in Section 3.

3 DYNAMIC FACTOR ANALYSIS

In order to discover the underlying patterns of the dynamic factors, unsupervised learning is adopted to analyze the volume and distribution of recent processed shipping bills for each distribution center. Considering the unique characteristics of each factor, different unsupervised learning techniques are applied for the information extraction from each one.

3.1 Volume Analysis

For each distribution center, the volume of shipping bills is analyzed using clustering method such that the distribution centers with similar volume of shipping bills can be arranged in the same group. Subsequently, the index of clusters can be used as features for the estimation of sorting time. Although there exists a number of clustering methods, we choose hierarchical clustering because we expect to generate relatively smaller clusters that can accurately represent the within-cluster similarities (Zhang et al. 2013; P. Tan and Kumar 2006).

In order to illustrate the clustering procedure, we consider a set of distribution centers denoted by $\mathscr{D}$. For each distribution center $j \in \mathscr{D}$, we compute the average volume of shipping bills for each day over a recent time period, which is denoted by b_j. As mentioned in Section 2, it contains the global scale and processing capability information of each distribution center. Subsequently, hierarchical clustering is used for the discovery of data patterns. The hierarchical clustering method is executed in a greedy fashion. It initializes each distribution center as a cluster and iteratively merge the clusters with minimum distance together until the desired number of clusters is achieved. For the computation of distance between two clusters, complete-link method is utilized since it can naturally break down the long connection between data points by penalizing the large clusters in the merging step and form relatively smaller clusters (Zhang et al. 2013). Thus, the distance between two clusters ψ_i and ψ_j is defined as

$$d(\psi_i, \psi_j) = \max |b_\alpha - b_\beta|, \forall b_\alpha \in \psi_i, b_\beta \in \psi_j. \tag{7}$$

Furthermore, note that the variable for clustering b_j is a scalar, the clustering procedure can be accelerated by pre-ordering the distribution centers according to the value of b_j and only compute the distances between adjacent clusters.

A commonly used technique to automatically determine the optimal number of clusters along with hierarchical clustering is L-method (Salvador and Chan 2004). It varies the number of desired clusters from 1 to $|\mathscr{D}|-1$ and for each desired number of clusters t, it records the distance between the latest merged clusters as θ_t. Given the decreasing sequence $\theta = [\theta_1, \theta_2, \dots, \theta_{|\mathscr{D}|-1}]$, it aims to discover the location with sharpest transition. To implement this, we separate θ into two parts, which are $\theta_l = [\theta_1, \dots, \theta_c]$ and $\theta_r = [\theta_{c+1}, \dots, \theta_{|\mathscr{D}|-1}]$. Subsequently, θ_l and θ_r are fitted using least squared method and the corresponding root-mean-squared errors are denoted by e_l and e_r, respectively. Finally, the optimal number of clusters c^* is computed by minimizing the weighted sum of the root-mean-squared errors as

$$c^* = \mathbf{argmin}_c c \cdot e_l + (|\mathscr{D}| - 1 - c) \cdot e_r. \tag{8}$$

Considering all the above procedures, the main steps of the hierarchical clustering for shipping bill volume analysis are presented in Algorithm 1.

3.2 Distribution Analysis

Similar to Subsection 3.1, the distribution of shipping bills for each distribution center is analyzed using clustering method as well. The distribution of shipping bills for clustering is constructed as follows.

- Uniformly divide a day into T time slots.
- For each distribution center j, take the average volume of shipping bills in a recent time window at each time slot t as $v_{j,t}$.

Algorithm 1 Hierarchical Clustering for Shipping Bill Volume Analysis.

```
Obtain the average volume of shipping bills $b_j$ for each distribution center $j \in \mathscr{D}$.
Initialize each distribution center as a cluster and hence the number of initial clusters is $t = |\mathscr{D}|$.
while $t > 1$ do
    $t = t - 1$.
    Use Equation (7) to compute the distance between any two adjacent clusters.
    Merge the two clusters with minimum distance and record the distance as $\theta_t$.
end while
Use Equation (8) to compute the optimal number of clusters $c^*$.
Return the clustering result with $c^*$ clusters.
```

- Construct a vector $\mathbf{v}_j = [v_{j,1}, v_{j,2}, \dots, v_{j,T}]$ for each distribution center $j \in \mathscr{D}$ and normalize it such that $\sum_t v_{j,t} = 1$.

Apart from the local processing capability information, the feature vector $\mathbf{v}_j$ also contains the scheduling information of distribution center j.

Subsequently, clustering algorithm is applied on $\mathbf{v}_j$ to group the distribution centers with similar shipping bill distribution patterns together. Among the existing clustering methods, spectral clustering is selected because it can provide robust solution for high-dimensional and noisy data (Zhang et al. 2011; Ng et al. 2001; Bach and Jordan 2003). Given the data vectors, it first construct a similarity matrix $\mathbf{W} \in \mathbf{R}^{|\mathscr{D}| \times |\mathscr{D}|}$, where $\mathbf{W}_{ij}$ is the similarity between distribution centers i and j, and hence $\mathbf{W}$ is symmetric. Among the multiple approaches to construct the similarity matrix, we choose the widely used Gaussian similarity. Thus, $\mathbf{W}_{ij}$ is computed by (Zhang et al. 2011; Ng et al. 2001; Bach and Jordan 2003)

$$\mathbf{W}_{ij} = \begin{cases} \exp - \frac{\|\mathbf{v}_i - \mathbf{v}_j\|^2}{2\sigma^2}, \; i \neq j \\ 0, \; i = j \end{cases} . \tag{9}$$

Subsequently, we compute the degree matrix (Zhang et al. 2011; Ng et al. 2001; Bach and Jordan 2003)

$$\mathbf{D} = diag(d_{11}, d_{22}, \dots, d_{|\mathscr{D}||\mathscr{D}|}), \tag{10}$$

where $d_{ii} = \sum_j \mathbf{W}_{ij}$, and the Laplacian matrix (Zhang et al. 2011; Ng et al. 2001; Bach and Jordan 2003)

$$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{W} \mathbf{D}^{-\frac{1}{2}}. \tag{11}$$

Suppose that the desired number of clusters is c. We obtain the top c largest eigenvalues of $\mathbf{L}$ and the corresponding eigenvectors, denoted by $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_c\}$. The eigenvectors are stacked column-wise and form the matrix $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_c] \in \mathbf{R}^{|\mathscr{D}| \times c}$. Subsequently, we obtain the matrix $\mathbf{A} \in \mathbf{R}^{|\mathscr{D}| \times c}$ by normalizing each row of $\mathbf{U}$ such that $\mathbf{A}_{ij} = \mathbf{U}_{ij} / (\sum_j \mathbf{U}_{ij}^2)^{1/2}$. Finally, we take each row of $\mathbf{A}$ as a data sample representing a distribution center and apply K-means clustering method to assign them into c clusters. The optimal number of clusters c and the parameter σ are determined by Calinski-Harabasz score, which is defined as the ratio of the between-clusters dispersion and the within-cluster dispersion (Caliński and JA 1974). Thus, the parameters with the largest Calinski-Harabasz score are finally selected to generate the desired clustering result. Considering all the above procedures, the main steps of spectral clustering for shipping bill distribution analysis are summarized in Algorithm 2.

After clustering, the transferred dynamic factors, which are the cluster indices, of each sample are concatenated with static features and form the feature vector $g(X_i)$. Finally, the feature vector is fed into the XGBoost model $\phi(\cdot)$ for the estimation of sorting time. Thus, the complete procedure for sorting time estimation is summarized in Algorithm 3.

Algorithm 2 Spectral Clustering for Shipping Bill Distribution Analysis.

```
Construct the shipping bills distribution vector $\mathbf{v}_j$ for each distribution center $j \in \mathscr{D}$.
Define the candidate set of parameters $c$ and $\sigma$ as $\mathscr{C}$ and $\Sigma$.
for $c \in \mathscr{C}$ and $\sigma \in \Sigma$ do
    Compute the similarity matrix, degree matrix and Laplacian matrix according to Equation (9), (10) and (11).
    Obtain the eigenvectors corresponding to the top $c$ eigenvalues and construct the matrix $\mathbf{U}$ as $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_c]$.
    Normalize each row of $\mathbf{U}$ and obtain the matrix $\mathbf{A}$ as $\mathbf{A}_{ij} = \mathbf{U}_{ij}/(\sum_j \mathbf{U}_{ij}^2)^{1/2}$.
    Apply K-means method to assign the rows of $\mathbf{A}$ into $c$ groups and compute the Calinski-Harabasz score.
end for
Return the clustering result with largest Calinski-Harabasz score.
```

Algorithm 3 Sorting Time Estimation.

```
Given a set of data samples, separate the impacting factors $X_i$ into static factors and dynamic factors for each $i \in \{1, 2, \ldots, N\}$.
Use Algorithms 1 and 2 to transfer dynamic factors into cluster indices, named as transferred features.
Concatenate the static factors with transferred features to form feature vector $g(X_i)$, $\forall i$.
Feed $g(X_i)$ into XGBoost model for the estimation of sorting time.
```

4 EXPERIMENTAL RESULTS

The proposed modeling framework is validated on the distribution center sorting time estimation task, which is extracted from the real logistics network simulation data of JD Logistics. We consider 350 distribution centers and divide each day into 144 time slots, where the length of each time slot is 10 minutes. Under this logic, we take a set of data samples from historical record and separate them into 135196 training samples and 81959 testing samples according to the recording date. While each sample represents the record of a shipping bill, the features represent the static factors (location, type, scale, etc.) and dynamic factors (volume and distribution of shipping bills) of the distribution center and the index of the time slot in which the shipping bill is sorted. The label is the sorting time of the corresponding shipping bill. Subsequently, the training samples are used for the study of dynamic factors and model training. The testing samples are used to evaluate the performance of the model. The key parameters of XGBoost are selected using cross-validation on the training samples.

4.1 Dynamic Factor Patterns Discovery

The data patterns of the dynamic factors are analyzed in this part. We construct the features using the shipping bill information of each distribution center as described in Subsections 3.1 and 3.2. Subsequently, hierarchical clustering and spectral clustering are applied to the constructed features and assign the distribution centers into clusters.

The result of hierarchical clustering is shown in Fig. 1, where we present the merging distance sequence θ and the least-squared fitting lines. Based on the weighted sum of root-mean-squared error, the distribution centers are grouped into 14 clusters.

The result of spectral clustering is shown in Fig. 2. In order to present the high-dimensional data, we reduce the dimension of the shipping bill distribution vectors using t-SNE (Maaten and Hinton 2008). Subsequently, the reduced data samples are shown in a 2-dimensional space while the cluster label of a data sample is represented by the color. Note that Fig. 2 shows the low dimensional representations of the

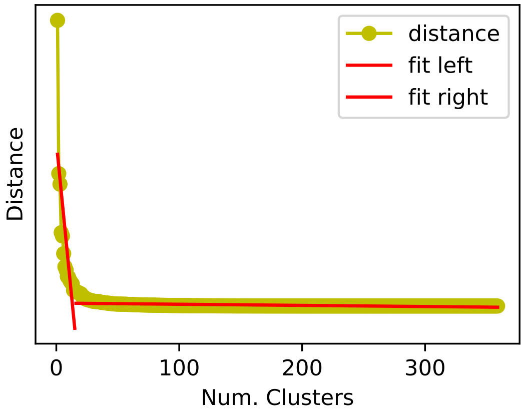

Figure 1: The distribution centers are grouped into 14 clusters based on the volume of shipping bills using hierarchical clustering.

shipping bill distribution vectors and the X and Y values of a point do not have a physical meaning. Thus, the X and Y axis of Fig. 2 are not labeled. It can be observed from Fig. 2 that the distribution centers are divided into 33 clusters including 3 major clusters and several minor ones. After clustering, the cluster indices obtained from both techniques are used as features of the data samples.

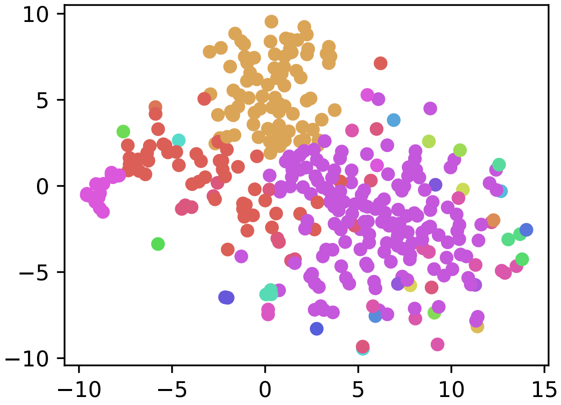

Figure 2: The distribution centers are grouped into 33 clusters based on the distribution of shipping bills over the day using spectral clustering.

4.2 Sorting Time Estimation

In this part, XGBoost is employed to train the estimation model using the training samples. Comparison study is also conducted, where the sorting time of a testing sample is estimated by the average sorting time of training samples corresponding to the same distribution center at the time slot. This is referred to as "average method". The estimation results of the aforementioned methods on testing samples are compared in terms of mean absolute percentage error (MAPE).

The absolute percentage error for each data sample of the proposed method and average method are shown in Fig. 3(a) and Fig. 3(b), respectively. It can be obviously observed from the figures that the

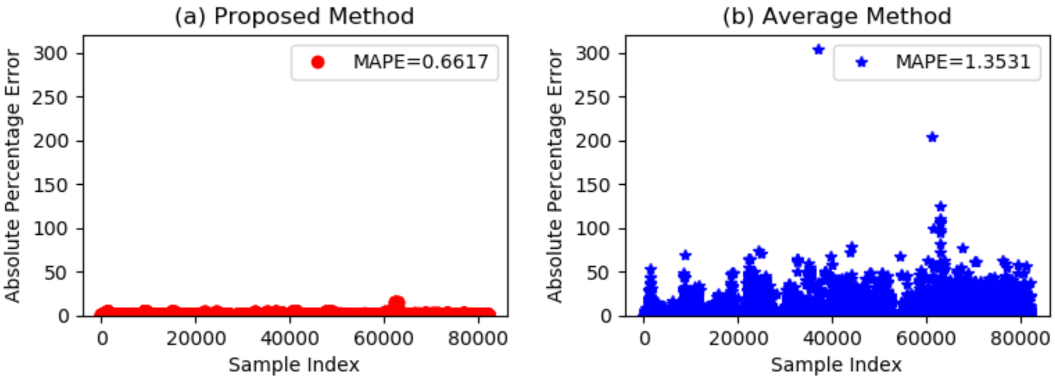

Figure 3: The MAPE of the proposed method is 0.6617 while that of average method is 1.3531.

estimation error of the proposed method is much smaller than that of the average method. Furthermore, the MAPE of the proposed method is 0.6617 while that of average method is 1.3531, which is more than twice of the proposed method. That is because it considers various impacting factors and learns the underlying connections of them with distribution center sorting time, other than simply smoothing on the historical records.

Another important observation we can make is that the sorting time of 3715 testing samples cannot be estimated using average method. That is because the distribution center and time slot combinations of these data samples have not appeared in the training samples. This can verify the aforementioned weakness of sampling and statistical based methods, which is the vulnerability to the absence of data with the same configuration. However, the proposed method is not limited to this issue and can always generate a reasonable result even when the considered configuration have not been visited in the past experience.

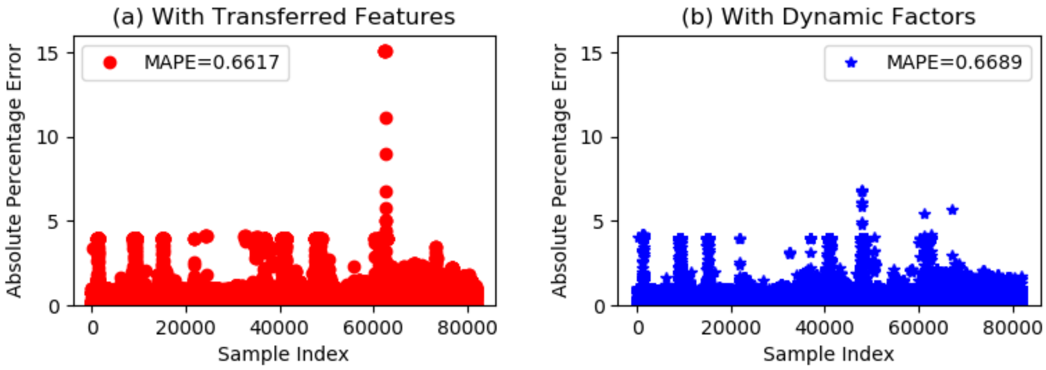

Figure 4: The MAPE is 0.6617 when transferring dynamic factors using clustering techniques while it is 0.6689 when directly using the dynamic factors as features.

In order to analyze the impact induced by transferring dynamic factors into static features, another set of experiment is conducted. In this set of experiment, we keep the general settings, but directly use the raw dynamic factors as features other than transferring them through the clustering methods in Section 3. The absolute mean errors by using transferred features and directly using raw dynamic factors are shown in Fig. 4(a) and Fig. 4(b), respectively. Intuitively, the feature transformation should induce a loss of information and degrade the modeling accuracy. However, as can be observed from these figures, the estimation errors in these scenarios are very close. It means that transferring the dynamic factors into static features did not sacrifice the estimation accuracy. On the contrary, there is a slight reduction on the estimation error. That is because the clustering operation is able to extract the dominant underlying patterns from the dynamic factors and mitigate the interference induced by the fluctuations.

Table 1: The sorting time estimation using different regression methods and feature engineering mechanisms are compared here. For each combination of regression method and feature engineering mechanism, the resulted MAPE is recorded in this table.

	XGBoost	CART
Static + Hierarchical&Spectral Clustering	0.6617	1.8090
Static + Spectral Clustering	0.6636	1.9363
Static + Hierarchical Clustering	0.6668	1.9922
Static Features Only	0.6670	2.2513
Clustered Dynamic Features Only	0.7239	5.2311

Although our proposed modeling framework is not limited to any specific regression method, different technique may induce different estimation quality due to the prediction capability. Furthermore, feature engineering is also a critical part that impacts the modeling accuracy. As regression techniques and feature engineering mechanisms are both important to the developed estimation framework, we would like to evaluate the contribution of each part in order to draw a deeper insight of the sorting time estimation problem. For regression techniques, we compare XGBoost and classification and regression tree (CART) to show the improvement brought by the boosting mechanism. For feature engineering, we compare the proposed combination of features (Static+Hierarchical&Spectral Clustering) to other sets of features consisted of only static features, only clustered dynamic features and the feature sets containing static features and one clustered dynamic feature. The MAPEs for the combinations of feature engineering mechanism and regression method are summarized in Table 1. It can be observed from these results that XGBoost demonstrates a much superior estimation accuracy compared to CART on all combinations of features due to the advantages of boosting. We can also observe from the comparison of features that the model with only static features can serve as a reasonably good base model while the clustered dynamic features can bring additional improvements. In addition, simultaneously using hierarchical clustered features of the shipping bill volume and spectral clustered features of shipping bill distribution can further reduce the MAPE than using only one clustered feature.

5 CONCLUSION

In this work, a machine learning framework is developed for the input parameter modeling of large-scale logistics networks simulation. It uses supervised learning to construct a mapping from the impacting factors to desired input data. Furthermore, it applies clustering techniques to analyze the dynamic factors and transfer them into static features. This mitigates the limitation induced by the lacking of future data. The proposed modeling framework is validated using the real logistics network simulation data from JD Logistics. As demonstrated by the experimental results, the proposed modeling framework outperforms conventional method on estimation accuracy.

REFERENCES

Bach, F. R., and M. I. Jordan. 2003. “Learning spectral clustering”. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, 305–312.

Barton, R. R. 2012. “Tutorial: Input uncertainty in outout analysis”. In *Proceedings of the 2012 Winter Simulation Conference*, edited by C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. M. Uhrmacher, 1–12. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Batata, O., V. Augusto, and X. Xie. 2018. “Mixed machine learning and agent-based simulation for respite care evaluation”. In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 2668–2679. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Caliński, T., and H. JA. 1974, 01. “A dendrite method for cluster analysis”. *Communications in Statistics - Theory and Methods* 3:1–27.

Chen, T., and C. Guestrin. 2016. “XGBoost: A scalable tree boosting system”. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794.

Elbattah, M., O. Molloy, and B. P. Zeigler. 2018. “Designing care pathways using simulation modeling and machine learning”. In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 1452–1463. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Feng, K., S. Chen, and W. Lu. 2018. “Machine learning based construction simulation and optimization”. In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 2025–2036. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Ghiani, G., G. L., and R. Musmanno. 2013. *Introduction to Logistics Systems Management*. 2nd ed. Chichester, West Sussex, United Kingdom: John Wiley Sons.

Giabbanelli, P. J. 2019. “Solving challenges at the interface of simulation and big data using machine learning”. In *Proceedings of the 2019 Winter Simulation Conference*, edited by N. Mustafee, K.-H. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, 572–583. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Hoog, V. D., and Sander. 2019. “Surrogate modelling in (and of) agent-based models: A prospectus”. *Computational Economics* 53(3):1245–1263.

Law, A. M. 2013. “A tutorial on how to select simulation input probability distributions”. In *Proceedings of the 2013 Winter Simulations Conference*, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, 306–320. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Li, Y., and W. Ji. 2019. “Enhanced input modeling for construction simulation using bayesian deep neural networks”. In *Proceedings of the 2019 Winter Simulation Conference*, edited by N. Mustafee, K.-H. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, 2978–2985. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Maaten, L., and G. Hinton. 2008. “Visualizing high-dimensional data using t-SNE”. *Journal of Machine Learning Research* 9:2579–2605.

Negahban, A. 2017. “Neural networks and agent-based diffusion models”. In *Proceedings of the 2017 Winter Simulation Conference*, edited by W. K. V. Chan, A. D’Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, 1407–1418. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Ng, A. Y., M. I. Jordan, and Y. Weiss. 2001. “On Spectral Clustering: Analysis and an Algorithm”. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, 849–856.

P. Tan, M. S., and V. Kumar. 2006. *Introduction to Data Mining*. Addison-Wesley.

Poropudas, J., J. Pousi, and K. Virtanen. 2011. “Multiple input and multiple output simulation metamodeling using Bayesian networks”. In *Proceedings of the 2011 Winter Simulation Conference*, edited by S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. Fu, 569–580. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Rabe, M., M. Ammouriova, and D. Schmitt. 2018a. “Improving the performance of a logistics assistance system for materials trading networks by grouping similar actions”. In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 2861–2872. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Rabe, M., M. Ammouriova, and D. Schmitt. 2018b. “Utilizing domain-specific information for the optimization of logistics networks”. In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 2873–2884. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Rabe, M., and A. A. Scheidler. 2014. “An approach for increasing the level of accuracy in Supply Chain simulation by using patterns on input data”. In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, 1897–1906. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Rushton, A., P. C., and P. Baker. 2006. *The Handbook of Logistics and Distribution Management*. 3rd ed. London: Kogan Page.

Salvador, S., and P. Chan. 2004. “Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms”. In *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence*, 576–584.

Xie, W., B. L. Nelson, and R. R. Barton. 2014. “Statistical uncertainty analysis for stochastic simulation with dependent input models”. In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Y. Diallo, I. O. Ryzhov,

L. Yilmaz, S. Buckley, and J. A. Miller, 674–685. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Zhang, J., G. Sudre, X. Li, W. Wang, D. J. Weber, and A. Bagic. 2011. "Clustering linear discriminant analysis for MEG-based brain computer interfaces". *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 19(3):221–231.

Zhang, W., X. Li, S. Saxena, A. Strojwas, and R. Rutenbar. 2013. "Automatic clustering of wafer spatial signatures". In *Proceedings of the 50th ACM/EDAC/IEEE Design Automation Conference*, 1–6.

Zhou, E., and W. Xie. 2015. "Simulation optimization when facing input uncertainty". In *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 3714–3724. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

YANG LIU is a machine learning engineer at JD Logistics. He received his Ph.D. degree from Michigan Technological University. His research focuses on the industrial applications of data science and artificial intelligence techniques. His e-mail address is liuyang130@jd.com.

LIANG YAN is a algorithm engineer at JD logistics. He obtained his M.S. degree from Shenzhen University, and has worked as a network planning engineer at the S.F. EXPRESS from 2014 to 2018. His research interests focus on simulation and optimization of large-scale logistics systems. His e-mail address is yanliang3@jd.com.

SHENG LIU is an associate professor at the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include combinatorial optimization, intelligent logistics, intelligent manufacturing. He is a technical consultant at JD Logistics. His e-mail address is sheng.liu@ia.ac.cn.

TING JIANG is a machine learning engineer at JD Logistics. She received her M.S. degree on Statistics from Renmin University of China. She has been with JD Logistics since 2011, where she is leading the artificial intelligence team. She focuses on the application of data science, machine learning and computer vision techniques on logistics management for the intelligenzation of JD Logistics scenarios. Her e-mail address is jiangting@jd.com.

FENG ZHANG is an algorithm engineer at JD Logistics. He received his M.S. degree on Computational Mathematics from Shandong University. He focuses on improving the timeliness of JD routing networks and reducing logistics costs through methods such as machine learning, heuristics, operational research optimization, and simulation optimization. His e-mail address is zhangfeng10@jd.com.

YU WANG is the Director of Operations Research at JD Logistics, leading a RD team to develop intelligent decision tools to optimize the design, planning and operations of complex logistics networks. He received his Ph.D. degree from the University of Pittsburgh, and his B.S. M.S. from Tsinghua University. Before joining JD, he has worked as the Principle Data Scientist at Supply Chain Analytics of Walmart (USA), and Sr. Manager of Operations Research at CSX Transportation Inc. (USA). His e-mail address is bjwangyu3@jd.com.

SHENGNAN "SHANE" WU has 15+ years track record of applying big data analytics and algorithms to help businesses optimize performance. He currently serves as the Chief Data & Analytics Officer at JD (Jing Dong) Logistics, leading the development of its overall data infrastructure, products and intelligent decision systems. Previously, he held various positions in both private and public sectors globally. Dr. Wu is the inventor of several U.S. and China patents, published articles in prestigious journals, and delivered speeches in professional conferences world-wide. He received his Ph.D. degree in Operations Research from the University of Pittsburgh and a B.Eng. from Tsinghua University, respectively. His e-mail address is wushengnan1@jd.com.