




A data-driven system for cooperative-bus route planning based on generative adversarial network and metric learning

Jiguang Wang^{1,2} · Yilun Zhang³ · Xinjie Xing⁴ · Yuanzhu Zhan⁵ ·
Wai Kin Victor Chan^{1,2}  · Sunil Tiwari⁶

Accepted: 10 June 2022 / Published online: 30 September 2022
© The Author(s) 2022

Abstract

Faced with dynamic and increasingly diversified public transport requirements, bus operators are urged to propose operational innovations to sustain their competitiveness. In particular, ordinary bus operations are heavily constrained by well-established route options, and it is challenging to accommodate dynamic passenger flows effectively and with a good level of resource utilization performance. Inspired by the philosophy of sharing economy, many of the available transport resources on the road, such as minibuses and private vehicles, can offer opportunities for improvement if they can be effectively incorporated and exploited. In this regard, this paper proposes a metric learning-based prediction algorithm which can effectively capture the demand pattern and designs a route planning optimizer to help bus operators effectively deploy fixed routing and cooperative buses with traffic dynamics. Through extensive numerical studies, the performance of our proposed metric learning-based Generative Adversarial Network (GAN) prediction model outperforms existing ways. The effectiveness and robustness of the prediction-supported routing planner are well demonstrated for a real-time case. Further, managerial insights with regard to travel time, bus fleet size, and customer service levels are revealed by various sensitivity analysis.

Keywords Data-driven route planning · Generative adversarial network · Urban transportation system · Mixed-integer programming

✉ Wai Kin Victor Chan
chanw@sz.tsinghua.edu.cn

¹ Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen 518005, China

² Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, Shenzhen 518005, China

³ Department of Industrial Engineering and Management, Shanghai Jiao Tong University, Shanghai, China

⁴ Management School, University of Liverpool, Liverpool, UK

⁵ Birmingham Business School, University of Birmingham, Birmingham, UK

⁶ Operation Management & Decision Science Department, ESSCA School of Management, 4 Pont Pasteur, 69007 Lyon, France

1 Introduction

Recent decades have witnessed significant improvements in urban public traffic systems to satisfy dynamic and ever-increasing traveling demand including launching new bus lines (Lyu et al., 2019), developing more intelligent traffic signals (Liang et al., 2019), and designing innovative bus scheduling strategies (Wang et al., 2017). Unfortunately, the performance of above measurements is probably to be restricted since the buses still travel along fixed route lines, accompanied with insufficient capability to cope with dynamic passenger flows from time to time and place to place (Bouton et al., 2015). Hence, the gap between fluctuating demand and fleet capacity during rush hours arises, unbalanced utilization among all buses occurs, which can further result in poor customer satisfaction (e.g., ride difficulties, including catching the bus, and onboard crowdedness for passengers in busy bus lines) and low return on investment (e.g., the empty or under-utilized bus runs). To overcome these difficulties, one may choose to adjust the routes. Nevertheless, a wide-range bus route adjustment on ordinary buses cannot be effective and cost-saving in implementation (Chen et al., 2009). On the other hand, bus companies can invest in additional capabilities to buffer surge demand instead of performing wide-ranged changes to bus route design. However, purchasing new buses will sharpen the unbalanced fleet utilization from both spatial and temporal perspectives, and bring an intensive financial burden. Thus, we can conclude that it is not an effective choice to completely rely on ordinary buses to enhance transportation capabilities. Motivated by the sharing economy, we tend to employ flexible and shareable resources (e.g. minibuses and private vehicles) which are not dedicated to the public transportation operation previously, viewed as the part-time collaborators to assist the ordinary buses. This offers an effective and economic solution for the underlying capacity bottlenecks during peak times, improves the utilization and reduces the cost to better close the gap between the demand and capacity.

We assign new routes which are intrinsically different from the ordinary bus routes to overcome the shortcomings of fixed routes. The key feature of new routes should be traveling across different ordinary bus lines to meet dynamic demand. In this regard, the new vehicles collaborate with ordinary buses in respective ways, and the existing fixed routes are linked to each other by the new vehicles. For easy reference, we name these vehicles “Cooperative-bus” (“Co-bus” as an abbreviation). Note that the Co-bus reserves some capacity for their regular commitments (e.g., shuttle service for some companies, or other own purposes) and deploy the flexible capacity to serve ordinary bus lines whenever necessary and possible. This constitutes an innovative public transportation system operation scheme displayed in Fig. 1, which differs from previous research.

As illustrated in Fig. 1, the purple lines represent the original commuting scheme for the shuttle bus, these shuttle buses provide a commuting service to employees of a certain company. In the new scheme, these shuttle buses can be used as the to-be-deployed Co-bus. Specifically, the purple lines depict the commuting routes of the shuttle bus, which start from the parking yard, *init*, to a residential area, W_1 , and transit employees to the company C_1 . Subsequently, the shuttle bus will be idle until its night shift, when it starts to move employees back to W_1 and performs the shift at the yard *init*. However, this is not an ideal scheme because the capacity of the shuttle bus is not fully utilized. As a comparison, the Co-bus is shown in the green lines in the figure. The Co-bus can operate normal bus lines R_1 and R_2 sequentially after accomplishing the commuting task and returning to the company C_2 . Hence, higher vehicle utilization can be achieved through this scheme compared with the ordinary one.

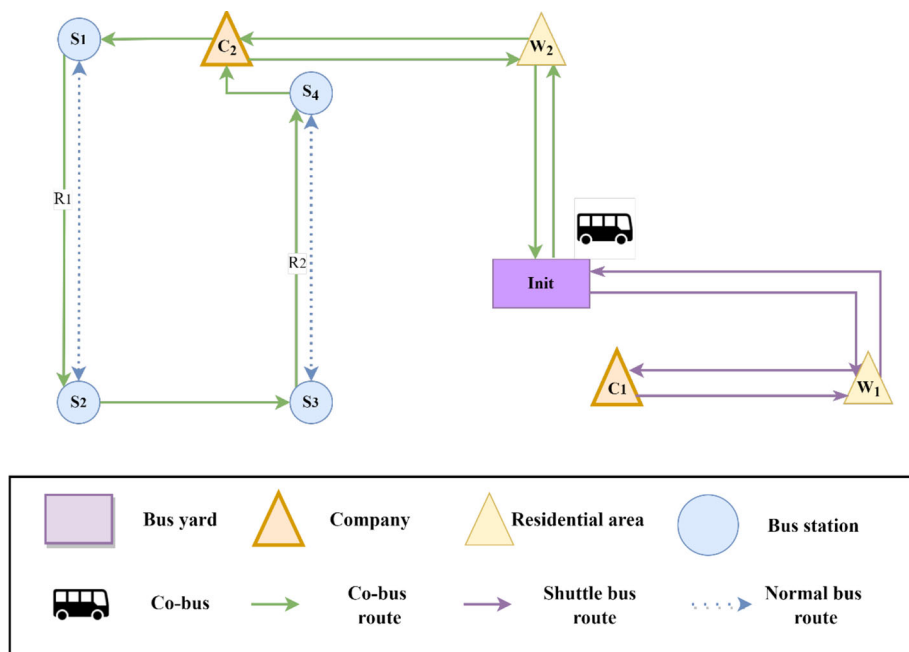


Fig. 1 Diagram of scheme interplayed by a normal bus and a Co-bus

To implement the new scheme, a decision support model needs to be established for Co-bus route planning. It aims at minimizing the gap between the demand and capacity, which is defined as the total demand divided by the total capacity of all vehicles for each bus line. We formulate this problem as a mixed-integer programming (MIP) model considering flow balance constraints, rationality constraints and time continuity constraints, which is an NP-hard problem and accompanied by complexity reduction and acceleration techniques to achieve better accuracy and efficiency in real-world applications.

To obtain high-quality final solutions, an accurate prediction of the passenger traveling demand is required (Kumar et al., 2018). Conventionally, most existing papers are devoted to developing complicated machine learning models (e.g. deep graph convolutional networks) to enhance prediction accuracy. However, since our input data contains various distributions from different time and space segments (or equivalently, different scenarios), the performance improvement brought by conventional approaches might not be significant in our setting. This is because most supervised learning models are not capable of distinguishing mixed distributions, where learnings from one distribution can be distorted by the learning process from others, so that they might unwittingly switch to learn another distribution while recognizing the pattern of one distribution. Alternatively, we need to firstly process our input to be almost one distribution. In this regard, we propose a preliminary scenario segmentation, and then train various models corresponding to each scenario (Yu et al., 2019). In most current studies, scenario segmentation can be realized by clustering algorithms (Davis et al., 2016), such as K-means. However, the conventional K-means algorithm might perform poorly because the difference among each feature has been neglected (Xu et al., 2016), leading to a biased distance metric. We develop an approach to identify the importance of each feature (represented by the weights) according to the clustering result, and modify the

objective of clustering algorithms with respect to the updated feature importance. Note that, this procedure must be iteratively operated since the obtained feature weights also contribute to fine-tuning the clustering procedure by modifying the objective function. In our research, we propose an ensemble learning-based algorithm to learn the best distance metric in the objective, by updating the weights of all features adaptively. The numerical results illustrate that the proposed metric learning technique outperforms similar approaches in existing studies.

In summary, our contributions are two-fold. First, we propose a new operation scheme for the public transportation system by incorporating Co-buses to better fulfill the commuting demand through cooperating with ordinary buses in the existing routes and enhancing flexibility and utilization of the entire transportation resources. Accordingly, a large-scale complex optimization problem is proposed and solved. Second, we adopt metric learning to modify the clustering algorithm used for scenario segmentation of traffic flow with updating the weights of all features adaptively, it facilitates the GAN to characterize the demand pattern in each scenario, and each GAN model is fed by a corresponding dataset where all samples follow similar distribution. This results in considerable improvement in prediction accuracy and makes full use of the value of data for decision-making support.

The remainder of this paper is organized as follows. Section 2 briefly reviews relevant literature in this field. Section 3 explains the prediction procedure for travel demand. Section 4 describes the research problem and proposes the mathematical formulation. Section 5 briefly introduces the solution strategy. The numerical experiments are presented in Sect. 6. Finally, the managerial insights, limitations, and future research are presented and discussed in Sect. 7.

2 Literature review

The core of our proposed research is determining a set of optimal routes performed by Co-buses with limited capacity and operational constraints to serve the surging demand in fixed ordinary bus lines. As a variant of the vehicle routing problem (VRP), our research considers the routing choice together with fleet allocation for Co-buses based on existing fixed bus lines (Ning et al., 2017). To articulate our contributions, we summarize the VRP in bus transport both operating on fixed bus lines and flexible routes. In addition, with respect to the dynamic daily demand, we propose a deep-learning-based prediction method for citizen travel demand to enhance the quality of the corresponding routing and capacity allocation decisions. In the following section, we review relevant studies in the domain of demand prediction algorithms and identify the research gaps accordingly.

2.1 VRP in bus transport

VRP in bus transport comprises two types of variants. First, all vehicles are repeatedly operated on fixed routes. Most studies are devoted to bus route planning of fixed routes that constitute a complex urban bus network (Mathew et al., 2015). Borndörfer et al. (2007) propose a multicommodity flow model for line planning and optimizing corresponding departure frequencies to minimize operating costs and travel times. A passenger-oriented approach called the “trip coverage model” has been proposed by Laporte et al. (2007) to integrate the steps of the transit planning model (trip attraction and generation, trip distribution, mode choice, and traffic equilibrium) into an optimization procedure.

Second, vehicle routes vary with time instead of being static or fixed. This is because uncertain demand in the urban network is involved. In related studies, “taxi route planning” and the “shared bus” (Cohen & Kietzmann, 2014) are two typical cases. Marin and Codina (2008) present a taxi planning network design (TPND) model. The TPND model is formulated as a binary multicommodity network flow problem with additional side constraints under a multi-objective approach. Choi et al. (2018) propose a Dantzig–Wolfe decomposition approach to decide the transportation routes and the number and types of vehicles. For the shared bus, Kong et al. (2018) employ a two-stage approach, which comprises travel requirement prediction and dynamic route planning to generate dynamic routes for shared buses. Ning et al. (2021) construct a joint bus scheduling and route planning framework to maximize the number of passengers, minimize the total length of routes and the number of buses required for the “last mile” scenario.

Regarding the above reviewed studies, their approaches have the following gaps that challenge their direct application into our problem. The fixed routing problem fails to meet the dynamic nature of traveling demand (Cordeau & Laporte, 2007), leading to serious resource waste and low operational efficiency. The varying routing type needs to design and operate new dynamic routes and ignore cooperation with buses on current routes (Pillac et al., 2013), which concentrates on internal fleet resources but overlooks the possibility of using the external bus fleet. However, if both can be incorporated, a better synergy of all the vehicle resources can be achieved.

2.2 Learning-based travel demand prediction

Travel demand prediction is essentially a time-series prediction because of the time-dependent property of traffic systems, which requires well-performed models to characterize the demand distribution pattern across temporal and spatial views. Thus, the key issue has become finding approaches with a high accuracy level. In recent years, a variety of publications have proposed abundant alternatives for this purpose, which can be roughly divided into two categories: discriminative models (Eachempati et al., 2021) and generative models (Banerjee et al., 2020). Discriminative models yield predictions directly rather than generating and learning the distribution explicitly (Gordon & Hernández-Lobato, 2020). These models are less likely to be compatible with our dataset containing multi distributions, due to the weak capability of identifying different distributions to prevent unstable performance. In contrast, generative models can be more suitable for our problem and have the appealing ability to learn from the distribution characteristics of unlabeled data.

In general, discriminative models focus on fitting the decision boundary between the classes. Cyprich et al. (2013), in the bus industry, extend Reg-ARIMA to incorporate trading days and holidays. Wang et al. (2013) compare the gray model against elastic coefficient models to predict yearly passenger volume. Jiang et al. (2015) use a gray support vector machine with empirical mode decomposition for passenger flow prediction in high-speed rail. In the past decade, with the improvement of computing power, deep learning models have been able to map complex nonlinear relationships, which reduce errors by iterating to make calculations and predictions, so deep neural networks are increasingly used in traffic time-series prediction. Zhang et al. (2016) propose a deep-learning-based traffic flow prediction model—the stacked autoencoder method. By fusing the convolutional neural network (CNN) and long short-term memory (LSTM) techniques, Ke et al. (2017) formulate a deep learning approach to forecast short-term taxi-passenger demand. Despite the excellent predictive performance of deep discriminative models, several shortcomings exist. First,

overfitting is more likely to be observed and thus produce highly confident predictions due to the limited fitting capabilities in capturing complicated relations and nonlinear patterns, compared with the sample size. Therefore, they require massive labeled data sets for training and validation (Tu, 2007). Further, discriminative models can only derive point estimation without distribution description, and accuracy loss may arise because we do not fully utilize the information extracted from historical data.

In contrast, generative models explicitly quantify the actual distribution pattern, providing more overall profiles of the entire data set. In this aspect, the GAN is frequently utilized. Liang et al. (2018) extend the original GAN proposed by (Goodfellow et al., 2014) to a two-level LSTM neural network, which is capable of capturing the spatial–temporal correlation of network-wide traffic statement estimation. Yu et al. (2019a) use conditional GAN-driven learning approaches to predict taxi-based mobility demand. Ai et al. (2019) propose a D-GAN model via a deep implicit generative model and an unsupervised manner to predict spatiotemporal data. Naji et al. (2021) use a GAN comprised of the recurrent network model and the conventional network model with multi-source data to forecast taxi demand.

Regarding the above reviewed papers, two main research gaps are identified here. First, sample segmentation is not implemented in some of them, which might lead to unstable prediction results caused by mixed distributions. Second, most existing studies fail to reflect true feature importance but use predefined weight for each feature based on experience.

2.3 Summary

To demonstrate the academic significance of our study, the following research gaps are presented. First, most existing studies focus on fixed routes or routes that vary from time to time. These approaches do not lead to a synergy effect of the entire vehicle resource and are hard to manage with dynamic scheduling. It can further lead to an inappropriate configuration when demand is time-dependent. Second, most existing urban transportation forecasting generative models ignore the difference of feature importance or predefine static weight for each feature, which is hard to capture the true metric distance and leads to biased results.

3 Prediction for travel demand

The accuracy of demand prediction will significantly affect the quality of route planning (Chen et al., 2021). Thus, we need to develop a well-performed algorithm to hourly predict travel demand with respect to the time-dependent pattern characterized by the 24-dimensional vector. Each dimension in the vector quantifies the number of customers boarding buses in each hour during a day, respectively. To this end, we choose the generative model instead of the discriminative model, because the generative model can fully explore the potential of data and provide a better profile of distribution patterns. In other words, we tend to generate the demand rather than estimate the value. Thanks to the outstanding learning capability to extract the pattern, we adopt the modified GAN for demand generation, with traffic scenario segmentation boosted by metric learning, as illustrated in Fig. 2. We need to fit different GANs with the data corresponding to each scenario, yielding better outcomes for demand prediction because the sample distribution varies among different scenarios.

We then elaborate on the key procedures to perform demand prediction, including data preprocessing, traffic scenario segmentation, and GAN-based generation.

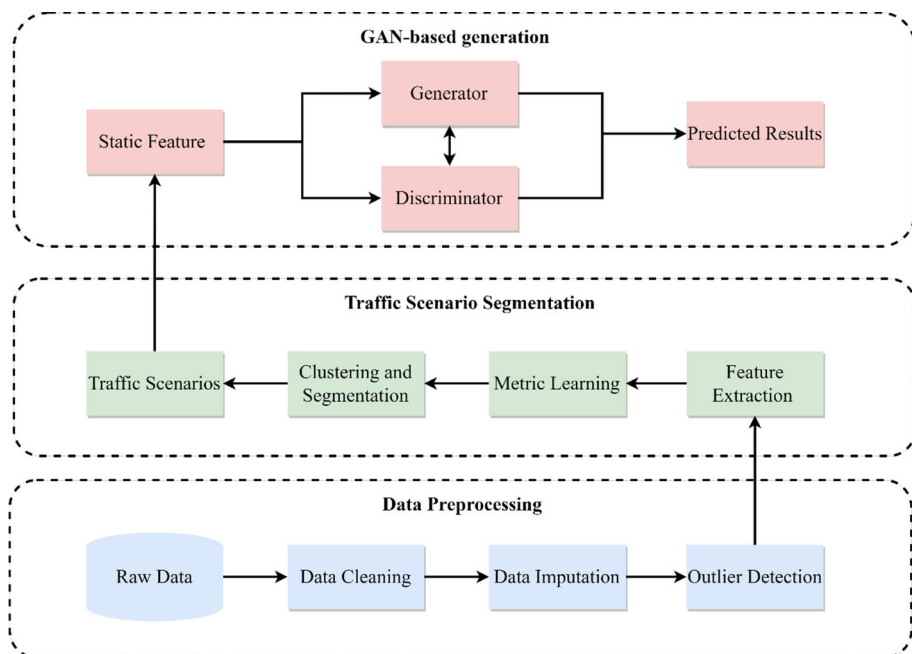


Fig. 2 Framework of generation for travel demand

3.1 Data preprocessing

Typical traffic data preprocessing operations include data cleaning, imputation, and outlier detection (Kumar et al., 2016). We first perform the data cleaning operation to extract useful information from raw data. Subsequently, the Bayesian tensor decomposition approach (Chen et al., 2019) for spatiotemporal traffic data imputation is appropriate to deal with the missing raw data from the automatic card reader and camera. Specifically, this approach extends the Bayesian matrix factorization model to learn the underlying statistical patterns in spatiotemporal traffic data, which can make good use of Bayesian inference for parameter estimation, updating factor matrices and parameters (or hyperparameters) iteratively. The entire structure of the data generation process of Bayesian matrix decomposition is shown in Fig. 3. Then we clear out errors and identify the outliers by using the modified isolation forests method. In this method, the probability of outliers occurring in the sparsely distributed region is very low. The data set is randomly divided recursively until all sample points are isolated, and outliers usually have shorter paths under this random segmentation strategy. Finally, the outliers are replaced using the interpolation method.

3.2 Traffic scenario segmentation

Before predicting travel demand, it is necessary to understand different traffic scenarios (Kumar et al., 2022). Further, different distributions need to be distinguished when predicting travel demand using GAN; otherwise, the performance will be poor as a result of losing distribution consistency between training and application environments. The segmentation

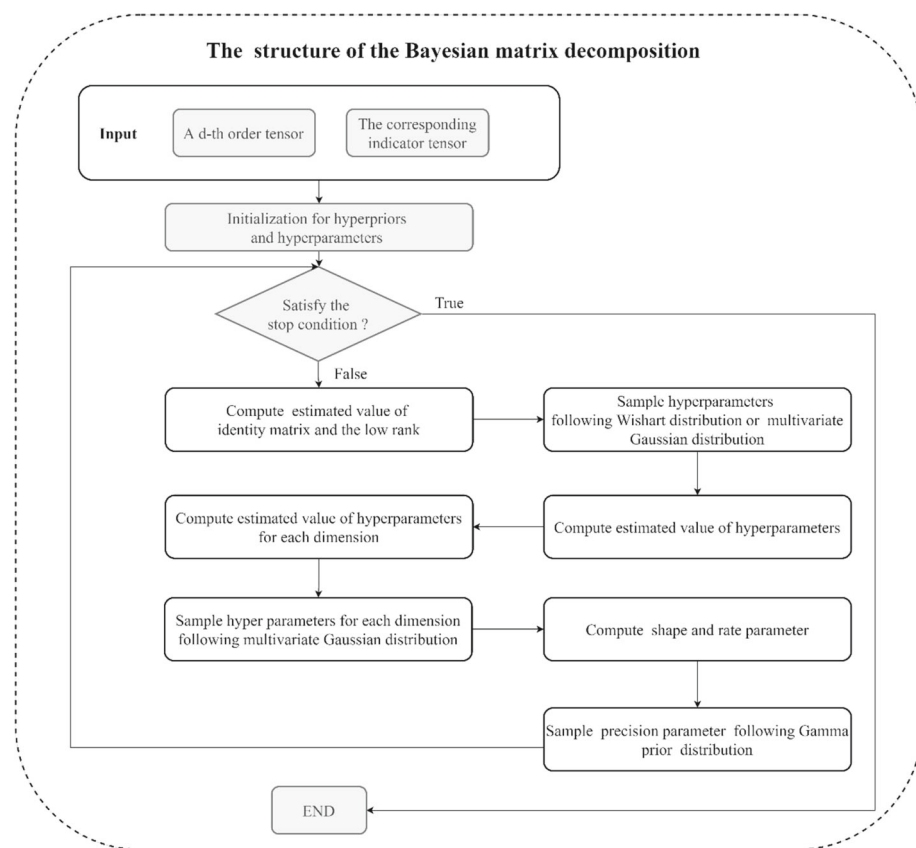


Fig. 3 The entire structure of the data generation process of Bayesian matrix decomposition

for traffic scenarios is essentially a clustering problem, which requires defining a feature vector to characterize behavior and determine the weights for each feature. Most existing clustering algorithms applied for this purpose assume equal weight or assign predefined weights to each feature. However, this fails to reflect the true difference among all feature importance and could be not effective because the manual setting only considers prior experience without a theoretical guarantee, leading to unrealistic clustering results as well as poor clustering performance as measured by some popular metrics, such as the Davies–Bouldin index (Demiriz et al., 1999) or the silhouette coefficient (Aranganayagi & Thangavel, 2007) based on the Euclidean metric or Manhattan distance. Motivated by this, we propose a metric learning method to assign proper weights for each feature, which enhances the effectiveness of the clustering algorithm.

The proposed method can learn the distance metric by updating all features' weights of a clustering algorithm (e.g., K-means) adaptively after justifying the importance by a classifier (e.g., random forests). Our proposed metric learning algorithm combines an unsupervised learning algorithm with a supervised learning algorithm. Figure 4 exhibits the framework of traffic scenario segmentation via metric learning; each step is explained below.

More specifically, we design a modified K-means algorithm, aiming to maximize the mean silhouette coefficient of each cluster for the entire data set. The silhouette coefficient

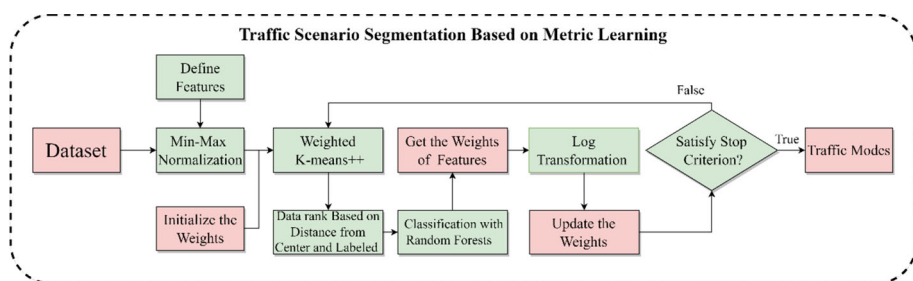


Fig. 4 The framework of traffic scenario segmentation by metric learning

measures how similar a feature vector is to its cluster compared with neighboring clusters (Aranganayagi & Thangavel, 2007). To compute the silhouette coefficient, we apply the weighted Euclidean distance to quantify the similarity.

As previously stated, it is critical to determine the proper weight of each feature. Hence, we need to adopt metric learning to adjust the distance expression to obtain a more reasonable clustering outcome. Metric learning relies on a supervised learning algorithm to interpret the importance of each feature. To this end, we employ random forest, which requires the label of cluster belongings. Note that not all cluster labels deserve high confidence, and only those vectors that are sufficiently close to the cluster center should be declared to be “well-clustered”; that is to say, the cluster label can be highly likely to reflect the ground truth. We select these vectors through a threshold-based rule that restricts the rank threshold (a proportion) of all vectors ranked in ascending order by distance to the corresponding cluster center. Each cluster constitutes the sample set for training, which can guarantee the reliability of random forest.

The random forest will display each feature’s importance, variable importance measures (VIMs), which can be obtained by computing the average change of the Gini index before and after branching in each node of the forest, and the branching process performs splitting in terms of the selected feature. We take normalized VIMs as the feature weight to derive the distance. Once we acquire the new weights by training the random forest, the distance definition also changes, and we need to modify the K-means algorithm, which also updates the input for the next training of random forest. Conclusively, this is essentially an iterative process when updating the weights until convergence meets, which refers to the gap of each weight compared with the last iteration being smaller than a predefined stopping threshold. Note that the new weights need to be logarithmically transformed and normalized before the next update. The pseudocodes are shown in Table 1.

3.3 GAN-based generation

GAN quantifies the actual distribution pattern and provides more overall profiles of the data set based on traffic scenario segmentation. On this basis, we split the data set into several groups, each of which corresponds to one specific scenario and can be employed to train a GAN, which will improve accuracy and convergence.

In particular, GAN consists of a generator and a discriminator (Goodfellow et al., 2014), as shown in Fig. 5.

The generator is a fully connected network. The input features of the generator model cover the time-series distribution of the traffic scenario, length of the bus line, number of bus

Table 1 Framework of weight adaptive metric learning algorithm

Weight adaptive metric learning algorithm:
<i>Input: all serial data A and all parameters</i>
<i>Output: clustering results for all data and final weights</i>
<i>Initialization weight vector \mathbf{w}, learning rate α, decay rate ε, rank threshold θ, accuracy threshold β, stopping threshold η</i>
<i>While stopping condition $\Delta\mathbf{w} < \eta$ is met:</i>
<i>Cluster for all data A using K-means algorithm based on weights \mathbf{w}, obtain the data for each cluster A_k</i>
<i>For k in number of clusters:</i>
<i>Calculate the distance d_{ki} between each data $A_{ki} \in A_k$ and the cluster center</i>
<i>Choose the better data A'_{ki} with the lesser distance d_{ki}, which is less than the threshold θ, and label it as k</i>
<i>End for</i>
<i>Classification using random forests algorithm with all A'_{ki}, and obtain \mathbf{w}' by the importance of the algorithm branching</i>
<i>If classification accuracy is bigger than β:</i>
<i>Update the weight $\mathbf{w} \leftarrow \mathbf{w} + \alpha \cdot \log(1 + \mathbf{w}')$ and normalization</i>
<i>Update the learning rate $\alpha \leftarrow e^{-\beta \cdot \alpha}$</i>
<i>End if</i>
<i>End while</i>
<i>Cluster for all data A using K-means algorithm based on final weights \mathbf{w}</i>
<i>Return clustering results and weights</i>

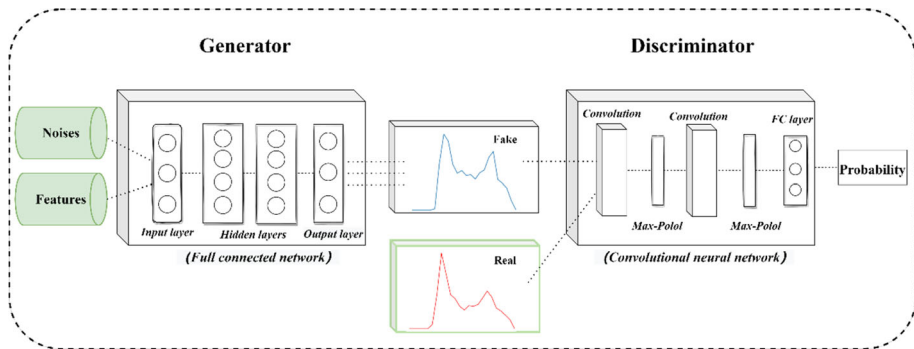


Fig. 5 The framework of GAN

Table 2 Quantitative description of the predictive features

Feature	Definition
Length of the bus line	The length of a bus line from the origin to the terminal
Number of bus stations	The number of stations on a bus line
Number of subway stations	The number of subway stations near the bus station on a bus line
Day type	Of Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, and Holiday, the one the target day belonged to
Weather	Of Sunny day, Rainy day, and Foggy day, the one the target day belonged to

stations, number of subway stations, day type, and weather, all of which need to be processed by min–max normalization. Table 2 presents the definition of these predictive features. The output is the generative travel demand distribution.

- (1) *Length of bus line*: The length of a bus line reflects the operational characteristics of a line to a certain extent. Generally, longer lines will result in lines spanning different areas, representing the dependence of travel behaviors on station location. This property can also reflect the space law from a new perspective.
- (2) *The number of bus stations*: The number of bus stations on a line is a significant factor in demand prediction because the predicted demand can be accumulated from all stations on the line. The number of passengers boarding at each station will affect the total demand.
- (3) *The number of subway stations*: The subway stations near the bus stop on a line will affect passenger choice because of the high commuting efficiency of the subway. In many cases, passengers need to transfer between buses and subways to reach their destinations, leading to large passenger movements occurring at adjacent subway stations and bus stops. Consequently, the number of subway stations adjacent to bus stations is also vital in forecasting.
- (4) *Day type*: The weekly nature of travel demand is also significant. The travel demand of people on weekdays follows “peak–valley” patterns. At weekends and holidays, their travel demand is relatively scattered. Even on all weekdays, the travel rule for each day is not the same (Xu et al., 2016). Further, discrete numeric values are labeled on Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, and Holiday.

- (5) *Weather*: The influence of weather attributes on ground traffic cannot be ignored. Low visibility, heavy rain, and snowstorms all affect people's travel choices. Therefore, Sunny days, Rainy or Snowy days, and Foggy days can be labeled discretely to represent weather information.

The discriminator is a CNN. The inputs of the discriminator comprise the generative travel demand distribution and the real demand distribution, labeled as fake or real. The discriminator is a classifier used to distinguish the real and the fake. When the discriminator fails to tell whether it is true or fake, the distribution generated by the generator is sufficiently close to the real distribution.

The generator and discriminator will be trained simultaneously, and a combat game occurs in this procedure. That is to say, the generator aims to cheat the discriminator, and the discriminator puts every effort into distinguishing the samples from the generator and real data. To implement the training process, we use the stochastic gradient descent method, with the loss function defined as a minimax form. We adjust parameters for G to minimize $\log(1 - D(G(z)))$ and adapt parameters for D to minimize $\log D(x)$.

$$\min_G \max_D (\mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))] \quad (1)$$

4 Mathematical formulation for the route planning model

We formulate this problem as a mixed-integer programming model, aimed at minimizing the gap between time-dependent demand and fleet capacity, to determine the sequence of tasks (existing bus line) for each Co-bus. Note that one bus line might be executed by different Co-buses at the same time, and one Co-bus can serve a bus line more than once.

Our formulation allows the inclusion of practical planning assumptions such as the following:

- Normal buses and Co-buses own the same capacity, they are homogeneous vehicles in our model.
- Customer preference for these buses remains constant.

We then elaborate the objectives, constraints, and parameters below, summarized in the framework shown in Fig. 6.

To address our problem, we introduce the following notations in Table 3 for easy reference. Then we define the following decision variables:

X_{itn} : 1 if Co-bus n operates task i in operating time $t \in T$, 0 otherwise.

Y_{ijn} : 1 if Co-bus n operates task j after task i , 0 otherwise.

Z_{in} : 1 if Co-bus n operates task i , 0 otherwise.

In this scheme, the Co-bus is designed to cooperate with ordinary buses to fill the gap. To better understand this we establish a corresponding network, G , where each node represents the event of running a bus line, consisting of a set of stations in successive order at a certain time, which can be regarded as a task $i \in R$ executed by the Co-bus, where R is defined as the set of all candidate line tasks. Each arc refers to the transfer process from one bus line to another. For instance, one Co-bus $n \in N$ has been scheduled to undertake two tasks successively, which generates a successive task pair $(1, 2)$. Task 1 is a node in the network that represents bus line No. 36 is executed between 10 to 11 AM, while task 2 is another

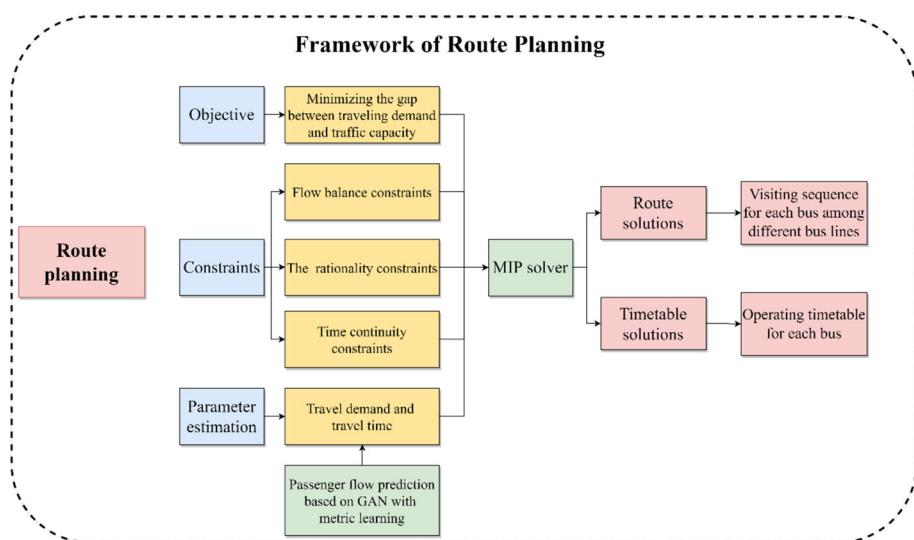


Fig. 6 Framework of route planning

Table 3 Notations

Symbol	Description
G	The bus service network (R, L)
R	Set of all candidate line tasks
L	set of all origin–destination pairs $(i, j), \forall i, j \in R$
P	Set of all periods
R_p	Set of all candidate line tasks in the period $p \in P$
N	Set of all Co-buses
T	Set of all time slots
m_n	The company morning shuttle task for Co-bus n
e_n	The company evening shuttle task for Co-bus n
M	A sufficiently large value
p_i	Number of ordinary buses in task $i \in R$
γ	Driver rest time
b_i	Travel time of a single task $i \in R$
B_{ij}	Transfer time associated with $(i, j) \in L$
D_i	Total customer demand for the task $i \in R$

node that stands for bus line No. M562 is operated at 11 AM to 12 AM. The arc shows the transfer process from task 1 to task 2, as illustrated in Fig. 7.

To this end, a more detailed mathematic example is introduced to describe the entire process of the Co-bus. Co-bus n is a shuttle bus providing a commuting service to employees of a specific company. The Co-bus will be idle in the periods between its morning commuting task and its evening task. In our scheme, Co-bus n departs from the company morning shuttle

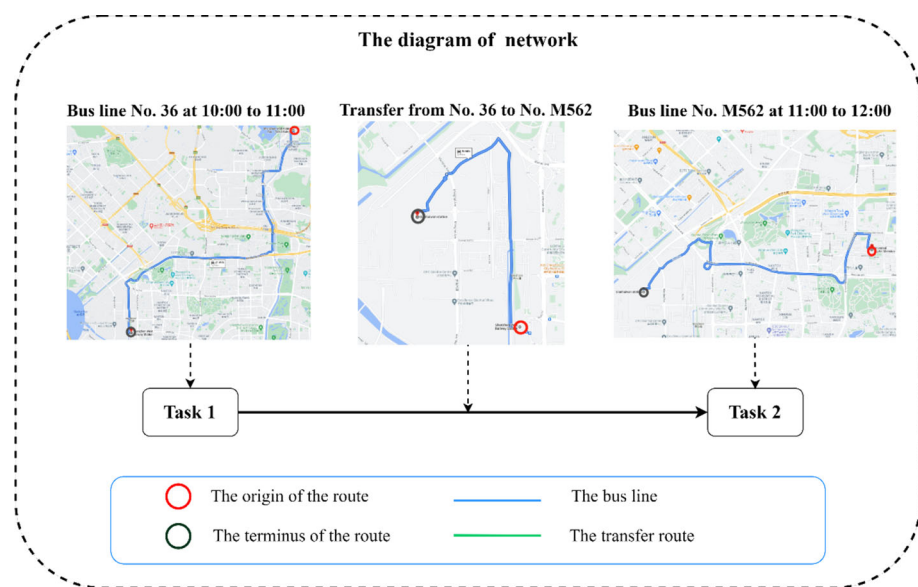


Fig. 7 Diagram of network

task, m_n , then chooses some candidate line tasks in sequence, such as $i, j \in R$, in its available idle time, and returns to the company evening shuttle task, e_n . We denote that the transfer processes as $(m_n, i), (i, j), (j, e_n) \in L$ need to be operated, and the ordinary bus always runs on a fixed bus line. The demand of each task is represented by parameters D_i , referring to the number of total passengers on board in the entire bus line, which is one dimension in the vector produced by GAN in the previous section. The parameter b_i is the travel time of a bus line, which is derived using historical averages based on stationarity tests. The transfer time B_{ij} quantifies the time consumption switching from tasks i to j , which could be calculated from the distance and average travel speed after verification of the stationarity by the augmented Dickey-Fuller (ADF) test.

On this basis, we derive the objective function in Formula (2).

$$\max \sum_{i \in R} \frac{p_i + \sum_{n \in N} Z_{in}}{D_i} \quad (2)$$

We aim to maximize the average service level, which characterizes the gap between demand and capacity. In general, the service level depicts the demand-capacity ratio (Liu et al., 2021). It represents the average amount of resources available to each passenger. Therefore, we adopt the ratio to quantify the service level simplicity, and thus we obtain a linear form. Let $p_i + \sum_{n \in N} Z_{in}$ be the total number of buses for a task i , containing the number of ordinary buses p_i and Co-buses $\sum_{n \in N} Z_{in}$, which measures the capacity.

In addition, we consider some restrictions in real-world applications, such as the flow balance and time continuity conditions.

a. *Flow balance.*

$$\sum_{i \in R} Y_{m_n in} = 1 \quad \forall n \in N \quad (3)$$

$$\sum_{i \in R} Y_{ien} = 1 \quad \forall n \in N \quad (4)$$

$$\sum_{i \in R} Y_{ijn} - \sum_{k \in R} Y_{jkn} = 0 \quad \forall j \in R, \forall n \in N \quad (5)$$

Similar to the idea of modeling techniques in VRP problems (Dror, 1994), we derive flow balance constraints from characterizing different types of nodes. Constraint (3) ensures that each Co-bus departs at the predefined origin node. Constraint (4) ensures that the Co-bus bus line tasks may eventually terminate at their destination. Flow conservation conditions for the rest nodes are defined by constraint (5), resulting in equal flow between the input and output of each node.

b. *Rationality statement.*

To guarantee the rationality of this model, we need to specify the following relation:

$$\sum_{i \in R_p} X_{itn} \leq 1 \quad \forall t \in T, \forall n \in N, \forall p \in P \quad (6)$$

Constraint (6) guarantees that each Co-bus will not operate more than one task in a period. Moreover, the decision variables should be correlated through the corresponding constraint.

$$\sum_{t \in T} X_{itn} \leq M \cdot Z_{in} \quad \forall i \in R, \forall n \in N \quad (7)$$

$$\sum_{t \in T} X_{itn} \leq Z_{in} \quad \forall i \in R, \forall n \in N \quad (8)$$

Constraints (7) and (8) indicate the relationship between X_{itn} and Z_{in} . Constraint (7) ensures that if a task i is not assigned to Co-bus n , the operating time for Co-bus n is zero. That is to say, Co-bus does not operate the line task that is not selected. Since X_{itn} can be utilized to record the operating time of the Co-bus, constraint (8) can also be interpreted to mean that the operation time is at least 1 if the bus line task is selected.

$$\sum_{i \in R} Y_{ijn} \leq Z_{jn} \quad \forall j \in R, \forall n \in N \quad (9)$$

Constraint (9) means that the bus line task is not operated if it is not selected.

c. *Time continuity.*

Co-bus needs to run continuously until the ending time of this task when task i is being started. For better instructions, we use constraint (11) to address the above requirement, under condition (10), which identifies the start time of the task i .

If:

$$X_{itn} - X_{i(t-1)n} = 1 \quad \forall n \in N, \forall i \in R, \forall t \in T \quad (10)$$

$$\sum_{t=t_0}^{t_0+b_i} X_{itn} = b_i \quad \forall n \in N \quad (11)$$

Then and;

$$Y_{ijn} = 1 \quad \forall j \in R \quad (12)$$

$$\sum_{i \in R_p} \sum_{t=t_0+b_i}^{t_0+b_i+B_{ij}+\gamma} X_{itn} = 0 \quad \forall n \in N \quad (13)$$

$$X_{j(t_0+b_i+B_{ij}+\gamma+1)n} = 1 \quad \forall n \in N \quad (14)$$

Further, the characteristics when the task j is operated after task i should also be addressed. Condition (12) indicates that the successive task of i is j . If the task j will be operated after task $i \in R$, the transfer process will occur. Note that it should not be regarded as an extension of task $i \in R$. That is to say, during the transfer process and rest time, the Co-bus runs with no task, imposed by constraint (13), and task j will start in the next period, as stated in constraint (14). Constraints (6)–(14) can guarantee that the subtour is eliminated.

In summary, we formulate the problem as mixed-integer linear programming through the above definition and transformation.

5 Solution strategy

Regarding the above formulation, Eqs. (10)–(14) are nonlinear constraints, leading to a significant computation burden. To cope with this issue, linearization techniques are applied concerning the structural properties of each scenario. In particular, we take the specific value of the variables under the proposed condition to perform the linearization process and simplify the model calculation. Consequently, we obtain Eqs. (15)–(17).

$$\sum_{t=t_0}^{t_0+b_i} X_{itn} - b_i \geq -M \cdot (1 - (X_{itn} - X_{i(t-1)n})) \quad \forall n \in N, \forall i \in R, \forall t \in T \quad (15)$$

$$\sum_{t=t_0+b_i}^{t_0+b_i+B_{ij}+\gamma} X_{itn} \leq M \cdot (2 - (X_{itn} - X_{i(t-1)n} + Y_{ijn})) \quad \forall n \in N, \forall i, j \in R, \forall t \in T \quad (16)$$

$$X_{j(t_0+b_i+B_{ij}+\gamma+1)n} \geq 1 - (2 - (X_{itn} - X_{i(t-1)n} + Y_{ijn})) \quad \forall n \in N, \forall i, j \in R, \forall t \in T \quad (17)$$

In these equations, the big- M method is used to support the linearization process, and M should be sufficiently large. Specifically, for Eq. (15), M should be greater than the upper bound of $\sum_{t=t_0}^{t_0+b_i} X_{itn}$, or b_i since X_{itn} are all binary variables. For Eq. (16), M should be greater than the upper bound of $\sum_{t=t_0+b_i}^{t_0+b_i+B_{ij}+\gamma} X_{itn}$, or $B_{ij} + \gamma$.

After Eqs. (10)–(14) are linearized as Eqs. (15)–(17), the original nonlinear model can be fed into state-of-the-art solvers for optimization.

6 Numerical experiment

In this section, we test the performance of the proposed scheme and validate its superiority. Section 6.1 shows that the proposed deep learning algorithm produces accurate prediction results for travel demand. In Sect. 6.2, we provide a detailed numerical experiment of the Co-bus, then compare the performance of the proposed scheme to some frequently-used schemes. Finally, we analyze the sensitivity for different uncertainties in Sect. 6.3. All numerical experiments are implemented using Python and the Gurobi solver v9.1 on a 2.90 GHz AMD Ryzen 7 4800H PC with 16 GB RAM.

The data set was collected by bus card readers and cameras from 159 bus lines in Shenzhen, a metropolis in China, between January 1 and January 31, 2021. Each sample contains detailed travel information, including the ID of devices (anonymized), passengers' boarding time, bus station ID, address of bus station, and bus line ID. This fine-grained data set guarantees the credibility of our travel demand analysis and modeling.

6.1 Travel demand prediction

First, the K-means algorithm is operated on all min–max normalized data, and the Euclidean distance measures the distance between vectors. The optimal number of clusters $K = 4$ is determined by the elbow method (Bholowalia & Kumar, 2014).

We then improve the performance of GAN prediction using metric learning. The proposed metric learning algorithm parameters are shown in Table 4.

We use the silhouette coefficient, the maximum value of the mean $s(i)$ overall data of the entire data set, to measure the quality of clustering. A higher silhouette coefficient indicates that a sample is well matched to its cluster and poorly matched to its neighboring clusters.

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (18)$$

where $a(i)$ can be interpreted as a measure of how well i is assigned to its cluster (the smaller the value, the better the assignment), and $b(i)$ can be interpreted as the smallest distance of i to all points in any other cluster.

The performance of clustering based on our method is compared with the predefined weights by experience. Specifically, the ratio of the predefined weight in peak periods (7–10 AM and 6–9 PM) and other periods is set to 2:1. At the same time, we test the influence of different rank thresholds on the algorithm. The performance is shown in Fig. 8. The proposed approach outperforms the experience-based approach, since the silhouette coefficient is improved under a rank threshold of 0.5 from 0.3125 to 0.3722.

As illustrated in Fig. 9, we visualize the clustering results on distribution patterns under the selected rank threshold of 0.5. The solid line is the center of each cluster, and the shaded parts are used to outline figures to show each cluster's boundaries (the upper and lower bounds).

In Fig. 10, we visualize the weights for each feature under the selected rank threshold of 0.5. These weights can be further employed in identifying the key traffic patterns effectively.

For a GAN, the two neural networks—discriminator D and generator G—could be formulated in any type (Yu et al., 2019a). In this study, to capture the profile of distribution, a CNN is used as the discriminator D. A fully connected neural network is used as the generator G. The CNN consists of four convolutional layers with 1, 16, 32, and 8 channels, three

Table 4 Parameters of metric learning algorithm

Parameter	Value
The number of clusters K	4
Learning rate	0.1
Decay rate	0.95
Iterations	100
Rank threshold	0.4/0.5/0.6
Accuracy threshold	0.9

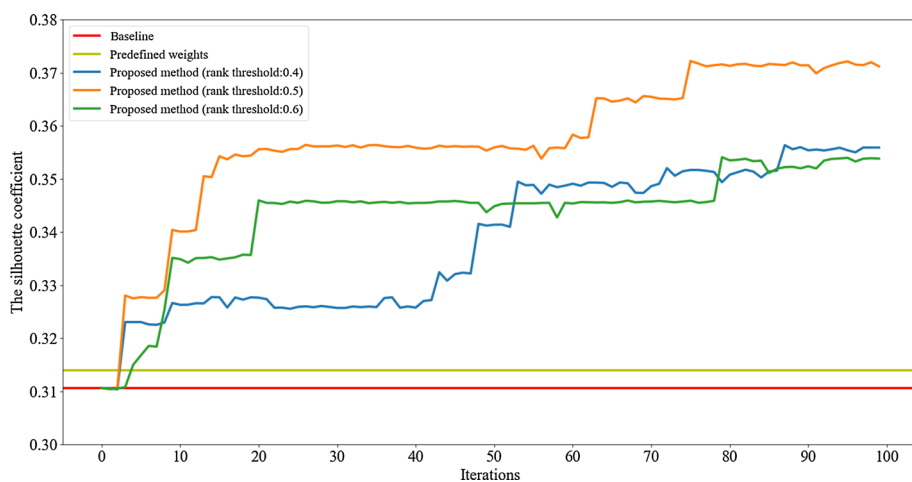


Fig. 8 The performance of silhouette coefficient by metric learning

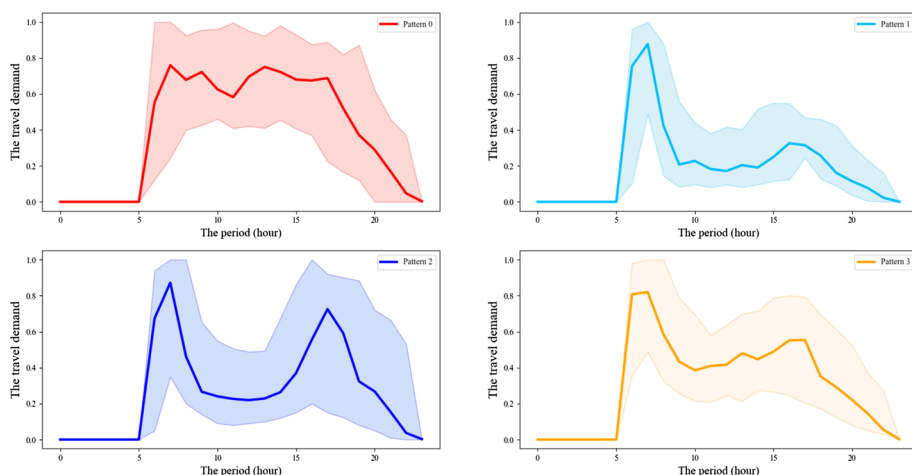


Fig. 9 The travel demand distribution for each cluster

pooling layers with 3 kernel sizes, and four fully connected layers with 96, 32, 8 and 1 neurons respectively. The activation functions of every middle layer are ReLU, while the final layer connects with a sigmoid activation function. A batch-normalization layer (Saxena & Cao, 2019) and a dropout layer are added after each convolutional layer to avoid overfitting. Generator G has five fully connected layers with 6, 128, 64, 32, and 24 neurons respectively. The activation function of every middle layer is ReLU, while the final layer connects with a sigmoid activation function, and trains GAN separately for each type of travel demand distribution.

The performance of GAN based on metric learning is shown in Fig. 11, exhibiting better fitting results than that of GAN without metric learning. We then compare the performance of

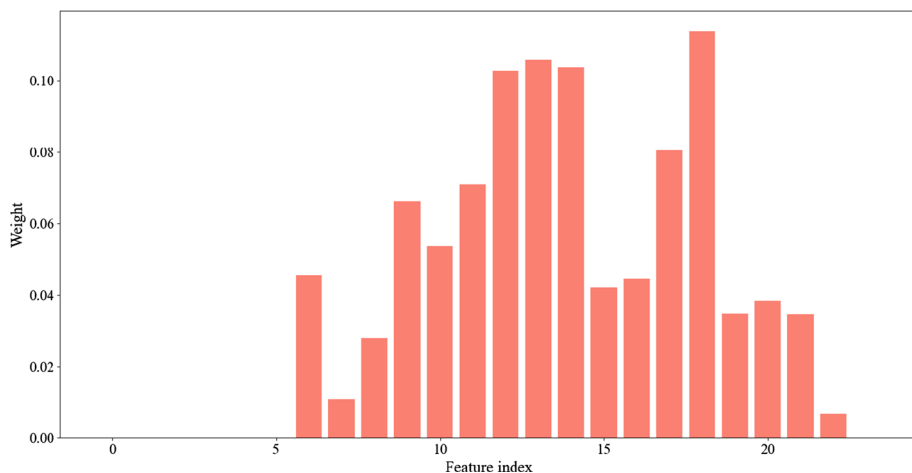


Fig. 10 The weight for each feature

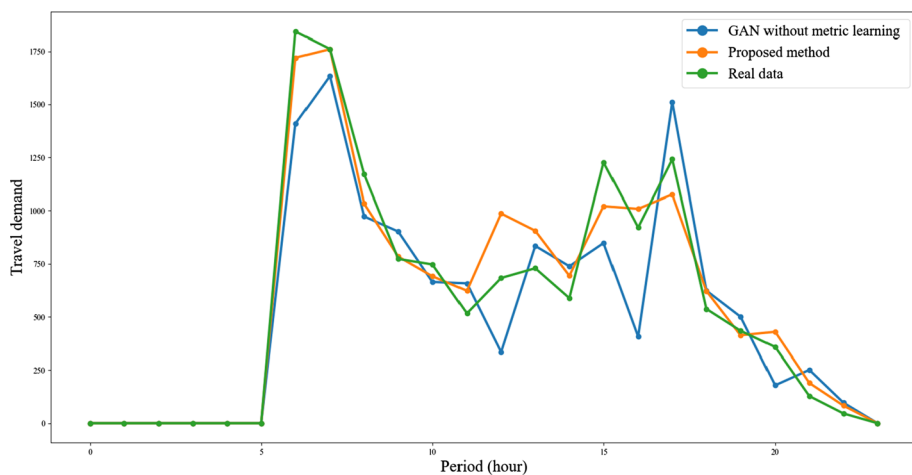


Fig. 11 The performance of the proposed GAN with the metric learning method

the original method with the proposed method, including the mean absolute error and mean absolute percentage error, as shown in Table 5.

In conclusion, the performance of GAN is greatly boosted by metric learning. These results can be further fed to the route planning model.

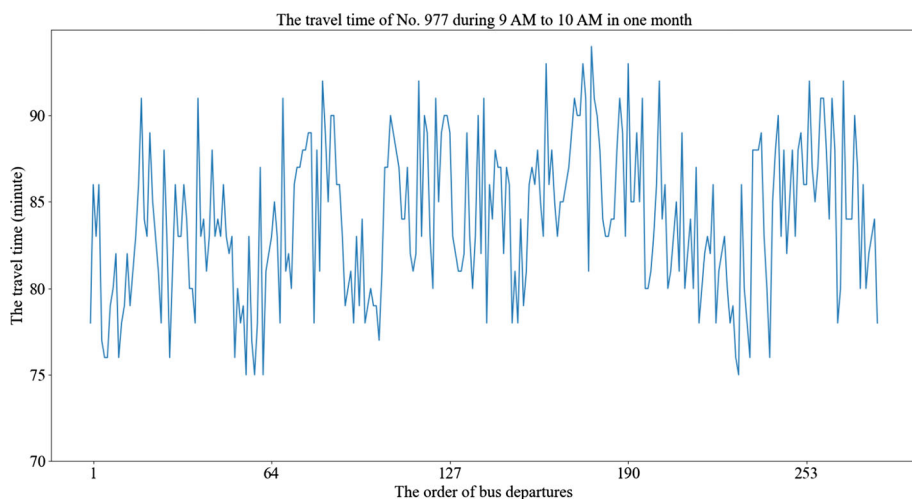
6.2 Route planning case study

We consider only 5 bus lines and 10 Co-buses for route planning to simplify the demonstration. Module information and model inputs are given as follows.

Travel demand parameters can be derived from Sect. 6.1. The travel time b_i , is set to the mean of historical travel time in each period for all tasks, which we will justify using the

Table 5 The result of the proposed method

	MAPE	MAE
Proposed method	14.285	73.06
GAN without metric learning	24.605	141.109

**Fig. 12** The travel time in the period**Table 6** The result of ADF

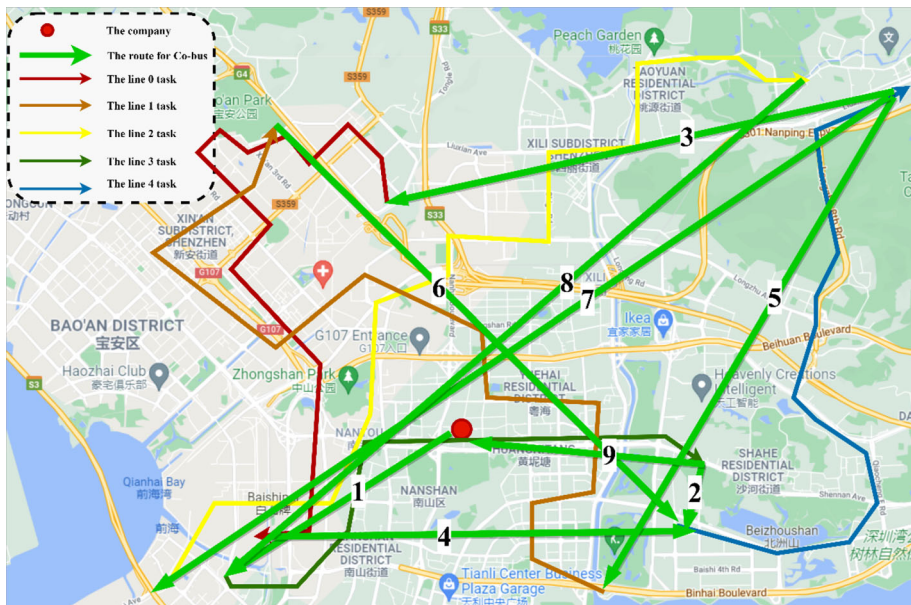
Index	Reference value	Test value
10% rejection region	−3.454355	−4.98979
5% rejection region	−2.872108	−4.98979
1% rejection region	−2.572401	−4.98979
<i>P</i> -value		2.32e−05

ADF test. We take the M176 bus running time from 9 to 10 AM over a month as an example for ADF analysis, as shown in Fig. 12, and the result is exhibited in Table 6.

The test results are all less than the reference values of the boundaries of 1, 5, and 10%, and the *p*-value is very close to 0. Therefore, we believe that the stationarity of the series is significant. The travel time of each line in each period is confirmed to be stable by the ADF test with a small variance. Hence, the average is desirable based on a large amount of historical data.

Since it is difficult to collect the actual transfer time data for Co-buses, we generate them by calculating the ratio of the between-task distance interval to the average travel speed. Note that this generation technique may affect the performance because the travel time in peak periods is longer than average in general. This effect will be discussed in detail later through sensitivity analysis.

Parameter	Value
The number of bus lines	5
The number of Co-buses	10
The number of periods	6
The number of timeslots	12
Rest time	10 min
The company for 10 Co-buses respectively	[0, 0, 1, 1, 2, 2, 3, 3, 4, 4]
The start time for 10 Co-buses respectively	[9:10, 9:20, 9:30, 9:20, 9:00, 9:10, 9:20, 9:10, 9:20, 9:30]
The end time for 10 Co-buses respectively	[20:00, 20:20, 19:30, 19:50, 20:40, 19:20, 19:10, 20:00, 20:50, 21:00]



The number of buses in each period of the original line is set to the number of departures in each period in historical data. Other parameters for the model are listed in Table 7, including (a) the number of bus lines, Co-buses, periods, and timeslots, (b) rest time, and (c) the company, the start time and the end time for each Co-bus.

 Springer



Fig. 14 Diagram of the timetable of all Co-buses

Table 8 The optimal route for each Co-bus

Co-Bus No	Route
Co-Bus 0	Company 0, line 2, line 4, line 2, line 4, line 1, line 0, line 2, Company 0
Co-Bus 1	Company 0, line 4, line 4, line 1, line 0, line 4, line 0, line 2, Company 0
Co-Bus 2	Company 1, line 2, line 3, line 1, line 0, line 4, line 0, line 1, Company 1
Co-Bus 3	Company 1, line 2, line 4, line 3, line 4, line 1, line 2, line 0, Company 1
Co-Bus 4	Company 2, line 3, line 4, line 0, line 4, line 1, line 4, line 2, line 3, Company 2
Co-Bus 5	Company 2, line 3, line 4, line 2, line 4, line 4, line 2, line 0, Company 2
Co-Bus 6	Company 3, line 2, line 4, line 3, line 4, line 1, line 0, line 2, Company 3
Co-Bus 7	Company 3, line 2, line 4, line 2, line 4, line 4, line 2, line 0, Company 3
Co-Bus 8	Company 4, line 1, line 4, line 1, line 0, line 4, line 2, line 0, Company 4
Co-Bus 9	Company 4, line 0, line 0, line 4, line 4, line 1, line 4, line 2, line 3, Company 4

We then compare the optimization results under three different schemes: (a) Co-bus with the proposed scheme, (b) Co-bus with the nearby-service scheme, which means the Co-bus only assists on the nearest ordinary bus line to the company, and (c) without Co-bus. Specifically, we compare their service level, average Co-bus utilization rate (in (a) and (b)), and the maximum difference in Co-bus utilization rate (in (a) and (b)), as shown in Fig. 15. Note that the utilization rate is defined as the ratio of the travel time on all bus line tasks to the total idle time. The proposed scheme (a) outperforms the scheme without Co-buses (c) by 56.2% and the nearby-service scheme (b) by 25.1% in service level. At the same time, the mean of the Co-bus utilization rate is also improved by 30.1% from the nearby-service scheme

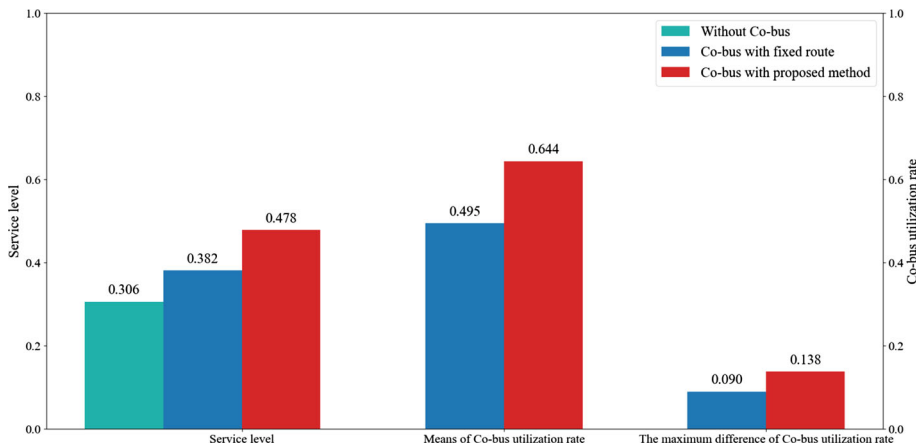


Fig. 15 The performance of the proposed scheme compared with other schemes

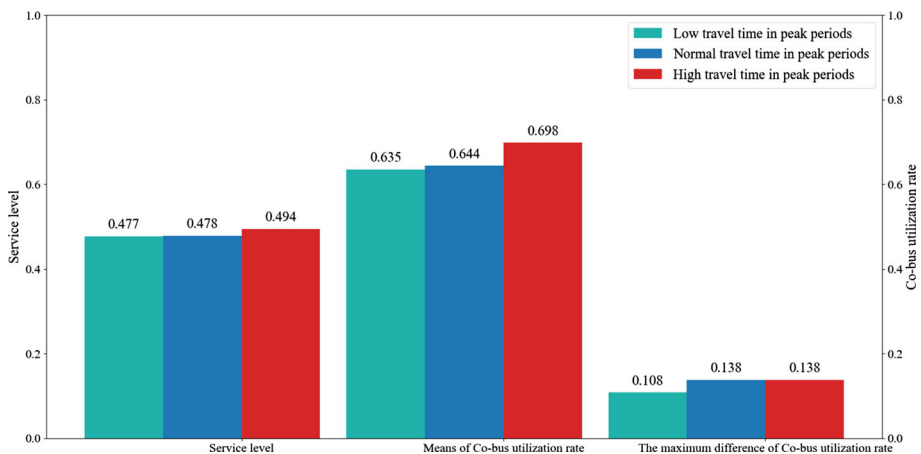


Fig. 16 The sensitivity analysis on travel time in the peak periods

(b). The utilization rate among all 10 Co-buses is balanced, with a maximum difference of 13.8% in this scheme.

6.3 Sensitivity analysis

Traffic environments in peak times are quite complicated, we take the morning peak in Shenzhen as an example for sensitivity analysis, as the morning peak period is more concentrated while the evening peak period in Shenzhen is much longer. More specifically, we perform sensitivity analysis on the influence of different parameters in the morning peak period (8 AM to 10 AM) such as travel time, transfer time, and the number of departures on ordinary lines.

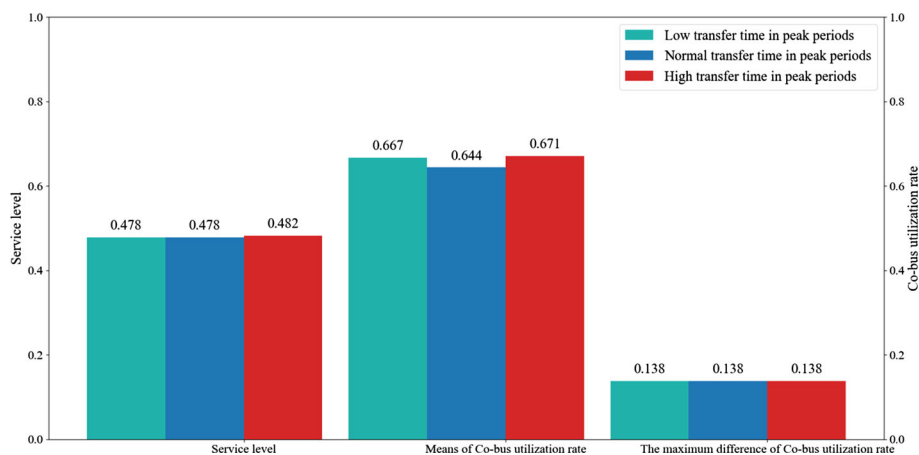


Fig. 17 The sensitivity analysis for transfer time in the peak periods

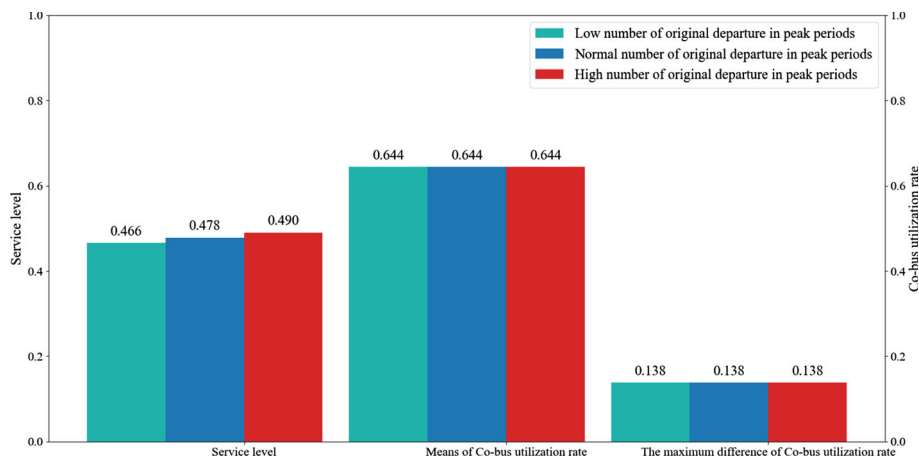


Fig. 18 The sensitivity analysis for the number of departures on ordinary line in the peak periods

First, we compare the situation in which the model's travel time in morning peak times increases or decreases by 20%. The results are shown in Fig. 16. It is found that the additional travel time in a short period does not damage the performance of the final result; on the contrary, the service level actually increases from 0.478 to 0.494 when the travel time increases by 20%. Empirical observations can explain that Co-buses will adjust their long-term decisions toward a more optimal service level when facing a current disadvantage in short-term travel time. Meanwhile, the transportation system can still maintain a balance among the utilization rates of different Co-buses.

We then compare the situation in which the transfer time in morning peak times increases or decreases by 20% on the model. The results are illustrated in Fig. 17. It is found that the variation of transfer time in a short time has little influence on the final result of the model, including the services level and Co-bus utilization. This can be explained by the fact that

the object of our study is a region in a city, which means that the bus lines are relatively concentrated, and the transfer process does not need to stop and pick up passengers along the bus line. Therefore, the overall influence is slight regardless of the variation in transfer time.

Finally, we compare the situation in which the number of departures on ordinary bus lines in morning peak periods increases or decreases by 20% on the model. The results are illustrated in Fig. 18. It suggests that the change will have little impact on the route planning of Co-buses. The service level will increase as the number of ordinary buses increases simultaneously, which indicates that the operation of the Co-bus has strong robustness and will not fluctuate significantly due to the fine-tuning of bus companies to the original bus line departure strategy.

Briefly, we have demonstrated the superiority and robustness of our proposed Co-bus scheme, and that the route planning model can improve the service level and utilization efficiency of Co-buses. Therefore, our proposed scheme (Co-bus) can provide effective solutions for urban bus management.

7 Conclusion

In this paper, we propose a new scheme “Co-bus” for urban bus transportation, which considers that cooperation with ordinary buses to improve the service level and reduce the gap between travel demand and capacity. To support our decision model, we use GAN with traffic scenario segmentation by metric learning, which can learn the weights of features adaptively to facilitate predictive performance. Based on the actual bus data, we conduct extensive experiments to verify that our scheme can generate effective routes to optimize the operation of Co-buses. The experiment results show that our planning routes can significantly improve the service level and Co-bus utilization rate. Our proposed Co-bus scheme effectively alleviates the imbalance between urban demand and capacity.

In addition, some managerial implications are obtained through our study. First, public transport operators are supposed to consider expanding the supply of Co-bus, to promote a more flexible new norm with the idea of sharing paradigm, so that the passenger transport resources are more likely to adapt to the changing passenger demand. Second, insightful strategic and operational decision-making can be facilitated by not only an accurate prediction model but also proper processing skills to enhance the data quality. This will completely convey the value of data and enable better prediction performance. To realize that, we need to reduce the overfitting risk and better reflect the ground truth, by including but not limited to carrying out rational segmentation and exploiting the synergy effect by ensemble learning, which also implies the key idea of “data-driven”.

Finally, we list some extensions to this work. (1) The integration of both the Co-bus and other available transport resources, such as minibuses and private vehicles, in the road transportation system. Through using these additional resources, the joint optimization of the overall resources of the entire city can be studied, combined with the prediction of time-dependent passenger demand. (2) Improved prediction algorithm to enhance the model’s ability to capture and recognize wave patterns through representation learning and automatic feature engineering. (3) Better acceleration strategies for the MIP, e.g., some cutting planes can be customized according to the scenario characteristics and optimal structures, in order to minimize the feasible domain and accelerate the solution.

Acknowledgements The authors Jiguang Wang and Yilun Zhang contribute to this work equally. This research is funded by the Shenzhen Science and Technology Innovation Commission (Grant No.

JCYJ20210324135011030), the National Natural Science Foundation of China (Grant No. 71971127), the Guangdong Pearl River Plan (2019QN01X890), and the ECR fund of the University of Liverpool.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ai, Y., Li, Z., Gan, M., Zhang, Y., Yu, D., Chen, W., & Ju, Y. (2019). A deep learning approach on short-term spatiotemporal distribution forecasting of dockless bike-sharing system. *Neural Computing and Applications*, 31(5), 1665–1677.
- Aranganayagi, S., & Thangavel, K. (2007). Clustering categorical data using silhouette coefficient as a relocating measure. In *International conference on computational intelligence and multimedia applications (ICCIMA 2007)*.
- Banerjee, N., Morton, A., & Akartunalı, K. (2020). Passenger demand forecasting in scheduled transportation. *European Journal of Operational Research*, 286(3), 797–810.
- Bholowalia, P., & Kumar, A. (2014). EBK-means: A clustering technique based on elbow method and k-means in WSN. *International Journal of Computer Applications*, 105(9).
- Borndörfer, R., Grötschel, M., & Pfetsch, M. E. (2007). A column-generation approach to line planning in public transport. *Transportation Science*, 41(1), 123–132.
- Bouton, S., Knupfer, S. M., Mihov, I., & Swartz, S. (2015). *Urban mobility at a tipping point*.**
- Chen, J., Lim, C. P., Tan, K. H., Govindan, K., & Kumar, A. (2021). Artificial intelligence-based human-centric decision support framework: an application to predictive maintenance in asset management under pandemic environments. *Annals of Operations Research*, pp 1–24.
- Chen, X., He, Z., & Sun, L. (2019). A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation. *Transportation Research Part c: Emerging Technologies*, 98, 73–84.
- Chen, X., Yu, L., Zhang, Y., & Guo, J. (2009). Analyzing urban bus service reliability at the stop, route, and network levels. *Transportation Research Part a: Policy and Practice*, 43(8), 722–734.
- Choi, J., Lee, C., & Park, S. (2018). Dantzig-Wolfe decomposition approach to the vehicle assignment problem with demand uncertainty in a hybrid hub-and-spoke network. *Annals of Operations Research*, 264(1), 57–87.
- Cohen, B., & Kietzmann, J. (2014). Ride on! Mobility business models for the sharing economy. *Organization & Environment*, 27(3), 279–296.
- Cordeau, J.-F., & Laporte, G. (2007). The dial-a-ride problem: Models and algorithms. *Annals of Operations Research*, 153(1), 29–46.
- Cyprich, O., Konečný, V., & Kilianova, K. (2013). Short-term passenger demand forecasting using univariate time series theory. *PROMET-Traffic Transportation*, 25(6), 533–541.
- Davis, N., Raina, G., & Jagannathan, K. (2016). A multi-level clustering approach for forecasting taxi travel demand. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*.
- Demiriz, A., Bennett, K. P., & Embrechts, M. J. (1999). Semi-supervised clustering using genetic algorithms. *Artificial neural networks in engineering*, pp 809–814.
- Dror, M. (1994). Note on the complexity of the shortest path models for column generation in VRPTW. *Operations Research*, 42(5), 977–978.
- Eachempati, P., Srivastava, P. R., Kumar, A., Tan, K. H., & Gupta, S. (2021). Validating the impact of accounting disclosures on stock market: A deep neural network approach. *Technological Forecasting and Social Change*, 170, 120903.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.
- Gordon, J., & Hernández-Lobato, J. M. (2020). Combining deep generative and discriminative models for Bayesian semi-supervised learning. *Pattern Recognition*, 100, 107156.
- Jiang, X., Pang, Y., Pan, J., & Li, X. (2015). Flexible sliding windows with adaptive pixel strides. *Signal Processing*, 110, 37–45.

- Ke, J., Zheng, H., Yang, H., & Chen, X. M. (2017). Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach. *Transportation Research Part C: Emerging Technologies*, 85, 591–608.
- Kong, X., Li, M., Tang, T., Tian, K., Moreira-Matias, L., & Xia, F. (2018). Shared subway shuttle bus route planning based on transport data analytics. *IEEE Transactions on Automation Science and Engineering*, 15(4), 1507–1520.
- Kumar, A., Shankar, R., Choudhary, A., & Thakur, L. S. (2016). A big data MapReduce framework for fault diagnosis in cloud-based manufacturing. *International Journal of Production Research*, 54(23), 7060–7073.
- Kumar, A., Shankar, R., & Thakur, L. S. (2018). A big data driven sustainable manufacturing framework for condition-based maintenance prediction. *Journal of Computational Science*, 27, 428–439.
- Kumar, A., Gopal, R. D., Shankar, R., & Tan, K. H. (2022). Fraudulent review detection model focusing on emotional expressions and explicit aspects: investigating the potential of feature engineering. *Decision Support Systems*, 113728.
- Laporte, G., Marín, Á., Mesa, J. A., & Ortega, F. A. (2007). An integrated methodology for the rapid transit network design problem. In *Algorithmic methods for railway optimization* (pp. 187–199). Springer.
- Liang, X. Y., Du, X. S., Wang, G. L., & Han, Z. (2019). A Deep Reinforcement Learning Network for Traffic Light Cycle Control. *Ieee Transactions on Vehicular Technology*, 68(2), 1243–1253. <Go to ISI>://WOS:000458803200017
- Liang, Y., Cui, Z., Tian, Y., Chen, H., & Wang, Y. (2018). A deep generative adversarial architecture for network-wide spatial-temporal traffic-state estimation. *Transportation Research Record*, 2672(45), 87–105.
- Liu, F., Niu, B., Xing, M., Wu, L., & Feng, Y. (2021). Optimal cross-trained worker assignment for a hybrid seru production system to minimize makespan and workload imbalance. *Computers & Industrial Engineering*, 160, 107552.
- Lyu, Y., Chow, C. Y., Lee, V. C. S., Ng, J. K. Y., Li, Y. H., & Zeng, J. (2019). CB-Planner: A bus line planning framework for customized bus systems. *Transportation Research Part C-Emerging Technologies*, 101, 233–253. <Go to ISI>://WOS:000466060900014
- Marín, A., & Codina, E. (2008). Network design: Taxi planning. *Annals of Operations Research*, 157(1), 135–151.
- Mathew, N., Smith, S. L., & Waslander, S. L. (2015). Planning paths for package delivery in heterogeneous multirobot teams. *IEEE Transactions on Automation Science Engineering*, 12(4), 1298–1308.
- Naji, H. A., Xue, Q., Zhu, H., & Li, T. (2021). Forecasting taxi demand using generative adversarial networks with multi-source data. *Applied Sciences*, 11(20), 9675.
- Ning, Z., Sun, S., Zhou, M., Hu, X., Wang, X., Guo, L., Kwok, R. Y. (2021). Online Scheduling and Route Planning for Shared Buses in Urban Traffic Networks. *IEEE Transactions on Intelligent Transportation Systems*.
- Ning, Z., Xia, F., Ullah, N., Kong, X., & Hu, X. (2017). Vehicular social networks: Enabling smart mobility. *IEEE Communications Magazine*, 55(5), 16–55.
- Pillac, V., Gendreau, M., Guéret, C., & Medaglia, A. L. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1), 1–11.
- Saxena, D., & Cao, J. (2019). D-gan: Deep generative adversarial nets for spatio-temporal prediction. *arXiv preprint arXiv:1908.08556*
- Tu, Z. (2007). Learning generative models via discriminative approaches. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*.
- Wang, Y., Chen, X., Han, Y., & Guo, S. (2013). Forecast of passenger and freight traffic volume based on elasticity coefficient method and grey model. *Procedia-Social and Behavioral Sciences*, 96, 136–147.
- Wang, Y., Zhang, D. X., Hu, L., Yang, Y., & Lee, L. H. (2017). A Data-Driven and Optimal Bus Scheduling Model With Time-Dependent Traffic and Demand. *Ieee Transactions on Intelligent Transportation Systems*, 18(9), 2443–2452. <Go to ISI>://WOS:000408755500015
- Xu, F., Li, Y., Wang, H., Zhang, P., & Jin, D. (2016). Understanding mobile traffic patterns of large scale cellular towers in urban environment. *IEEE/ACM Transactions on Networking*, 25(2), 1147–1161.
- Yu, H., Chen, X., Li, Z., Zhang, G., Liu, P., Yang, J., & Yang, Y. (2019). Taxi-based mobility demand formulation and prediction using conditional generative adversarial network-driven learning approaches. *IEEE Transactions on Intelligent Transportation Systems*, 20(10), 3888–3899.
- Zhang, Y., Guo, H., Li, C., Wang, W., Jiang, X., & Liu, Y. (2016). Which one is more attractive to traveler, taxi or tailored taxi? An empirical study in China. *Procedia Engineering*, 137, 867–875.