Information Processing and Management 61 (2024) 103733

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Combining travel behavior in metro passenger flow prediction: A smart explainable Stacking-Catboost algorithm

Jiarui Yu [a], Ximing Chang [a,*], Songhua Hu [b,*], Haodong Yin [a], Jianjun Wu [a]

[a] School of Systems Science, Beijing Jiaotong University, Beijing, China
[b] School of Automotive and Transportation Engineering, Shenzhen Polytechnic, Shenzhen, China

ARTICLE INFO

Keywords:
Passenger flow prediction
Explainable machine learning
Ensemble learning
Smart card data
Individual travel behavior

ABSTRACT

Accurately predicting short-term passenger flow is essential to optimize operation resources and improve transportation services in urban metro systems. However, it has become a challenging problem due to spatial-temporal demand fluctuations and heterogeneous passengers' travel behavior, i.e., the interaction between the departure and arrival passengers. In this paper, we develop an explainable Stacking-Catboost model for passenger flow prediction combining the passenger's return probability computation. The model explores several basic ensemble learning models and the best stacking strategy. To better characterize the macroscopic spatiotemporal travel patterns other than the micro individual travel behavior, several relevant variables such as train operation characteristics, the nearby bus stations, and points of interest are considered. Ablation studies are conducted to investigate the utility of each component of the proposed model. An explainable analysis is performed to interpret information from "black box" models and quantify the contribution of each feature. We present a real-world case study conducted in the Beijing metro, demonstrating that our proposed method achieves significant improvements over existing techniques in hourly prediction tasks. Specifically, our approach outperforms the CategoricalBoosting (CatBoost) by up to 11.11 % and the Random Forest (RF) by up to 12.90 %, showcasing the effectiveness of incorporating macroscopic spatiotemporal travel patterns and micro individual travel behavior to enhance short-term forecasting accuracy. The global interpretability of models reveals key factors that impact performance, such as returning passenger flow, historical travel demand, and timetable data. Our results offer valuable insights into short-term forecasting challenges and provide a framework for leveraging explainable artificial intelligence to improve transportation services in urban metro systems.

1. Introduction

Over the years, rapid population growth, increasing urbanization, and a surge in private vehicle usage have led to severe road traffic congestion. To alleviate this problem, many cities have developed metro systems that play a crucial role in daily passenger travel. At the end of 2021, 283 metro lines operated in 50 Chinese cities, with an annual passenger volume exceeding 23.69 billion, accounting for 43.4 % of public transport passengers (China Association of Metros, 2022). As the metro network continues to expand, passenger flow also significantly increases. Therefore, real-time monitoring and forecasting of passenger flow have become

* Corresponding authors.
E-mail addresses: ximingchang@bjtu.edu.cn (X. Chang), husonghua@szpt.edu.cn (S. Hu).

https://doi.org/10.1016/j.ipm.2024.103733
Received 16 May 2023; Received in revised form 25 October 2023; Accepted 28 March 2024
Available online 18 April 2024

increasingly vital. These measures help develop reasonable train operating plans and provide safety warnings in advance to ensure the efficient and safe operation of the metro system (Zhang et al., 2020).

Previous studies have often aggregated smart card transactions into time series data for extracting temporal travel demand distributions. However, this approach ignores the individual travel behavior characteristics that exhibit strong regularity (Chang et al., 2024). As a form of public transit, the metro typically accommodates daily commuter traffic, making it crucial to illustrate the commuting regularity for predicting metro passenger flow. For example, if the passenger alights at a metro station in the morning to work, there is a high probability that he/she will board at the same metro station in the evening to return home from work. On the other hand, if the passenger does not travel in the morning, it is unlikely to find relevant information about the return trip. This demonstrates the significance of incorporating previous individual travel behavior in future individual travel demand prediction. Recent studies have highlighted the importance of travel behavior in dynamic traffic assignment (Cantelmo & Viti, 2019; Cantelmo et al., 2020; Sun et al., 2013). Thus, it becomes necessary to incorporate the regularity of passengers' travel behavior for short-term passenger flow prediction in the metro system. By considering individual travel behavior in prediction models, we can make more accurate predictions, which can aid in developing effective strategies to manage passenger flow and ensure efficient operations of the metro system.

The metro system operates with unique characteristics that distinguish it from other forms of public transport. One key difference is that the metro operation follows a more precise timetable, which must be considered when predicting passenger flow (Liu et al., 2019). Passengers tend to prefer waiting at the station for the arrival of the train, leading to a small peak in boarding passenger flow at that time. Operators aim to match passenger flow with train capacity as closely as possible. Besides, commuters account for a significant portion of daily metro trips, and most of them come from the transfer of bus stations. More bus stops nearby may generate a larger transfer passenger flow because buses are often used to connect with the metro system. The land use around the metro station plays a role, as different regions have varying travel patterns. Points of interest (POI) provide insight into the functional characteristics of the area, such as stations near residential buildings usually dominated by commute trips.

Machine learning algorithms have become increasingly popular for prediction tasks due to their ability to handle dynamic, high-volume, and complex data in various formats. In passenger flow forecasting, advanced methods like ensemble learning have emerged, which use bagging, boosting, and stacking techniques to enhance model performance. (1) Bagging involves homogenous weak learners' models that learn independently in parallel and combine results to determine the model average, reducing variance and avoiding overfitting. (2) Boosting is also a homogeneous weak learners' model but its learners learn sequentially and adaptively to improve model predictions of a learning algorithm, which can decrease the bias while maintaining a smaller variance. (3) Stacking is a distinct paradigm that explores space for different models for the same problem, synthesizing their strengths and compensating for weaknesses to produce more accurate predictions and avoid overfitting. Although the ensemble learning method has a good predictive effect, its black-box nature cannot express the learned features explicitly and cannot do an effective interpretation. Model explainability can not only explain the complete behavior of the model but also help to understand how the model makes decisions for single instances and explain the individual predictions (Zhang et al., 2022).

This paper aims to investigate the impact of individual travel behavior on the accuracy of passenger flow prediction in the metro system in various scenarios. Metro demand prediction refers to the process of forecasting passenger numbers within a short time period. This type of analysis is crucial for transit operators, as it enables them to optimize scheduling, allocate resources effectively, and avoid overcrowding or underutilization of services. By accurately predicting passenger demand, operators can improve the efficiency and reliability of their metro systems, ultimately leading to a better commuting experience for riders. This paper characterizes both the macroscopic spatiotemporal travel patterns and the micro individual travel behavior and proposes a new explainable algorithm called Stacking-Catboost. Model explainability helps to understand how the model makes decisions for specific instances as well as the model's overall behavior. The global interpretability of models reveals different feature importance levels. The main contributions of this study are as follows:

- In terms of extracting micro-level individual travel behavior, this paper introduces the concept of "returning passenger flow" and proposes the "return probability parallelogram" as a computational method to characterize this individual travel behavior. This method is used to estimate the future returning passenger flow more accurately.
- In order to better characterize the macroscopic spatiotemporal travel patterns other than the micro individual travel behavior, this paper extensively explores the temporal characteristics, spatial characteristics, and unique timetable features of historical passenger flow data. Machine learning algorithms can analyze the relationship between passenger flow and features, such as historical time-step passenger flow, the close-by bus stops, points-of-interest data, and train operation characteristics, learning the patterns and regularities hidden within the data.
- A Stacking-Catboost algorithm incorporating macroscopic spatiotemporal travel patterns and micro individual travel behavior for short-term metro passenger flow prediction is proposed to improve the accuracy. Besides, explainable artificial intelligence is applied to extract and interpret information from "black box" models and quantify the contribution of each feature, which provides the basis for customized rail transit.

The rest of this paper is organized as follows. Section 2 reviews the literature on short-term passenger flow prediction. Section 3 presents the concepts of returning passenger and return probability parallelograms, explains other relevant variables, and then introduces the proposed models and stacking strategy. Section 4 applies the proposed method to the real Beijing metro dataset. Finally, the conclusion and future work are presented in Section 5.

2. Literature review

The passenger flow prediction problem can be divided into three categories based on the forecast range, including long-term, medium-term, and short-term predictions. Long-term predictions typically have a range of 1 to 30 years, whereas medium-term predictions predict up to 12 months into the future. The short-term predictions focus on travel demand variations in the near future, typically within a few hours (Li et al., 2018; Lei et al., 2022). Short-term passenger flow forecasting is particularly critical for real-time metro system operations, as it plays a vital role in improving resource utilization efficiency and service levels. With the advent of new technologies, the development of short-term passenger flow prediction algorithms has accelerated significantly and has garnered considerable attention from researchers (Bao et al., 2022; Huang et, al.,2023).

Previous studies have extensively explored the factors that affect the accuracy of passenger flow prediction, including temporal, spatial, and external factors. Holidays, emergencies, station locations, and weather also significantly impact metro passenger flow forecasts (Liu et al., 2019). Feature selection is essential for accurate short-term passenger flow predictions. Wen et al. (2022) considered holiday characteristics based on the analysis of passenger travel habits, and the results suggested that tagging the dates with holidays could improve prediction accuracy. Xue et al. (2022) used social media data to extract socioeconomic features of metro stations to forecast passenger flow during events, resulting in better-identified passenger flow patterns. By adding the socioeconomic characteristics, the passenger flow regularity could be better mined. Arana et al. (2014) found that rainfall and low temperatures were negatively correlated with the number of passengers traveling for leisure on weekends. Many studies also focus on extracting spatial features to predict metro passenger flow because neighboring stations or stations with similar functions have similar passenger flow travel structures that can influence prediction accuracy. Ma et al. (2019) developed a novel model that transformed ridership into images based on the cell-based method, retaining the topological structure between stations to reflect the spatial relationship of metro stations. This model outperformed traditional statistical models and sequential structures and was suitable for predicting ridership in large-scale metro networks. Grasping the characteristics of metro train operation is conducive to strengthening the forecasting performance because passengers will select the arrival time according to the timetable. Liu et al. (2019) proposed a fully data-driven approach to replace the timetable with the peak of passenger flow from raw data.

Metro passenger flow forecasting models have traditionally utilized time series statistical approaches, such as the historical average (HA), moving average model (MA), autoregressive difference moving average model (ARIMA), etc. ARIMA model, combining moving averages and linear differences, is used to predict time series data with a linearly smooth nature (Williams, 2001). Yang et al. (2021) used ARIMA to predict inbound passenger flow. To capture the seasonal and trend characteristics of passenger flow data, the seasonal ARIMA (SARIMA) is introduced.Milenković et al. (2018) proposed a model to forecast metro passenger flow using SARIMA. However, the application of the ARIMA or SARIMA model is limited because they assume a linear connection among time-lag variables. Besides, the ARIMA or SARIMA model is limited in its ability to capture the nonlinear characteristics of passenger flow, especially in shorter prediction time ranges (Chang et al., 2022). These statistical models, which focus on the fluctuation of historical trips, are not dominant in the mining of temporal and spatial characteristics.

To better capture the nonlinear characteristics of passenger flow, researchers have begun to use machine learning algorithms with the rise of intelligent computing (Chang et al., 2023). Shi et al. (2020) proposed a support vector regression (SVR) considering the periodic historical data. Results confirmed that the proposed model significantly improved predictive performance and generalization. Chang et al. (2012) developed a dynamic multi-interval technique based on k-nearest neighbors (KNN), which was a promising system-oriented approach for multi-interval traffic flow forecasting. Additionally, Chan et al. (2012) proposed a novel neural network (NN) training method that employed a hybrid exponential smoothing method, enhancing the generalization capabilities of NNs. Machine learning algorithms are also powerful in processing inadequate data. Roos et al. (2017) proposed a dynamic Bayesian network-based technique for short-term passenger flow prediction in the Paris urban rail network, which was effective even for inadequate data. Habtemichael and Cetin (2016) developed a non-parametric and data-driven short-term forecast system based on an improved KNN algorithm that better recognized similar travel patterns and fused them for more accurate predictions.

Combining traditional time series models with machine learning techniques is also a popular approach. Ming et al. (2014) combined ARIMA with SVM to predict air passenger traffic, and the hybrid model outperformed the single models alone. Currently, decomposing time series into subseries and forecasting each subsequence using a parametric or non-parametric model is a common combination method. The final result is obtained by adding or multiplying the forecasted subseries. Time series can be decomposed in various ways. Cheng et al. (2022) proposed a mixed model based on wavelet analysis to evaluate the historical passenger flow. Jiang et al. (2014) combined empirical modal decomposition (EMD) with a gray support vector machine model to predict railroad passenger flow. Wei and Chen (2012) combined EMD with a back propagation neural network to predict short-term metro ridership. Chan et al. (2012) employed the Levenberg-Marquardt method and a hybrid exponential smoothing method to train their neural network, which considerably enhanced its generalization ability.

The aforementioned researches focus more on historical time-step passenger flow data and external environmental factors like the weather when selecting features. Commuter trip accounts for a large proportion of metro travel. Therefore, short-term passenger flow prediction not only needs to effectively grasp the macroscopic travel patterns but also needs to integrate individual travel behavior. In addition, in terms of feature selection, temporal features, spatial features, and variables related to metro operation characteristics should be extracted, such as timetables, to improve passenger flow prediction accuracy.

3. Methodology

This section introduces the return probability parallelogram as a measurement to capture individual travel behavior, specifically in

relation to returning passenger flow. To further improve prediction accuracy, a Stacking-Catboost algorithm is integrated into a proposed model that takes into account returning passenger flow. Additionally, the model incorporates variables such as the timetable, number of nearby bus stops, and points-of-interest (POI) features. Detailed information is provided as follows.

All smart card transaction data in a metro system is accessible, including the anonymous ID number of each passenger, as well as the time and location (station) of each entry and exit. Boarding and alighting passenger flow at station s during the time interval t are represented by $y_{s,t}$ and $m_{s,t}$, respectively. Therefore, the passenger flow prediction problem is to learn a function that can map travel demand in the previous t time steps to that in the next l time steps based on individual travel behavior and other additional variables.

3.1. Modeling individual travel behavior with probability parallelogram

We introduce the concept of returning passenger flow for capturing individual travel behavior. According to the characteristics of passenger travel in a certain time window, passengers linked with station s (including board and alight) are classified into two categories and four sub-categories. Two categories are (G1) Passengers who alight at station s and have no historical trip boarding at station s; and (G2) Passengers who board at station s and have no historical trip alighting at station s. Passengers in group G1 are further divided into two subgroups, G1-A and G1-B, depending on whether they board the station s again within a specific time window. Similarly, passengers in group G2 are divided into G2-A and G2-B based on whether they alight at the station s within the same time frame. The four categories are shown in Fig. 1. The white dots represent the behavior of alighting, the black dots represent the behavior of boarding, the first column represents the time period, and the second to fifth columns are the trip records of passengers related to station s within the time period $[t-H,t]$. H is the time window length determined by quantifying the return time interval of passengers who alight at a certain station and then board that station again. Passengers are divided into two categories and four sub-categories according to the form of trip records, and the last two columns are the sum of the black and white dots in each row, $y_{s,t}$ and $m_{s,t}$ represent the boarding and alighting passenger flow at station s during the time interval t respectively.

The returning passenger flow concerned in this paper refers to group G1-A, who first get off and then get on from station s. They are likely to have some activities (shopping, play, work, home, etc.) near station s after arriving at station s, and then enter station s for the returning trip.

Fig. 2 displays the boarding flow structure categorized by the corresponding returning flow (G1-A), first board and then alight (G2-A), and one-way passengers (G1-B and G2-B) in dining, scenic, commercial, and residential areas. The analysis reveals that returning passenger flow makes up over 50 % of different types of stations, with residential areas having the highest proportion at 57.3 %. As the metro system mainly serves commuter traffic, returning passenger flow is particularly significant, especially in residential areas. Therefore, considering the travel patterns of returning passengers is essential for improving prediction accuracy.

The "returning passenger" $r_{s,t}$ is the number of people at time interval t in G1 who end their activities at station s and start their return trip within a certain time window. The definition of returning passenger flow only considers the continuation of metro trips and does not consider the connection with the metro generated by other travel modes. For example, passengers who commute on shared bikes in the morning and take the metro in the evening are not regarded to be returning passengers. This feature can be extracted with the support of multi-source data.

The returning passenger flow of station s at time t is obtained by the accumulation of returning passenger flow within a time window H, as shown in Eq. (1):

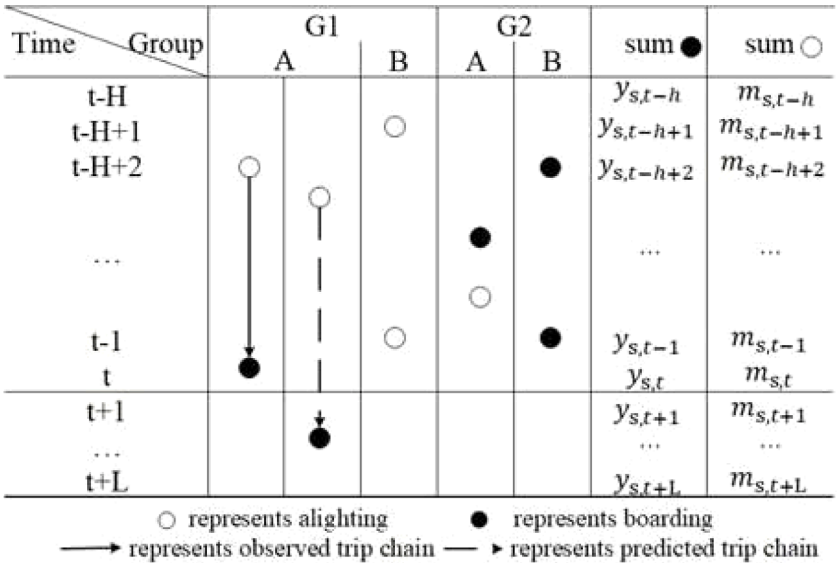


Fig. 1. Description of the two passenger groups G1 and G2.

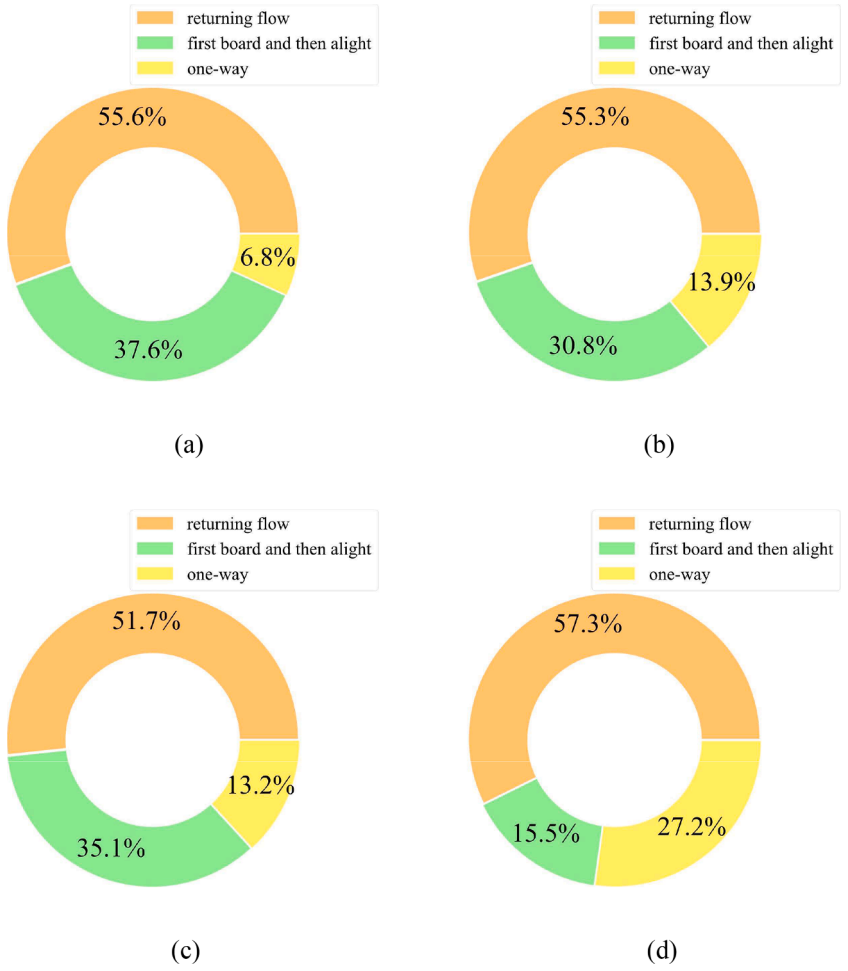


Fig. 2. Composition of boarding passenger flow (returning flow: G1-A; first board and then alight: G2-A; one-way: G1-B and G2-B) in a given week at different types of stations: (a) dining area; (b) scenic area; (c) commercial area; (d) residential area.

$$r_{s,t} = \sum_{t_a=t-H}^{t-1} r_{s,t_a,t} \tag{1}$$

Where $r_{s,t_a,t}$ is the number of passengers who alight at t_a at station s and then return (board) at station s at time t; H is the time window length determined by quantifying the return time interval of passengers who alight at a certain station and then board that station again.

When predicting inbound passenger flow at time $t+1$, the extraction of returning passenger flow $r_{s,t+1}$ should be considered. However, $r_{s,t+1}$ (the dashed arrows in Fig. 1) is not accessible since the returning passenger flow can only be observed in G1 before time t (the solid arrows in Fig. 1). Future returning passenger flow $r_{s,t+1}$ cannot be obtained by Eq. (1). Thus, a general method is needed to estimate $r_{s,t+1}$. The basic assumption is that there is a universal distribution $p_s(\tau_\alpha|\tau_\beta)$, which is called the return probability parallelogram. It can characterize the conditional probability of passengers in G1 who alight at the time τ_β will subsequently return at the time τ_α. For the time window g, p_s can be evaluated by the following equation:

$$p_s(g|g-h) = \frac{r_{s,t_a=g-h,t_b=g}}{m_{s,t_a=g-h}} \ (h=1,2,\ldots,H) \tag{2}$$

It is worth noting that the distribution p_s is defined in the entire group G1 so that subgroup G1-B is also included. Thus, for passengers who alight at the time t_a:

$$\sum_{t=t_a+1}^{t_a+H} p_s(\tau_\alpha = t|\tau_\beta = t_a) + p_s(\tau_\alpha = NA|\tau_\beta = t_a) = 1 \tag{3}$$

where $p_s(\tau_\alpha = NA|\tau_\beta = t_a)$ denotes the conditional probability that a passenger does not return within the time window length H (subgroup G1-B).

Then, $r_{s,t+1}$ is estimated by using the following equation:

$$\widehat{r}_{s,t+1} = \sum_{h=1}^{H} m_{s,t-h+1} p_s\left(\tau_\alpha = t+1 \middle| \tau_\beta = t\ +1-\ h\right) \tag{4}$$

It should be noted that estimating $r_{s,t+1}$ using Eq. (4) is different from predicting $r_{s,t+1}$ using a time series model based on historical data. This is because a simple time series model cannot capture the unique commuting behavior characteristics (going and returning) in the passenger flow data.

3.2. Relative spatiotemporal variable selection

3.2.1. Timetable characteristic

Metro systems are known for their precise timetables and the peak of boarding passenger flow that coincides with train arrivals. To ensure optimal service, operators take passenger flow into account when planning the train timetable, matching train capacity to expected passenger volume. Accurate prediction of passenger flow is crucial and there is a correlation between the number of arriving trains and passengers within a certain time window. To improve prediction accuracy, an additional variable is added in the form of the number of arriving trains in the upward and downward directions per hour. The notation $N_{s,t}$ represents the total number of train arrivals at station s at time t, with $N_{s,up,t}$ denoting the number of upward-direction train arrivals and $N_{s,down,t}$ denoting the number of downward-direction train arrivals. By incorporating this variable, operators can gain valuable insights into expected passenger volume and adjust services accordingly.

$$N_{s,t} = \left[N_{s,up,t}, N_{s,down,t}\right] \tag{5}$$

3.2.2. Bus stops characteristic

While the metro system boasts advantages such as high capacity, speed, and minimal interference, it falls short in terms of flexibility due to its predetermined layout and limited station locations. This often requires commuters to rely on alternative modes of transportation like buses and shared bikes to connect with the metro system. Moreover, the accessibility of metro stations via bus routes can have a significant impact on passenger flow. In this paper, we take into account the number of bus stops within 500 m, 1000 m, and 1500 m of the metro station denoted by $B_{s,1}, B_{s,2}$ and $B_{s,3}$, respectively, and incorporate them into our prediction model.

$$B_{s,dist} = \left[B_{s,1}, B_{s,2}, B_{s,3}\right] \tag{6}$$

3.2.3. Points of interest characteristic

Points of interest (POI) refer to location-based information about features like stores, houses, and stations on a map. This data is currently available through Amap's open platform and primarily includes four attributes: name, address, coordinates, and category. The passenger flow of a metro station is closely linked to the surrounding land use characteristics. For example, if there are many residential areas around a metro station, the passenger flow will mainly consist of morning and evening commuters. On the other hand, if there are several catering establishments in the vicinity, then the station will have a high passenger flow during meal times. This paper uses POI data to describe the land use around a metro station. Specifically, we introduce six variables that represent the number of POI in six categories: commercial, residential, scenic, financial, vehicular, and dining. We define I_s as the set of POI around a given station s, where $I_{s,tra}$, $I_{s,din}$, $I_{s,biz}$, $I_{s,int}$, $I_{s,res}$ and $I_{s,gov}$ represent the number of vehicular, dining, commercial, scenic, residential, and financial services, respectively.

$$I_s = \left[I_{s,tra}, I_{s,din}, I_{s,biz}, I_{s,int}, I_{s,res}, I_{s,gov}\right] \tag{7}$$

This paper proposes several passenger flow prediction models in the metro system based on various characteristics such as timetable, bus accessibility, and POI. The baseline model, M_{base}, is a basic time series model with no additional variables. Other models include $M_{r,t+1}$, which uses estimated values $\widehat{r}_{s,t+1}$ to approximate actual passenger flow $r_{s,t+1}$. And model uses $r_{s,t}$, which can be obtained, as an alternative to $r_{s,t+1}$ is illustrated as $M_{r,t}$. Model $M_{r,t}^{train}$ is conducted by adding the number of train arrivals per hour to $M_{r,t}$. Then the POI is added ($M_{r,t}^{POI}$) to explore the utility based on the model $M_{r,t}^{train}$. Last, this paper takes all characteristics into account ($M_{r,t+1}^{all}$). Table 1 summarizes the different prediction models and their defining characteristics.

Table 1
Prediction models with various characteristics.

	$y_{s,1:t}$	$r_{s,t+1}$	$r_{s,t}$	$N_{s,t+1}$	I_s	$B_{s,dist}$
M_{base}	√					
$M_{r,t+1}$	√	√				
$M_{r,t}$	√		√			
$M_{r,t}^{train}$	√		√	√		
$M_{r,t}^{POI}$	√		√	√	√	
$M_{r,t+1}^{all}$	√	√		√	√	√

3.3. Smart prediction models

3.3.1. XGBoost algorithm

Extreme gradient boosting (XGBoost) is a powerful machine learning algorithm that blends multiple tree models to create a robust predictor (Chen & Guestrin, 2016). The fundamental idea behind the XGBoost approach is to train a new function at every iteration to capture the residuals of the previous predictions. This new function computes scores for each node based on the characteristics of the sample data, and the predicted value of the sample is the sum of all these scores. In effect, XGBoost leverages an ensemble of decision trees that progressively learn from the errors of previous models, resulting in remarkable accuracy and generalization performance.

$$\widehat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{8}$$

where $\widehat{y}_i$ is the predicted value of the model, x_i is the label of the ith sample, k is the number of trees, and f_k is the kth tree model.

The XGBoost algorithm is to learn these K trees in the prediction process. When the loss function of the tree is the smallest, the model is the optimal model, and the prediction accuracy is also the highest, which can be expressed as follows:

$$obj(t) = \sum_{i=1}^{n} l(y_i, \widehat{y}_i) + \sum_{k=1}^{K} \Omega(f(t)) \tag{9}$$

$$\Omega(f(t)) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{10}$$

where t is the number of iterations. $obj(t)$ is the value of the loss function. $l(y_i, \widehat{y}_i)$ is the training error, which is generally a constant and is used to measure the difference between the predicted and actual scores. $\Omega(f(t))$ is the complexity of the whole tree. T is the total number of leaf nodes. γ is the difficulty of node slicing, used to control the leaf nodes scores and prevent overfitting. w is the node vector mode of the leaves. λ is the regularization factor.

The objective function is used to characterize whether the algorithm is optimal (Shi et al., 2021). XGBoost uses a greedy algorithm to traverse the feature division points of all features, and if the objective function after splitting gains over the pre-split and exceeds the set threshold, it can be split. The splitting is stopped when the weight and maximum depth exceed the set threshold, and the appropriate learning function is continuously found in the splitting. The conditional function to determine if split or not is as follows:

$$Gain = \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma \tag{11}$$

where G_L and G_R are the sum of the first-order partial derivatives of the samples contained in the left and right leaf nodes respectively. H_L and H_R are the sum of the second-order partial derivatives of the samples contained in the left and right leaf nodes respectively. $\frac{G_L^2}{H_L+\lambda}$ is the left node score after the cut of a node. $\frac{G_R^2}{H_R+\lambda}$ is the right node score after the node cut. $\frac{(G_L+G_R)^2}{H_L+H_R+\lambda}$ is the pre-cut score. $Gain$ is the conditional function to judge whether to split or not, which is equivalent to the pre-cut objective function minus the post-cut left and right objective functions, if $Gain > 0$ then split and if $Gain < 0$ then do not split.

3.3.2. CatBoost algorithm

CatBoost is based on gradient-boosting decision trees. The advantages of CatBoost are that it can handle category-based feature variables efficiently and reasonably, and build trees for unbiased gradient estimates in each iteration to mitigate prediction bias. Thus, it can improve the prediction accuracy and generalization ability of the model (Chang et al., 2019).

Category-based features are discrete features, usually in the form of strings, where each value represents a specific category. They cannot be used as inputs directly, and they need to be processed. The processing method is to scramble the order of the number $D = \{(x_i, y_i)\}_{i=1,\ldots n}$. The scrambled sequence is $\sigma = (\sigma_1, \cdots, \sigma_n)$ and then traverses σ_1 to σ_n and the zth record is used to calculate the value of the categorical feature.

$$\sigma_{z,k} = \frac{\sum_j^{z-1}\left[x_{\sigma_{j,k}} = x_{\sigma_{i,k}}\right] \cdot Y_{\sigma_j} + \alpha \cdot z}{\sum_j^{z-1}\left[x_{\sigma_{j,k}} = x_{\sigma_{i,k}}\right] + \alpha} \tag{12}$$

where z represents the prior term and $\alpha > 0$ represents the weight coefficient of the prior term. Adding the prior term can reduce the noise caused by low-frequency features in the category features. For the regression problem, the prior term is the mean value of the data set labels.

The GBDT algorithm will produce gradient bias and overfitting in the process of fitting the gradient of the current model because it uses the same data points for estimation. To address this problem, CatBoost uses the Ordered Boosting method to change the gradient estimation in GBDT from biased to unbiased. The Ordered Boosting method first generates a $[1, n]$ permutation σ randomly. And σ is used to sort the original sample and initialize n different models $M_1, M_2, \ldots, M_n$. Each M_i only use the top ith samples of the random permutation, and at each iterative step, the unbiased gradient estimate of the jth sample is obtained by model M_{j-1}.

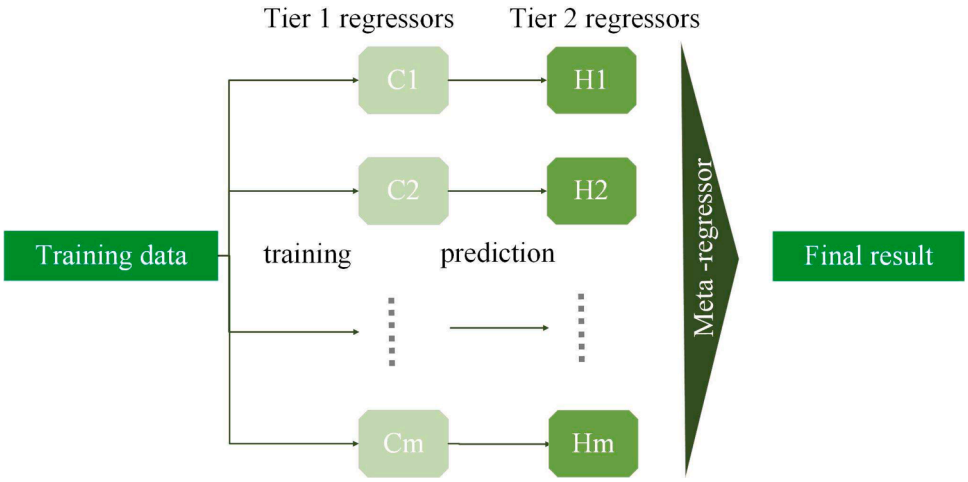


Fig. 3. Illustration of stacking strategy.

3.3.3. *Smart stacking-catboost algorithm*

Stacking is a framework for integrating multiple models in a hierarchical manner, where stacking regression is a type of ensemble learning that leverages a meta regressor to combine multiple base regression models (Wang & Zhang, 2023). In a two-level stacking approach, multiple primary learners are first trained on the training set. Next, the primary learners are used to predict the testing set, and the resulting output values are treated as inputs for the next stage of training. The final label generated by the primary learners is then used as the output value for training the secondary learner. The stacking strategy is illustrated in Fig. 3. By employing this approach, stacking can improve the accuracy of predictions by leveraging the strengths of multiple models and combining their outputs in a meaningful way.

Various algorithms observe data from different scopes and structures, creating models based on their observations to make predictions on new datasets. In this paper, we adopt the stacking strategy to merge a diverse range of algorithms for more comprehensive results. The combination of models can synthesize strengths and compensate for weaknesses, resulting in highly accurate predictions. Additionally, utilizing different training data in two sessions reduces the risk of overfitting (Cao et al., 2022).

This paper first analyzes the prediction performance of Random Forest (RF), Gradient Boosting Decision Tree (GBDT), Light Gradient Boosting Machine (LightGBM), Extreme Gradient Boosting (XGBoost), and CategoricalBoosting (CatBoost) models individually. Afterward, we utilize the stacking strategy to investigate performance improvement. We explore the optimal stacking strategy by examining various combinations of base regressors and meta-regressors formed through permutation and combination of the five models.

4. Experiments and results

In order to examine passenger travel patterns, this study uses the Beijing metro dataset from a one-month period, which is collected and obtained from the smart card automatic fare collection (AFC) system. The data includes the ID number of each passenger, the time and the station of each entry and exit, and other information. The main data forms used in this paper are shown in Table 2. CARD_CODE represents the ID number of a single passenger; ENTRY_TIME indicates the time a passenger swipes their card to enter the metro station; DEAL_TIME indicates the time to exit; ORIGION_LOCATION represents the AFC clearing center (ACC) code of the station where the passenger swipes into the station, while CURRENT_LOCATION represents the station swiping to exit.

We aggregate the individual travel records to get the time-sharing site-level passenger flow. The spatial distribution of inbound flow in peak and non-peak hours on weekdays and weekends is shown in Fig. 4. Fig. 4(a) represents the bubble diagram of inbound passenger flow from 10:00 to 11:00 on a weekday, in which the size and color of bubbles represent passenger flow. It can be seen that there is a large number of metro passengers flowing into the station during peak hours of the weekday, which is especially obvious at the transfer station. The spatial distribution of origin-destination (OD) flow in peak and non-peak hours on weekdays and weekends is shown in Fig. 5. Fig. 5(a) represents the OD flow from 8:00 to 9:00 on a weekday, in which the width and brightness of the lines represent passenger flow. OD passenger flow is notably higher on weekdays as well. Therefore, accurate short-term demand prediction for large passenger flow stations is crucial to optimize passenger flow control and to seek the optimal matching of transport capacity and traffic volume.

As for the availability of the dataset for further study. The main data used in this study is the Beijing metro dataset, which contains the individual ID number. To ensure privacy protection, the individual IDs should be anonymized. The passenger flow volumes at different periods in the Beijing metro can be made publicly available. The metro timetable data should be handled with caution as it is related to the operation and safety of the metro system. After anonymization, the metro timetable data is available on request from the authors. Points of interest data and bus stop numbers around metro stations are openly available and can be crawled from Amap's open platform.

Table 2
Description of data.

Name	Description	Format
CARD_CODE	ID number	8094****
ENTRY_TIME	Entry time	2017–03–16 04:45:00
DEAL_TIME	Exit time	2017–03–16 06:02:03
ORIGION_LOCATION	ACC code of the entry station	15,101****
CURRENT_LOCATION	ACC code of the exit station	15,099****

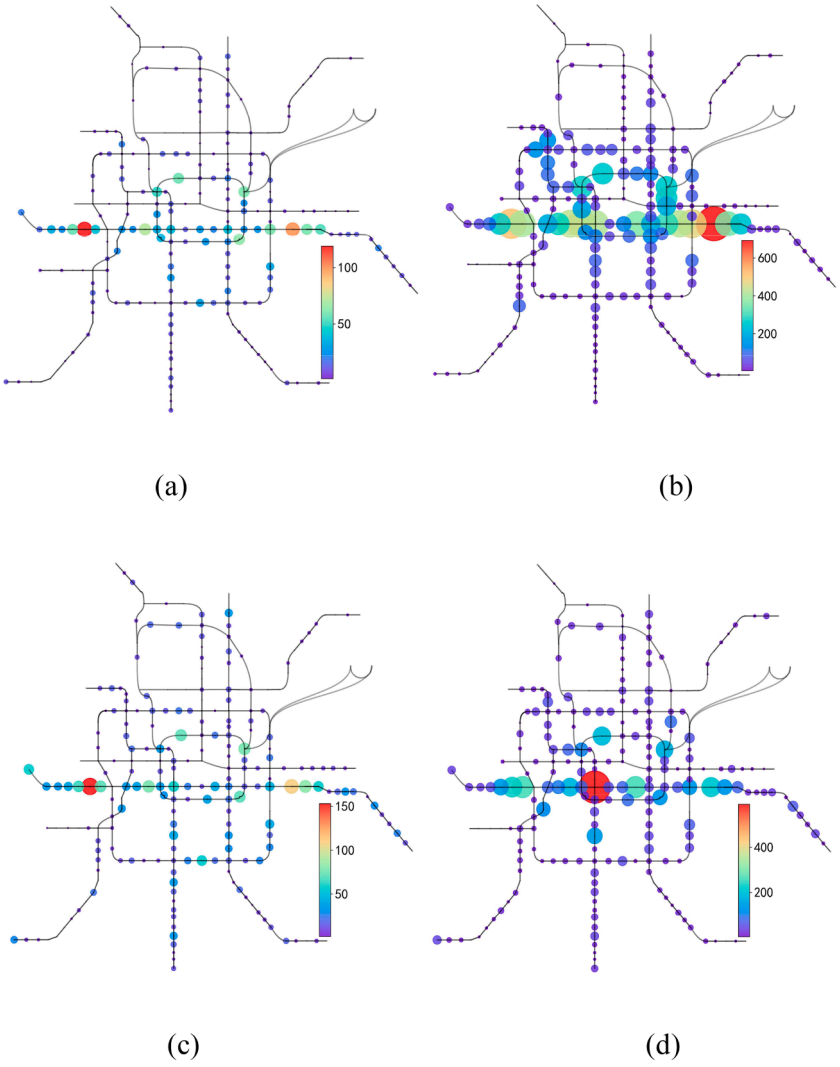


Fig. 4. The spatial distribution of inbound passenger flow: (a) weekday, 10:00–11:00; (b) weekday, 18:00–19:00; (c) weekend, 10:00−11:00; (d) weekend, 18:00–19:00.

To evaluate the effectiveness of our proposed integrated model based on individual travel behavior, we use real-world Beijing metro AFC data from a month-long period. We compare our proposed model with various travel demand prediction baseline models, including History Average (HA), Linear Regressor (LR), ARIMA, RF, GBDT, LightGBM, XGBoost, and CatBoost. Table 1 displays the distinctive variables used in our proposed model. We use 70 % of the data for training and the remaining 30 % for testing. Model accuracy is assessed by the coefficient of determination (R2 score), mean absolute error (MAE), and weighted mean absolute percentage error (WMAPE). WMAPE is used to overcome the drawbacks of MAPE when dealing with extreme values, such as the real value being zero.

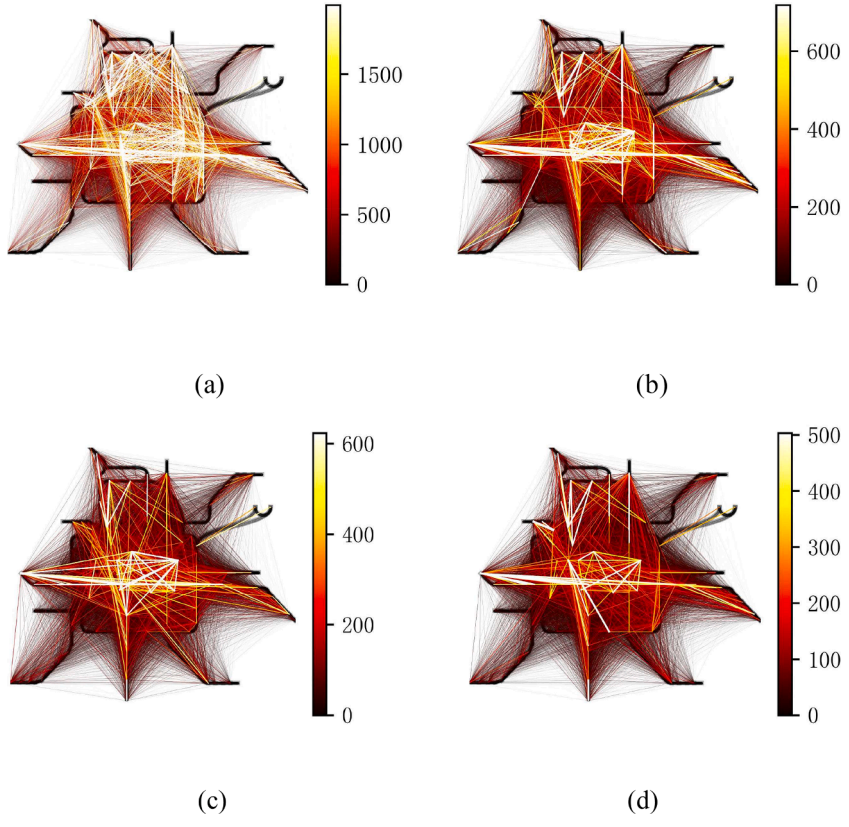


Fig. 5. The spatial distribution of OD passenger flow: (a) weekday, 7:00–9:00; (b) weekday, 15:00–16:00; (c) weekend, 7:00–9:00; (d) weekend, 15:00–16:00.

$$MAE = \frac{1}{N}\sum_{t=1}^{N}\left|\left(y_{s,t} - \widehat{y}_{s,t}\right)\right| \tag{13}$$

$$R^2 = 1 - \frac{\left(\sum_{t=1}^{N}\left(y_{s,t} - \widehat{y}_{s,t}\right)\right)^2}{\sum_{t=1}^{N}\left(y_{s,t} - \overline{y}_{s,t}\right)^2} \tag{14}$$

$$WMAPE = \frac{\sum_{t=1}^{N}\left|\left(y_{s,t} - \widehat{y}_{s,t}\right)\right|}{\sum_{t=1}^{N} y_{s,t}} \tag{15}$$

where $y_{s,t}$ and $\widehat{y}_{s,t}$ are the actual boarding passenger flow and the predicted boarding passenger flow respectively, and $\overline{y}_{s,t}$ is the average of the actual boarding passenger flow.

The unknown parameter H and $p^s(\tau_\beta|\tau_\alpha)$ in Eq. (4) is determined through analysis of the Beijing metro dataset. The time window H is determined by calculating the return time interval $(\tau_\beta - \tau_\alpha)$ of passengers who alight at one station and subsequently board again at the same station. Fig. 6 shows the return time interval of weekdays and weekends at four representative stations located in dining, residential, commercial, and scenic areas. The horizontal axis represents the return time interval, that is, the time that passengers re-enter the station minus the time that they previously exited from the station$(\tau_\beta - \tau_\alpha)$, and the vertical axis represents the number of passengers. Fig. 6 illustrates the distribution of passengers' return time intervals at each representative station. All the alighting records on Monday and Saturday are collected, and the returning passenger flow within 40 h after alight is tracked. Bimodal peaks are observed on weekdays for all four stations. The first peak (less than 4 h) corresponds to shorter-duration activities (such as eating and shopping), while the longer peaks (around 10 h in commercial areas, residential areas, and scenic areas) are usually associated with longer-duration activities such as work, home, and entertainment. On weekends, the return time interval is mainly concentrated in about 2 h in the dining and scenic areas, while it is evenly distributed in the commercial and residential areas. The chosen time interval threshold H in this study is 24 h. This is because, as observed in Fig. 6, the majority of return time intervals are less than 24 h, and 24 h can encompass both short-duration and long-duration travel activities.

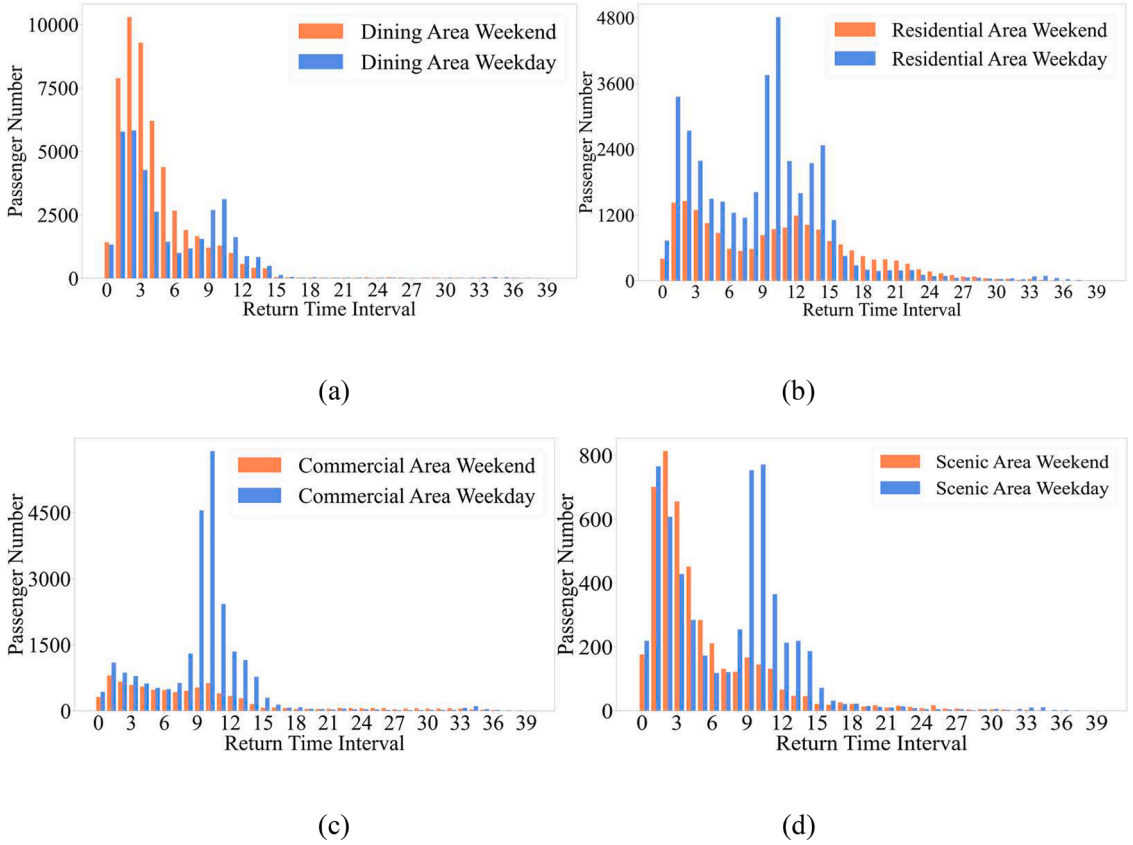


Fig. 6. Distribution of return time intervals on weekdays/weekends at different types of stations: (a) dining area; (b) residential area; (c) commercial area; (d) scenic area.

Fig. 7 depicts the conditional probability distribution $p^s(\tau_\beta|\tau_\alpha)$ based on the result of the return probability parallelogram. The figure illustrates the conditional distribution of weekends/weekdays at representative stations in the dining, scenic, residential, and commercial areas, as shown in (a), (b), (c), and (d), respectively. The time range considered is from 5:00 to 23:00 (metro system operation time) with a time granularity of one hour. The horizontal axis connects 23:00 on day k with 5:00 on day $k+1$. There are two blank triangles in Fig. 7, the left one is undefined and corresponds to $t_b \leq t_a$; and the right one corresponds to $t_b > t_a + H(H = 24)$. It should be noted that the sum of each column should be less than 1, because it does not include passengers of G1-B (passengers without a return trip, i.e., $\tau_\beta = NA$). As shown in Fig. 7, the parallelogram structure of various functional regions can reflect the regional returning passenger flow's distinctive features, such as the residential area and commercial area being different. The structure of returning passenger flow varies dramatically between weekdays and weekends, with weekdays having a much larger proportion. However, estimating the existing Eq. (2) poses a challenge due to its range of conditional probabilities τ_α for each day. Upon analysis, it is found that the conditional distributions between weekdays (weekends) and weekdays (weekends) are approximately similar during the same period for different dates. However, the conditional distributions between weekdays and weekends differ considerably. Therefore, this paper assumes that the return probability for the same period is the same on different weekdays and on different weekends.

4.1. *Prediction performance integrating different features*

To examine the impact of various factors on inbound passenger flow prediction, we develop several models. The first is a basic time series model, M_{base}, that does not incorporate any additional variables. We then construct $M_{r,t+1}$, which utilizes an estimated value $\widehat{r}_{s,t+1}$ as an approximation for $r_{s,t+1}$. Model $M_{r,t}$, which uses $r_{s,t}$ as a substitute for $r_{s,t+1}$, is also tested. To evaluate the effect of train arrivals, we create $M_{r,t}^{train}$ by adding the number of train arrivals per hour. Additionally, we introduce points of interest (POI) to the model $M_{r,t}^{POI}$. Finally, we combine all of the characteristics into one model, $M_{r,t+1}^{all}$. Our experiment begins with an investigation of the standard time series prediction model M_{base}, which only utilizes historical passenger flow data as a feature. Fig. 8 displays the performance of each model across different historical time steps. Our analysis reveals that XGBoost and CatBoost outperform the other integration models. XGBoost achieves MAE scores of 295.6, 280.77, 248.20, and 217.59 with the historical step equal to 4, 5, 6, and 7, respectively. The R2 score of LightGBM is 0.79, 0.76, 0.77, 0.82 with the historical step equal to 4, 5, 6, and 7. Based on these results,

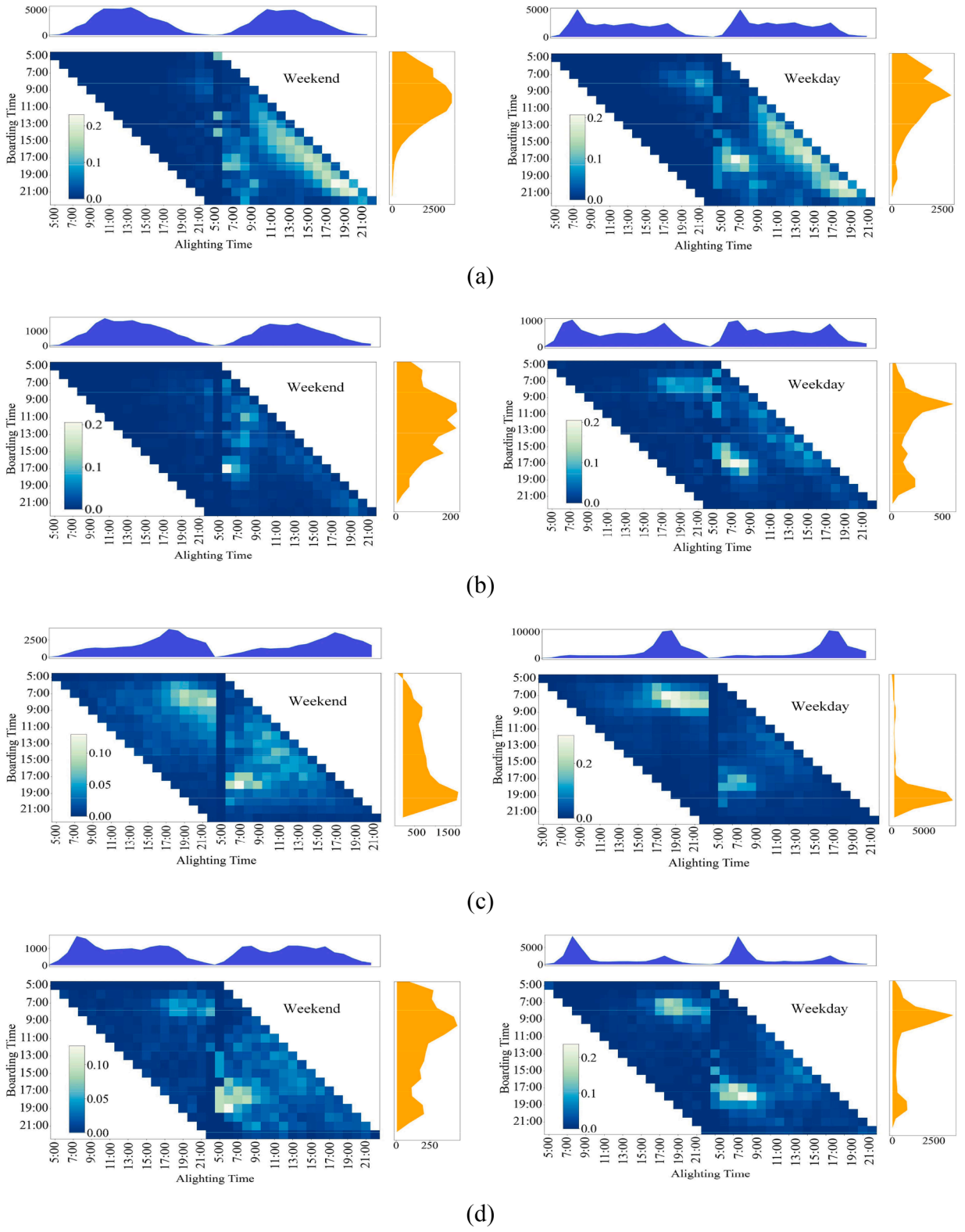


Fig. 7. Return probability parallelogram on weekends/weekdays: (a) dining area; (b) scenic area; (c) residential area; (d) commercial area.

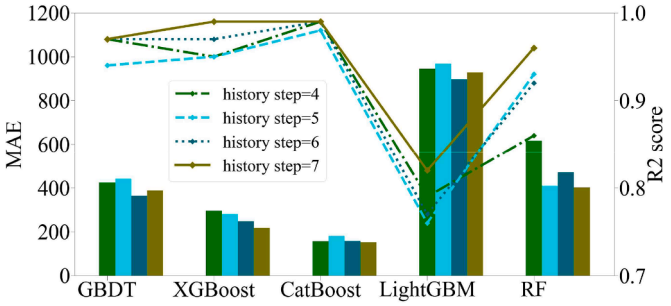


Fig. 8. Model prediction performance under different historical time steps.

we select a historical time step of 7 for further analysis.

Table 3 presents a comparison of the prediction effects of three models: M_{base}, $M_{r,t}$ and $M_{r,t}^{train}$. The results indicate that the addition of the timetable feature leads to improved prediction accuracy. For instance, in the residential area, the MAE for CatBoost on M_{base} is 173.20 with a WMAPE value of 0.10, while for $M_{r,t}$ it is 151.93 with a WMAPE of 0.06. The MAE on the model $M_{r,t}^{train}$ is 136.55 and WMAPE is 0.05. Thus, the inclusion of $r_{s,t}$ and timetable enhances the MAE by 12.28 % and 21.16 % compared to M_{base}. Similarly, for the dining area, the MAE for CatBoost on model M_{base} is 198.77 with a WMAPE value of 0.09, whereas for $M_{r,t}$ it is 181.96 with a WMAPE of 0.08. The MAE on the model $M_{r,t}^{train}$ is 161.86 and the WMAPE value is 0.07. The addition of $r_{s,t}$ and timetable improves the WMAPE by 11.11 % and 22.22 % compared to M_{base}. Although adding the returning passenger flow characteristic $r_{s,t}$ for predicting the boarding passenger flow $y_{s,t+1}$ has some improvement effect, it is negligible when comparing the results of M_{base} and $M_{r,t+1}$ in Table 3. Therefore, this paper aims to further explore the effect of the model $M_{r,t+1}$ with the addition of the feature $\hat{r}_{s,t+1}$.

Table 4 shows the prediction performance integrating returning passenger flow. In the dining area, the XGBoost model achieves an MAE of 245.58 and a WMAPE of 0.11 on the model M_{base}, while for the model $M_{r,t}$, it achieves an MAE of 238.98 and a WMAPE of 0.10. For the model $M_{r,t+1}$, the MAE is 167.67 and the WMAPE is 0.07. Compared to the model M_{base}, the model $M_{r,t}$ improves by 2.69 % in MAE and 9.09 % in WMAPE, while the model $M_{r,t+1}$ shows significant improvement with 31.72 % in MAE and 36.36 % in WMAPE when estimated $\hat{r}_{s,t+1}$ is incorporated. These results suggest that adding returning passenger flow data significantly improves the accuracy of boarding passenger flow prediction compared to only considering boarding passenger flow.

The passenger flow of a metro station is closely linked to the facilities available in its surrounding area. Metro stations that have more dining options nearby typically experience a higher volume of round-trip passengers during prime dining hours. To describe the

Table 3
Prediction performance integrating historical returning passenger flow and timetable.

	Measurement	HA	LR	ARIMA	GBDT	XGBoost	CatBoost	LightGBM
(a) Dining Area								
M_{base}	MAE	2321.55	471.31	2321.52	356.93	245.58	198.77	707.29
	WMAPE	1.00	0.20	1.00	0.15	0.11	0.09	0.30
$M_{r,t}$	MAE	2367.89	434.45	2367.86	304.52	238.98	181.96	578.58
	WMAPE	1.00	0.18	1.00	0.13	0.10	0.08	0.24
$M_{r,t}^{train}$	MAE	2367.83	384.78	2367.88	302.67	223.76	161.86	581.58
	WMAPE	1.00	0.16	1.00	0.13	0.09	0.07	0.25
(b) Commercial Area								
M_{base}	MAE	2553.92	1112.84	2553.92	388.64	217.59	152.47	928.68
	WMAPE	1.00	0.44	1.00	0.15	0.09	0.06	0.36
$M_{r,t}$	MAE	1791.64	596.28	1791.72	337.42	207.80	153.74	558.17
	WMAPE	1.00	0.33	1.00	0.19	0.12	0.09	0.31
$M_{r,t}^{train}$	MAE	2553.84	1023.90	2554.01	367.22	199.21	134.96	924.09
	WMAPE	1.00	0.40	1.00	0.14	0.08	0.05	0.36
(c) Scenic Area								
M_{base}	MAE	331.05	101.54	331	66.11	53.27	48.46	84.49
	WMAPE	1.00	0.31	1.00	0.20	0.16	0.15	0.26
$M_{r,t}$	MAE	331.06	102.13	331.07	64.8	49.11	47.19	83.4
	WMAPE	1.00	0.31	1.00	0.20	0.15	0.14	0.25
$M_{r,t}^{train}$	MAE	330.99	73.89	331.08	50.68	43.96	39.36	75.56
	WMAPE	1.00	0.22	1.00	0.15	0.13	0.12	0.23
(d) Residential Area								
M_{base}	MAE	1791.63	586.57	1791.58	321.33	204.19	173.20	581.01
	WMAPE	1.00	0.33	1.00	0.18	0.11	0.10	0.32
$M_{r,t}$	MAE	2553.93	1139.71	2554	388.23	217.67	151.93	924.84
	WMAPE	1.00	0.45	1.00	0.15	0.09	0.06	0.36
$M_{r,t}^{train}$	MAE	2553.84	1058.80	2554.00	394.94	201.79	136.55	927.96
	WMAPE	1.00	0.41	1.00	0.15	0.08	0.05	0.36

Table 4
Prediction performance integrating returning passenger flow.

	Measurement	HA	LR	ARIMA	GBDT	XGBoost	CatBoost	LightGBM
(a) Dining Area								
M_{base}	MAE	2321.55	471.31	2321.52	356.93	245.58	198.77	707.29
	WMAPE	1.00	0.20	1.00	0.15	0.11	0.09	0.30
$M_{r,t}$	MAE	2367.89	434.45	2367.86	304.52	238.98	181.96	578.58
	WMAPE	1.00	0.18	1.00	0.13	0.10	0.08	0.24
$M_{r,t+1}$	MAE	2367.89	175.05	2367.91	228.91	167.67	131.80	563.23
	WMAPE	1.00	0.07	1.00	0.10	0.07	0.06	0.24
(b) Commercial Area								
M_{base}	MAE	2553.92	1112.84	2553.92	388.64	217.59	152.47	928.68
	WMAPE	1.00	0.44	1.00	0.15	0.09	0.06	0.36
$M_{r,t}$	MAE	1791.64	596.28	1791.72	337.42	207.8	153.74	558.17
	WMAPE	1.00	0.33	1.00	0.19	0.12	0.09	0.31
$M_{r,t+1}$	MAE	1791.63	285.93	1791.67	242.46	174.41	110.53	460.18
	WMAPE	1.00	0.16	1.00	0.14	0.10	0.06	0.26
(c) Scenic Area								
M_{base}	MAE	331.05	101.54	331	66.11	53.27	48.46	84.49
	WMAPE	1.00	0.31	1.00	0.20	0.16	0.15	0.26
$M_{r,t}$	MAE	331.06	102.13	331.07	64.8	49.11	47.19	83.40
	WMAPE	1.00	0.31	1.00	0.20	0.15	0.14	0.25
$M_{r,t+1}$	MAE	331.05	57.28	331.04	42.7	40.61	34.3	69.12
	WMAPE	1.00	0.31	1.00	0.20	0.15	0.14	0.25
(d) Residential Area								
M_{base}	MAE	1791.63	586.57	1791.58	321.33	204.19	173.20	581.01
	WMAPE	1.00	0.33	1.00	0.18	0.11	0.10	0.32
$M_{r,t}$	MAE	2553.93	1139.71	2554.00	388.23	217.67	151.93	924.84
	WMAPE	1.00	0.45	1.00	0.15	0.09	0.06	0.36
$M_{r,t+1}$	MAE	2553.93	628.80	2553.99	337.88	169.15	130.40	888.64
	WMAPE	1.00	0.25	1.00	0.13	0.07	0.05	0.35

Table 5
Prediction performance in different functional areas.

Station type	Model	MAE (M_{base})	MAE ($M^{all}_{r,t+1}$)	WMAPE (M_{base})	WMAPE ($M^{all}_{r,t+1}$)	WMAPE Improvement	MAE Improvement
Residential Area	HA	1374.88	1374.77	1.00	1.00	0.00 %	0.01 %
	LR	594.66	467.62	0.43	0.34	26.47 %	27.17 %
	ARIMA	1374.92	1374.58	1.00	1.00	0.00 %	0.02 %
	GBDT	389.05	359.7	0.28	0.26	7.69 %	8.16 %
	XGBoost	186.18	147.48	0.14	0.11	27.27 %	26.24 %
	CatBoost	140.02	121.11	0.10	0.09	11.11 %	15.61 %
	LightGBM	614.27	564.67	0.45	0.41	9.76 %	8.78 %
	RF	481.00	427.06	0.35	0.31	12.90 %	12.63 %
Commercial Area	HA	1887.36	1887.36	1.00	1.00	0.00 %	0.00 %
	LR	590.02	511.12	0.31	0.27	14.81 %	15.44 %
	ARIMA	1887.38	1887.16	1.00	1.00	0.00 %	0.01 %
	GBDT	471.33	415.68	0.25	0.22	13.64 %	13.39 %
	XGBoost	276.59	207.08	0.15	0.11	36.36 %	33.57 %
	CatBoost	235.17	175.80	0.12	0.09	33.33 %	33.77 %
	LightGBM	611.15	560.88	0.32	0.3	6.67 %	8.96 %
	RF	532.92	485.31	0.28	0.26	7.69 %	9.81 %
Dining Area	HA	1536.96	1536.86	1.00	1.00	0.00 %	0.01 %
	LR	471.45	370.35	0.31	0.24	29.17 %	21.44 %
	ARIMA	1536.99	1536.64	1.00	1.00	0.00 %	0.02 %
	GBDT	365.61	313.02	0.24	0.2	20.00 %	14.38 %
	XGBoost	181.98	149.03	0.12	0.10	20.00 %	18.11 %
	CatBoost	183.79	126.45	0.12	0.08	50.00 %	31.20 %
	LightGBM	561.5	542.69	0.37	0.35	5.71 %	3.35 %
	RF	400.87	369.76	0.26	0.24	8.33 %	7.76 %
Scenic Area	HA	562.2	562.2	1.00	1.00	0.00 %	0.00 %
	LR	172.48	159.56	0.31	0.28	10.71 %	8.10 %
	ARIMA	562.21	562.1	1.00	1.00	0.00 %	0.02 %
	GBDT	145.2	144.19	0.26	0.26	0.00 %	0.70 %
	XGBoost	86.39	64.91	0.15	0.12	25.00 %	33.09 %
	CatBoost	80.87	56.97	0.14	0.10	40.00 %	41.95 %
	LightGBM	231.38	223.15	0.41	0.4	2.50 %	3.69 %
	RF	163.1	155.92	0.29	0.28	3.57 %	4.60 %

land use around metro stations, we use POI data. We take a metro station as the center of a circle with a radius of 500 m, 1000 m, or 1500 m and obtain POI data within that range using the Amap open platform. We then statistically summarize the data to get the POI number for six categories: commercial, residential, scenic, financial, vehicular, and dining. In addition, since metros usually connect with other modes of public transportation for people's commutes, this paper also considers the number of bus stops around the metro station.

Table 5 demonstrates how bus accessibility and POI near metro stations affect passenger flow. Take the commercial area as an example, the MAE of the baseline model HA, LR, ARIMA, GBDT, XGBoost, CatBoost, LightGBM, and RF is 1887.36, 590.02, 1887.38, 471.33, 276.599, 235.17, 611.15, 532.92. After incorporating bus accessibility and POI near metro stations in model $M^{all}_{r,t+1}$, the MAE for models HA, LR, ARIMA, GBDT, XGBoost, CatBoost, LightGBM, and RF becomes 1887.36, 511.12, 1887.16, 415.68, 207.08, 175.80, 560.88, 485.31. Compared with M_{base}, model $M^{all}_{r,t+1}$ improves the MAE of GBDT, XGBoost, CatBoost, and LightGBM by 13.39 %, 33.57 %, 33.77 %, and 8.96 %. Incorporating POI and bus accessibility leads to a significant improvement in prediction accuracy in the commercial area, demonstrating that these factors have a substantial impact on prediction accuracy.

4.2. Selection of different stacking strategies

Stacking regression is a machine learning technique that combines multiple regression models using a meta-regressor. The output of each base regression model, known as a meta-feature, serves as input for the meta-regressor. In this study, five models (RF, GBDT, LightGBM, XGBoost, and CatBoost) are used as base regressors, with LR serving as the meta-regressor. Different combinations of base regressors and meta-regressors are tested to determine the best stacking strategy. Table 6 shows the prediction results of each strategy.

Table 6 presents the performance of different stacking models, where two single machine learning algorithms are chosen as the base regressors for models 1–5. For instance, in model 1, XGBoost and CatBoost serve as the base regressors, while LR, XGBoost, LightGBM, CatBoost, RF, and GBDT are used as the meta-regressor in turn. We observe that using XGBoost as the base regressor yields superior results compared to the other algorithms. Therefore, in models 6–9, XGBoost is adopted as the base regressor. Among these models, the best prediction accuracy, the MAE value of 161.22, is obtained with the combination of XGBoost, CatBoost, and RF as the base regressors, and CatBoost as the meta-regressor. However, when all the different algorithms are fused as the base regressor in model 10, the best prediction result achieves an MAE of only 163.71, which is a 1.5 % decrease compared to the best accuracy. This suggests that selecting the optimal base models for the stacking strategy requires careful consideration of the fusion strategy and parameter selection, rather than simply using more or fewer models.

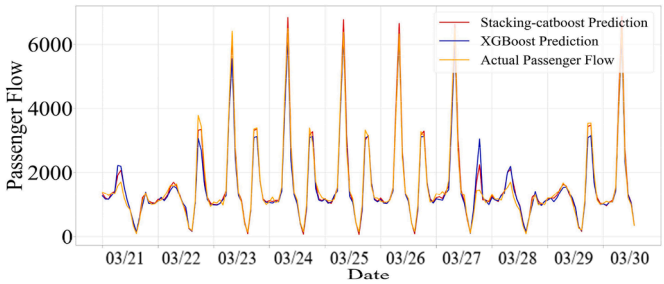
Fig. 9 displays the predicted and actual boarding passenger flow for different station types using XGBoost and Stacking-Catboost for seven days on the testing set. The predicted and actual values follow a similar trend, suggesting a high level of prediction accuracy. Notably, the proposed Stacking-Catboost strategy outperforms the well-performed XGBoost method.

In order to further demonstrate the effectiveness of the proposed return probability parallelogram for predicting the returning passenger flow. With the actual returning passenger flow as the horizontal axis and the estimated value as the vertical axis, the scatter diagram is drawn, as shown in Fig. 10. The scatter points in representative stations of each type are distributed on both sides of the line $y = x$. The residential area has very few anomalies, which can be ignored. This further demonstrates that the proposed return probability parallelogram can predict the returning passenger flow at time $t+1$ accurately.

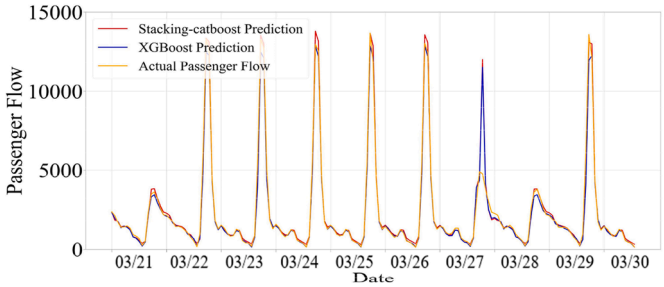
This paper explores the performance of the individual travel behavior model in different time periods, including morning peak,

Table 6
Effects of different smart stacking strategies.

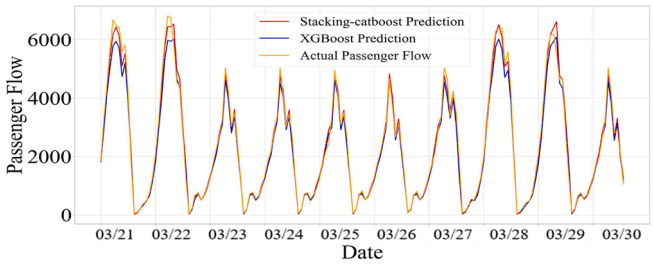
	Base regressor					Measurement	Meta regressor					
	XGBoost	LightGBM	CatBoost	GBDT	RF		LR	XGBoost	LightGBM	CatBoost	GBDT	RF
1	√		√			MAE	173.94	195.86	433.47	171.18	235.50	194.28
						WMAPE	0.09	0.10	0.23	0.09	0.13	0.10
2	√			√		MAE	173.19	207.39	439.08	181.68	249.01	205.54
						WMAPE	0.09	0.11	0.23	0.10	0.13	0.11
3		√		√		MAE	341.91	335.71	513.07	322.15	365.91	333.05
						WMAPE	0.18	0.18	0.27	0.17	0.19	0.18
4		√			√	MAE	374.07	353.42	510.11	341.23	389.23	375.79
						WMAPE	0.20	0.19	0.27	0.18	0.21	0.20
5			√		√	MAE	166.82	198.87	435.84	167.52	240.33	201.18
						WMAPE	0.09	0.11	0.23	0.09	0.13	0.11
6	√	√		√		MAE	175.08	207.00	439.08	182.28	248.38	205.54
						WMAPE	0.09	0.11	0.23	0.10	0.13	0.11
7	√	√			√	MAE	168.69	209.79	436.39	180.42	244.39	201.39
						WMAPE	0.09	0.11	0.23	0.10	0.13	0.11
8	√		√		√	MAE	165.72	193.54	433.36	**161.22**	239.95	194.77
						WMAPE	0.09	0.10	0.23	**0.09**	0.13	0.10
9	√	√		√	√	MAE	175.10	205.36	457.85	181.29	249.93	206.05
						WMAPE	0.09	0.11	0.24	0.10	0.13	0.11
10	√	√	√	√	√	MAE	163.71	188.71	434.12	172.10	237.70	194.35
						WMAPE	0.09	0.10	0.23	0.09	0.13	0.10



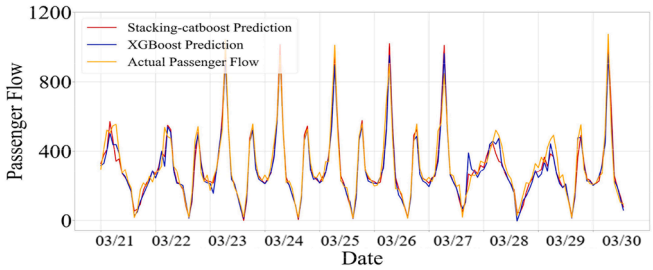
(a)



(b)



(c)



(d)

Fig. 9. Predicted and actual passenger flow of Stacking-Catboost and XGBoost: (a) commercial area; (b) residential area; (c) dining area; (d) scenic area.

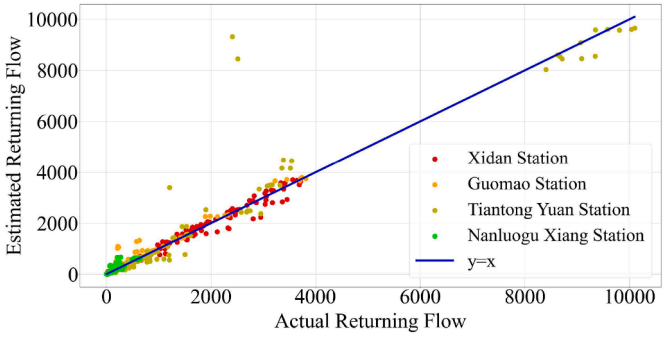


Fig. 10. Comparison between actual and estimated returning passenger flow.

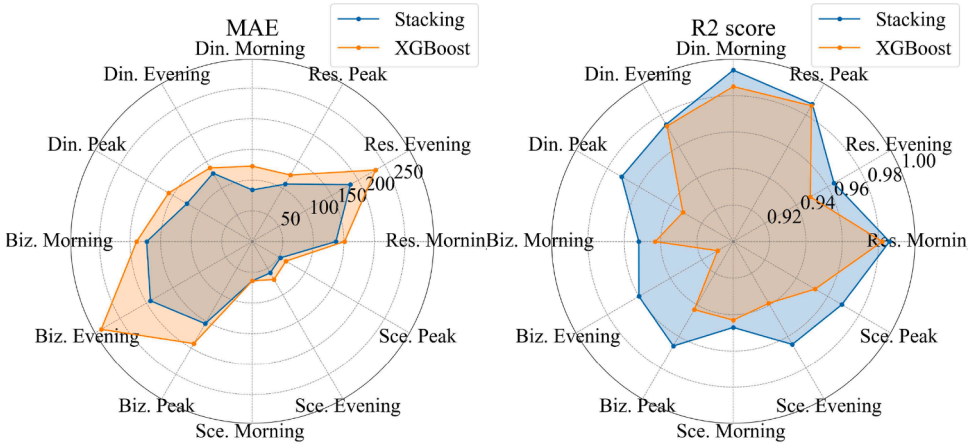


Fig. 11. Radar plot of XGBoost and Stacking during the morning, evening, and non-peak hours. (Res.: the residential area, Biz.: the commercial area, Sce.: the scenic area, Din.: the dining area).

evening peak, and non-peak hours. Since the metro is often used for commuting, its passenger flow exhibits distinct characteristics during rush hours, making the prediction of these periods crucial. Fig. 11 presents radar plots comparing the performance of XGBoost and Stacking during morning, evening, and non-peak hours. The yellow range depicting the MAE of the stacking strategy consistently falls within the blue range of XGBoost, while the yellow range describing the R2 score of the stacking strategy surpasses the blue range of XGBoost, indicating that the stacking strategy outperforms XGBoost for any station type in all time periods.

4.3. Features ranking and interpretation

Although past studies have focused on predicting passenger flow, they have neglected to interpret the output of machine learning models. To address this gap in research, our study utilizes explainable artificial intelligence through the SHapley Additive exPlanation (SHAP) method to provide an intuitive interpretation of our model's output (Lundberg & Lee, 2017). By employing SHAP, not only can we gain insights into the model itself, but we can also rank the features according to their importance. Fig. 12 shows the global importance ranking of all features to XGBoost, CatBoost, and Stacking methods in commercial and residential areas. The vertical axis represents the feature, while the horizontal axis shows the SHAP value. Each point represents the feature value, with color indicating its characteristic values.

The significance of features varies across different functional areas. In particular, returning passenger flow $r_{s,t+1}$ is a crucial feature, especially in the residential area. Conversely, near-time step $y_{s,t}$ and timetable $N_{s,t+1}$ hold considerable weight in the commercial area. Notably, different models prioritize distinct features. For instance, for predicting inbound passenger flow in the commercial area, the most influential characteristics are near-time steps $y_{s,t}$ (step=1 and step=2), timetable, and returning passenger flow. For the residential area, the returning passenger flow, near-time steps (step=1 and step=3), and timetable have the greatest impact on results. It is worth noting that Fig. 12 depicts global importance rather than individual importance, implying that some features, such as vehicular and financial services, may not be important for most groups but could be crucial for specific subsets.

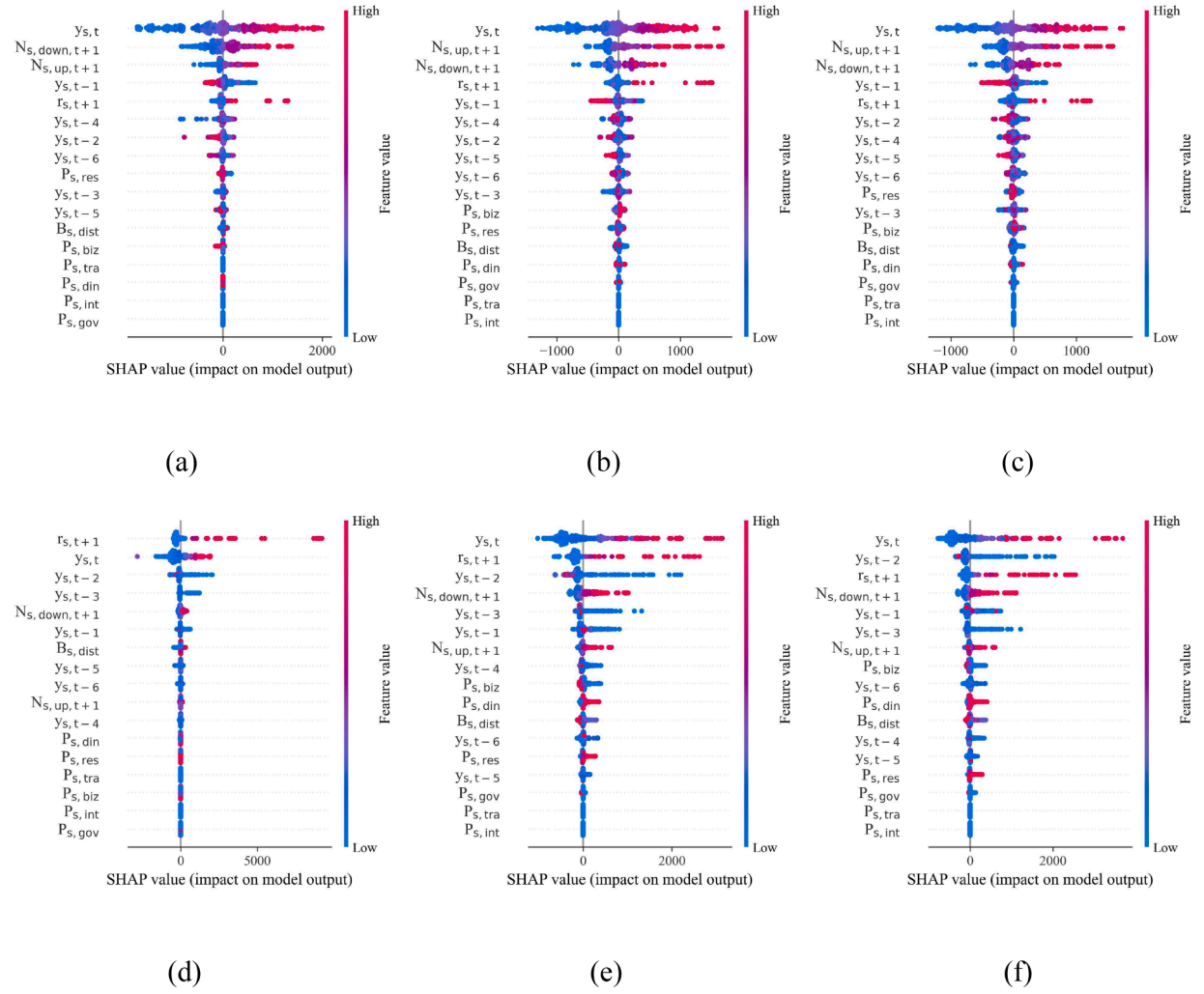


Fig. 12. SHAP summary plot of global feature importance:(a) XGBoost, commercial; (b) CatBoost, commercial; (c) Stacking, commercial; (d) XGBoost, residential; (e) CatBoost, residential; (f) Stacking, residential.

5. Conclusion

This paper concentrates on predicting the short-term passenger flow in the metro system, which creatively adds the returning passenger flow into the prediction model to describe the regularity of individual travel behavior and proposes a method of return probability parallelogram to estimate returning passengers. To better characterize the macroscopic spatiotemporal travel patterns other than the micro individual travel behavior, the relevant variables such as train operation characteristics, the close-by bus stations, and points-of-interest data are integrated into the model. Machine learning algorithms can analyze the relationship between passenger flow and features, learning the patterns and regularities hidden within the data. The contribution of these variables is explained and quantified by model interpretability analysis. By stacking Random Forest (RF), Gradient Boosting Decision Tree (GBDT), Light Gradient Boosting Machine (LightGBM), Extreme Gradient Boosting (XGBoost), and CategoricalBoosting (CatBoost) algorithms, the proposed model improves prediction accuracy. Various experiments are conducted to evaluate the proposed method with a real-world metro dataset in Beijing spatially and temporally. The experiments conducted with real-world metro data in Beijing show that the Stacking-Catboost model effectively considers returning passenger flow and multiple related features. The global interpretability of models reveals different feature importance levels, with returning passenger flow, historical time step, and timetable being particularly significant factors.

This study takes into account both macroscopic spatiotemporal travel patterns and micro individual travel behavior to predict metro passenger flow. Further study can utilize various data sources such as metro video data and individual mobile communication data to further investigate individual behaviors. The travel purpose estimation can be integrated into the inbound passenger flow prediction to improve passenger flow prediction accuracy. However, there are challenges associated with incomplete information, strong privacy concerns, and limited accessibility in obtaining the data.

CRediT authorship contribution statement

Jiarui Yu: Methodology, Software, Formal analysis, Writing – original draft, Visualization. **Ximing Chang:** Conceptualization, Data curation, Methodology, Writing – review & editing. **Songhua Hu:** Writing – review & editing, Supervision. **Haodong Yin:** Project administration, Supervision. **Jianjun Wu:** Methodology, Writing – review & editing, Supervision.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Nos. 72171020, 72301021), Beijing Natural Science Foundation (L221006), and the 111 Project (No. B20071).

References

Arana, P., Cabezudo, S., & Peñalba, M. (2014). Influence of weather conditions on transit ridership: A statistical study using data from smartcards. *Transportation Research Part A: Policy and Practice, 59*, 1–12.

Bao, J., Kang, J., Yang, Z., & Chen, X. (2022). Forecasting network-wide multi-step metro ridership with an attention-weighted multi-view graph to sequence learning approach. *Expert Systems With Applications, 210*, Article 118475.

Cantelmo, G., & Viti, F. (2019). Incorporating activity duration and scheduling utility into equilibrium-based dynamic traffic assignment. *Transportation Research Part B: Methodological, 126*, 365–390.

Cantelmo, G., Qurashi, M., Prakash, A., Antoniou, C., & Viti, F. (2020). Incorporating trip chaining within online demand estimation. *Transportation Research Part B: Methodological, 132*, 171–187.

Cao, Y., Hou, X., & Chen, N. (2022). Short-term forecast of OD passenger flow based on ensemble empirical mode decomposition. *Sustainability, 14*, 8562.

Chan, K., Dillon, T., Singh, J., & Chang, E. (2012b). Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and Levenberg-Marquardt algorithm. *IEEE Transactions on Intelligent Transportation Systems, 13*(2), 644–654.

Chang, H., Lee, Y., Yoon, B., & Baek, S. (2012). Dynamic near-term traffic flow prediction: System-oriented approach based on past experiences. *IET Intelligent Transport Systems, 6*(3), 292–305.

Chang, X., Wu, J., Liu, H., Yan, X., Sun, H., & Qu, Y. (2019). Travel mode choice: A data fusion model using machine learning methods and evidence from travel diary survey data. *Transportmetrica A: Transport Science, 15*(2), 1587–1612.

Chang, X., Wu, J., Yu, J., Liu, T., Yan, X., & Lee, D. (2024). Addressing COVID-induced changes in spatiotemporal travel mobility and community structure utilizing trip data: An innovative graph-based deep learning approach. *Transportation Research Part A: Policy and Practice, 180*, 103973.

Chang, X., Wu, J., Gonçalo, H., Sun, H., & Feng, Z. (2022). A cooperative strategy for optimizing vehicle relocations and staff movements in cities where several carsharing companies operate simultaneously. *Transportation Research Part E: Logistics and Transportation Review, 161*, 102711.

Chang, X., Wu, J., Sun, H., & Yan, X. (2023). A Smart Predict-then-Optimize method for dynamic green bike relocation in the free-floating system. *Transportation Research Part C: Emerging Technologies, 153*, 104220.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the KDD'16: 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794).

Cheng, C., Tsai, M., & Cheng, Y. (2022). An intelligent time-series model for forecasting bus passengers based on smartcard data. *Applied Sciences, 12*(9), 4763.

China Association of Metros. (2022). *The statistics and analysis report of urban metro system in 2021*. https://www.camet.org.cn/tjxx/9944.

Habtemichael, F., & Cetin, M. (2016). Short-term traffic flow rate forecasting based on identifying similar traffic patterns. *Transportation Research Part C: Emerging Technologies, 66*, 61–78.

Huang, H., Mao, J., Lu, W., Hu, G., & Liu, L. (2023). DEASeq2Seq: An attention based sequence to sequence model for short-term metro passenger flow prediction within decomposition-ensemble strategy. *Transportation Research Part C: Emerging Technologies, 146*, Article 103965.

Jiang, X., Zhang, L., & Chen, X. (2014). Short-term forecasting of high-speed rail demand: A hybrid approach combining ensemble empirical mode decomposition and gray support vector machine with real-world applications in China. *Transportation Research Part C: Emerging Technologies, 44*, 110–127.

Lei, X., Mohamad, U., Sarlan, A., Shutaywi, M., Daradkeh, Y., & Mohammed, H. (2022). Development of an intelligent information system for financial analysis depend on supervised machine learning algorithms. *Information Processing & Management, 59*(5), Article 103036.

Li, T., Wang, B., Zhou, M., & Watada, J. (2018). Short-term load forecasting using optimized LSTM networks based on EMD. *International Conference on Communications, Circuits and Systems, 10*, 84–88.

Liu, Y., Liu, Z., & Jia, R. (2019). DeepPF: A deep learning based architecture for metro passenger flow prediction. *Transportation Research Part C: Emerging Technologies, 101*, 18–34.

Lundberg, S., & Lee, S. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 4765–4774.

Ma, X., Zhang, J., Du, B., Ding, C., & Sun, L. (2019). Parallel architecture of convolutional bi-directional LSTM neural networks for network-wide metro ridership prediction. *IEEE Transactions on Intelligent Transportation Systems, 20*(6), 2278–2288.

Milenkovic, M., Svadlenka, L., Melichar, V., Bojovic, N., & Avramović, Z. (2018). SARIMA modelling approach for railway passenger flow forecasting. *Transport, 33*(5), 1113–1120.

Ming, W., Bao, Y., Hu, Z., & Xiong, T. (2014). Multistep-ahead air passenger traffic prediction with hybrid ARIMA-SVMs models. *The Scientific World Journal, 55*, 567246.

Roos, J., Gavin, G., & Bonnevay, S. (2017). A dynamic Bayesian network approach to forecast short-term urban rail passenger flows with incomplete data. *Transportation Research Procedia, 26*, 53–61.

Shi, Z., Zhang, N., Schonfeld, P., & Zhang, J. (2020). Short-term metro passenger flow forecasting using ensemble-chaos support vector regression. *Transportmetrica A Transport Science, 16*(2), 194–212.

Shi, R., Xu, X., Li, J., & Li, Y. (2021). Prediction and analysis of train arrival delay based on XGBoost and Bayesian optimization. *Applied Soft Computing, 109*, Article 107538.

Sun, L., Axhausen, K., Lee, D., & Huang, X. (2013). Understanding metropolitan patterns of daily encounters. *Proceedings of the National Academy of Sciences, 110*, 13774–13779.

Wang, L., & Zhang, W. (2023). A qualitatively analyzable two-stage ensemble model based on machine learning for credit risk early warning: Evidence from Chinese manufacturing companies. *Information Processing & Management, 60*(3), Article 103267.

Wei, Y., & Chen, M. (2012). Forecasting the short-term metro passenger flow with empirical mode decomposition and neural networks. *Transportation Research Part C: Emerging Technologies, 21*, 148–162.

Wen, K., Zhao, G., He, B., & Ma, J. (2022). Hongxiang Zhang, A decomposition-based forecasting method with transfer learning for railway short-term passenger flow in holidays. *Expert Systems with Applications, 189*, Article 116102.

Williams, B. (2001). Multivariate vehicular traffic flow prediction: Evaluation of ARIMAX modeling. *Transportation Research Record, 1776*(1), 194–200.

Xue, G., Liu, S., Ren, L., Ma, Y., & Gong, D. (2022). Forecasting the subway passenger flow under event occurrences with multivariate disturbances. *Expert Systems with Applications, 188*, Article 116057.

Yang, X., Xue, Q., Yang, X., Yin, H., Qu, Y., Li, X., et al. (2021). A novel prediction model for the inbound passenger flow of urban rail transit. *Information Sciences, 566*, 347–363.

Zhang, Z., Wang, C., Gao, Y., Chen, J., Zhang, Y. (2020). Short-term passenger flow forecast of metro system station based on MIC feature selection and ST-LightGBM considering transfer passenger flow. Scientific Programming, 1–15.

Zhang, Z., Wu, C., Qu, S., & Chen, X. (2022). An explainable artificial intelligence approach for financial distress prediction. *Information Processing & Management, 59* (4), Article 102988.