Study on real-time prediction model of railway passenger flow based on big data technology

Yiyi Yin, *Yong* Zhang, *Zhengzheng* Wei*, and *Xiang* Zhao

Institute of Computing Technologies, China Academy of Railway Sciences, Beijing

Abstract. In order to solve the limitation of traditional offline forecasting application scenarios, the author uses a variety of big data open source frameworks and tools to combine with railway real-time data, and proposes a real-time prediction model of railway passenger flow. The model architecture is divided into four levels from bottom to top: data source layer, data transmission layer, prediction calculation layer and application layer. The main components of the model are data flow and prediction flow. Through message queue and ETL, the data process part realizes the synchronization of offline data and real-time data; through the big data technology frameworks such as Spark, Redis and Hive and the GBDT (Gradient Boosting Tree) algorithm, the prediction process partially realizes the real-time passenger flow of the train OD section prediction. The experimental results show that the model proposed by the author has certain practicability and accuracy both in performance and prediction accuracy.

Keywords: Big data technology, Real-time prediction model, Spark, GBDT.

1 Introduction

The prediction of railway passenger flow is an effective means to reasonably allocate passenger transport resources and alleviate the contradiction between passenger travel demand and limited transport capacity. In the field of railway passenger flow prediction, whether it is based on statistical analysis, machine learning, deep learning and other model research, the research direction of scientists is mainly focused on offline passenger flow forecasting[1-3], that is, the model uses offline traffic data to build training set and test set with days as the maximum data granularity, However, the influence of the real-time dynamic change of the delivery volume on the final delivery volume in the pre-sale period is not considered. However, for dynamic ticket pre-assignment or capacity control[4], this depends on the real-time business scenario of passenger flow. Due to the limitation of data granularity and real-time performance, the application of offline prediction model will be limited. Therefore, the author believes that if the real-time prediction model is constructed

* Corresponding author: ninna.weina@163.com

according to the dynamic changes of ticket sales during the pre-sale period, it can provide some auxiliary decision support for the above two business scenarios.

In other transportation fields, the research on real-time passenger flow prediction mainly involves the innovation of prediction model or algorithm[5-7], and rarely involves the research on prediction performance or feasibility. With the gradual improvement of passenger transport marketing system of China Railway and Railway Administration, Railway passenger transport business data presents the characteristics of rich business scenarios, large amount of data and high real-time. In the process of real-time prediction, how to make full, reasonable and effective use of these data has become the key to build a high availability real-time prediction model.

In view of the above problems, the author puts forward a real-time prediction model of railway passenger flow. The model is based on big data open source architectures such as Spark, message queue and distributed memory database, based on the business data such as ticket records data and remaining tickets to make real-time predictions of the number of trains in the pre-sale period.

2 Model architecture

The real-time prediction model architecture is shown in Figure 1. In terms of function, the system is divided into four layers: data source layer, data transmission layer, prediction calculation layer and application layer.

① Data source layer. The data source layer includes the real-time and offline databases involved in the data center of the China Railway and the data center of the Railway Administration.

② Data transmission layer. Data transmission layer includes message middleware Rocket MQ and ETL tools. Among them, Rocket MQ is mainly used for real-time data transmission; ETL tools such as Sqoop and Kettle are mainly used for data source layer persistent data transmission and statistics.

③ Prediction calculation layer. It consists of two modules: prediction and monitoring. The prediction module includes Spark computing framework, Redis, Hive and GBase. The monitoring module includes workflow scheduling framework Easy Scheduler and real-time message queue transmission monitoring, it is mainly used to monitor the execution of workflow scripts and the loading of real-time data statistics

④ Application layer. The application layer mainly applies the real-time prediction results to railway passenger transport marketing for decision support, including pre-assignment, capacity control and other business scenarios.

3 Data flow

Figure 2 shows the data flow of the real-time prediction model. Data flow is mainly divided into two parts: data transmission and data processing and calculation.

3.1 Data Transmission

As mentioned above, the data involved in the model is divided into two parts: real-time data and offline data. As shown in the Figure 2, the real-time data consists of the number of remaining tickets counted by the Internet Ticketing data center China Railway and the ticket records data counted by the ticket sales centers of 18 Railway Administration. Offline data is composed of ticket records data and remaining tickets data collected by marketing

center of China Railway. The ticket records data is the record of ticket sale, refund, void and change in railway ticketing system.

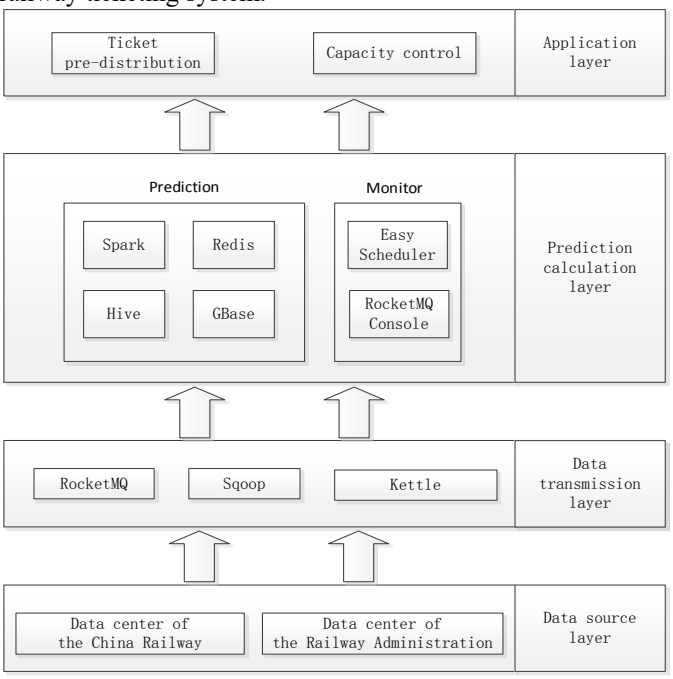

Fig. 1. Model architecture.

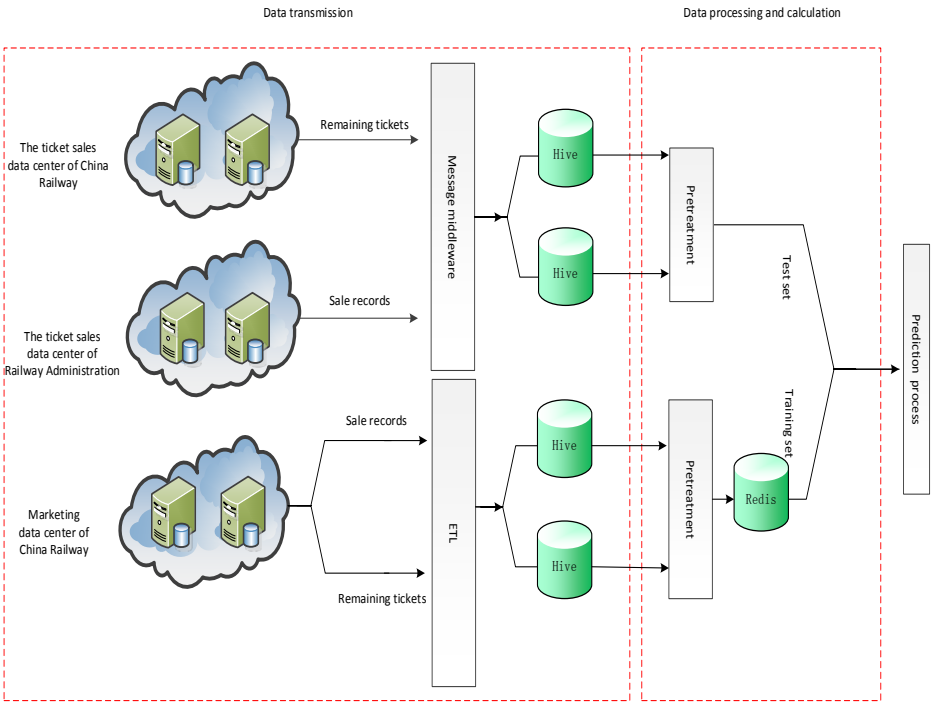

Fig. 2. Data flow.

In the data transmission phase, the model obtains the train date, train code, departure station, arrival station, selling date and selling time from the ticket records data (This field is the operation time for ticket refund, void and change) and the number of tickets (The value of this field is negative for ticket refund, void and change). The remaining tickets data is the record of the number of tickets left during the pre-sale period.In the data transmission stage, the system obtains the train code, train departure date, departure station, arrival station, remaining tickets statistics date, remaining tickets statistics time and remaining ticket quantity from the remaining ticket data.

These data are synchronously copied to Hive through message middleware Rocket MQ and ETL. For real-time data, the related data center of the source data layer generates the SQL statements of ticket records data and remaining tickets and puts them into the message queue producer, the message queue consumes these SQL statements according to the set time interval, and updates or inserts the relevant data into Hive. For offline data, the ticket records data and remaining tickets data of 24 points of train statistics are synchronized from the marketing data center of China Railway by ELT tool and according to the departure date.

3.2 Data processing and calculation

As shown in Figure 2, the real-time prediction model preprocesses the real-time data and offline data stored in hive to build test set and training set that can be used by the prediction model. In this stage, the records data in hive is processed as follows:

According to the time interval of real-time data acquisition setted by model, segment the offline record data by ticket selling time.

Organize the data. The total number of tickets sold minus the number of refund, void and change,to get the net sales in each period.

Supplement missing records in time intervals. When the ticket records data splits the ticket selling time according to the time interval set by the model, the sorting results will be discontinuous, That is to say, there is no one of the four events of sale, refund ,void and change in any OD section of a train during this period, the current ticket marketing business data statistics will not produce records, therefore, in order to ensure the consistency and consistency of the samples in the training set, it is necessary to fill zero for the missing stub data.

Statistics of the cumulative number of tickets sold in different periods. Since the ultimate goal of the model prediction is the final delivery volume of the train od interval, it is necessary to accumulate the net sales in different periods.

Count the hours between the tickets selling time and the train departure time.

After the above processing and calculation, for a certain od interval of a train, the offline data is used to build a training set with the cumulative amount of net ticket sales, the number of hours from departure and the number of remaining tickets as the characteristic variables, and the final sending amount as the target variable. For the same training set with the same prediction date, train number and OD, it is stored in distributed memory database Redis in order to improve the prediction speed. In the next time interval prediction, it is not necessary to recalculate the training set, but directly read from Redis, which can effectively improve the calculation speed of prediction.

In addition, for non-holiday prediction, the model selects the ticket sales in different time periods from 60 days before the departure date (excluding holidays) to yesterday as the training set; For the prediction of holidays, the model selects the ticket sales of the same holidays and the same train in the pre-sale period in the past five years as the training set.

Similarly, the real-time data is used to build a test set takes the cumulative amount of net tickets sold up to the current time, the number of hours to departure, and the number of remaining tickets as the characteristic variables.

4 Prediction model

4.1 Prediction process based on spark framework

The prediction part of the real-time prediction model is based on the distributed memory computing framework Spark[8].It can synchronize the data loaded in the data source storage to the memory, through the abstraction of data and a series of operator processing. Finally, the calculation results are output to memory in abstract form or persistently written to other storage media in the form of data stream.

The real-time passenger flow prediction needs to meet the application requirements of large amount of calculation, frequent calculation, and calculation results in a short time. Firstly, the training set and test set after data processing are abstracted into the form of distributed data frame in spark, which is a distributed data set based on RDD (elastic distributed data set). Then, partition according to a basic attribute of the prediction object as a key. Partition is a computing unit of parallel computing in DataFrame. DataFrame is logically divided into multiple partitions, and each partition is called partition. The partition method makes full use of the computing resources of the cluster, reduces the network transmission and improves the performance of real-time prediction. Finally, the local machine learning algorithm uses the training set and test set in each partition to construct an independent prediction task, which is submitted to the server in the cluster for calculation.

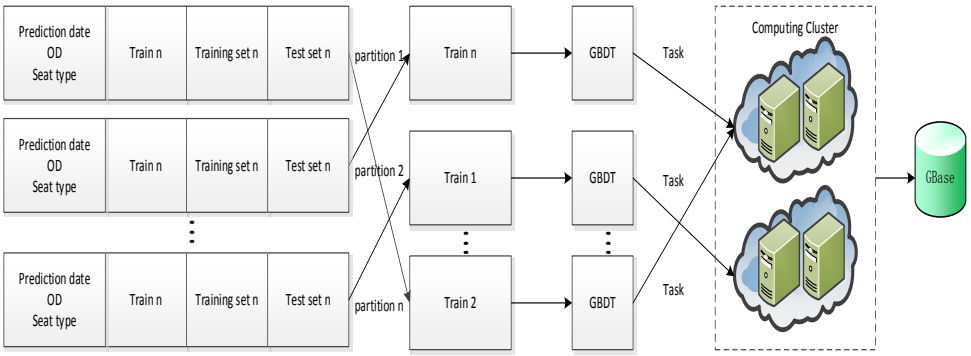

Fig. 3. Prediction process.

Figure 3 is a flow chart of the real-time prediction model. It can be seen from the figure that the prediction task of passenger flow in OD section of each train is independent of each other. Therefore, the model is divided according to different train, which are allocated to the computing cluster. Each partition calls machine learning algorithm GBDT for prediction. Finally, the prediction results are written into GBase database.

4.2 Prediction algorithm

GBDT [9] , which uses Boosting promotion idea, promotes multiple weak classifiers to a strong learning classifier, belongs to the category of integrated learning. For GBDT algorithm, the weak classifier is composed of multiple decision trees, as shown in Equation (4-1) below, Suppose there are decision trees, then the prediction result of each tree is:

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

where, $f_k(x;\theta_k)$ is the decision tree and θ is the parameter of the decision tree.

It can be seen that Equation (4-1) is an additive model, in order to approach the optimization target step by step, the learning process is shown in Equation (4-2)

$$\begin{gathered}\hat{y}_i^0 = 0 \\ \hat{y}_i^1 = f_1(x_i;\theta_i) = y_i^0 + f_1(x_i;\theta_i) \\ \hat{y}_i^2 = f_1(x_i;\theta_i) + f_2(x_i;\theta_i) = \hat{y}_i^1 + f_2(x_i;\theta_i) \\ \hat{y}_i^t = \sum_{k=1}^{t} f_k(x_i;\theta_i) = \hat{y}_i^{t-1} + f_t(x_i)\end{gathered} \tag{2}$$

The loss function of each iteration is shown in Equation (4-3) (Here, take the square loss function as an example, it can also be exponential loss function and other loss functions)

$$L(y_i, \hat{y}_i^t) = (y_i - \hat{y}_i^t)^2 \tag{3}$$

In order to simplify the optimization process and improve the optimization speed, In GBDT algorithm, The value obtained by using the negative gradient of the loss function is regarded as the approximation of residual, and is also called pseudo residual. As shown in Equation (4-4) below,

$$r_{ki} = -\left[\frac{\partial L(y_i, \hat{y}_i)}{\partial \hat{y}_i}\right] \tag{4}$$

The above pseudo residual value is used to fit a regression tree, to get leaf node region of the regression tree $R_{kj}(0 \le j \le J)$, J is the number of leaf nodes. Use linear search to recalculate the output of the second tree as shown in Equation (4-5)

$$\gamma_{jk} = \arg\min_{\gamma} \sum L(y_i, \hat{y}_i^{t-1} + \gamma_t) \tag{5}$$

And sum these output values to update the predicted values (as shown in Equation (4-6))

$$\hat{y}_i^t = \hat{y}_i^{t-1} + \sum \gamma_{jk} I \tag{6}$$

In Equation (4-6), I is an indicator function to judge whether the sample point is in the corresponding area of the leaf node of the tree. When the loss error reaches the specified threshold, the lifting tree can be obtained.

According to the above description of GBDT algorithm process, it can be seen that, the algorithm uses the negative gradient of the loss function as the residual to construct the regression tree, to a certain extent, over fitting is prevented. At the same time, the negative gradient can be used to calculate a variety of loss functions, so that the algorithm has strong robustness.

There are two main reasons for using GBDT algorithm in the model:

GBDT is a non-linear model, which can better fit the characteristics of passenger flow whose cycle law is not obvious.

Since there are few features in the training set of the input model, GBDT can quickly consider all features in each partition without feature selection.

5 Experiment and result analysis

The author built a cluster of 8 Linux servers in the ticketing network. Each server has 32/64 bit processors and 64GB memory. At the same time, we deployed Spark2, Hive, Redis, RocketMQ, Kettle, Maven and other related open source frameworks and tools in the cluster, and developed the prediction model using Scala language.

In this section, the high-speed trains of Nan-Guang passenger dedicated line and Jing-Hu High-speed line are taken as the experimental objects. The prediction date is from August 1, 2020 to August 31, 2020. During this period, there were 66 trains and 518 OD sections on the two lines. For each train, the model predicts from 72 hours to 6 hours before the train leave, and the frequency is 1 hour.

Figure 4 shows the cumulative time consumed by the model at each prediction date. It can be seen that the time consumed by the data flow is greater than the prediction time. According to the statistics, the average cumulative consumption time of the model is 3 minutes and 86 seconds. Among them, the average time of prediction process consumption is 1 minute 77 seconds, and the time of data process consumption is 2 minutes 9 seconds.

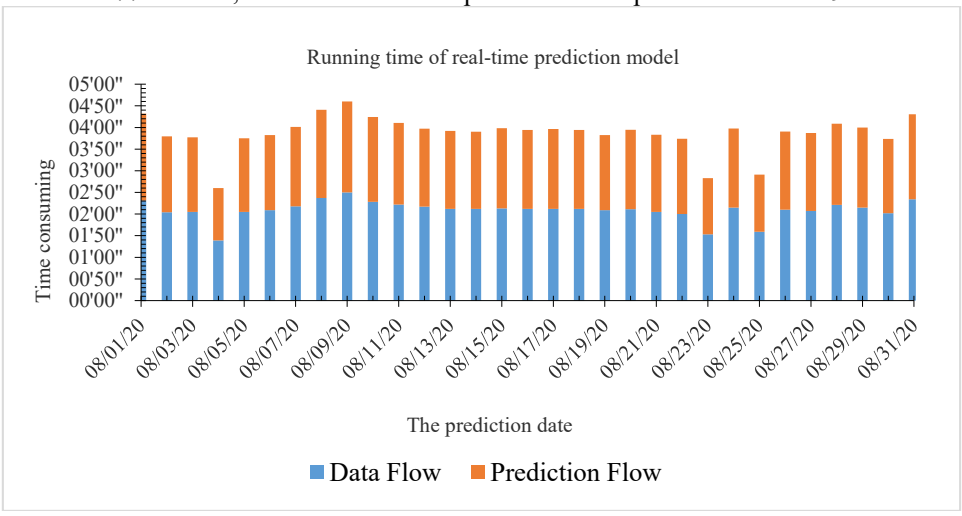

Fig. 4. The running time of real-time prediction model.

For a given prediction date, Figure 5 shows the time consumed by the model to predict the train distance per hour. By statistics, the average time consumed by the model is 1 minute and 35 seconds per hour. The average time consumed by data flow is 95 seconds, and the time consumed by prediction flow is 40 seconds.

Figure 4 and Figure 5 statistics the time consumed by the model to predict passenger flow from the perspective of two data granularity. According to the experimental results, it can be seen that the real-time prediction model is feasible in the actual application scenarios. The model can calculate the results in a short time, which ensures the timeliness of the prediction results of other services.

In order to verify the accuracy of the prediction results, this paper compares the results of the real-time prediction model with the offline prediction using the same prediction algorithm and training set selection strategy. The maximum prediction time granularity of offline prediction model is days. Therefore, the prediction percentage errors of 72 hours, 48 hours and 24 hours of the real-time prediction model are compared with the prediction percentage errors of 3 days, 2 days and 1 day of the off-line prediction model. The results are shown in figures 6 to 8.

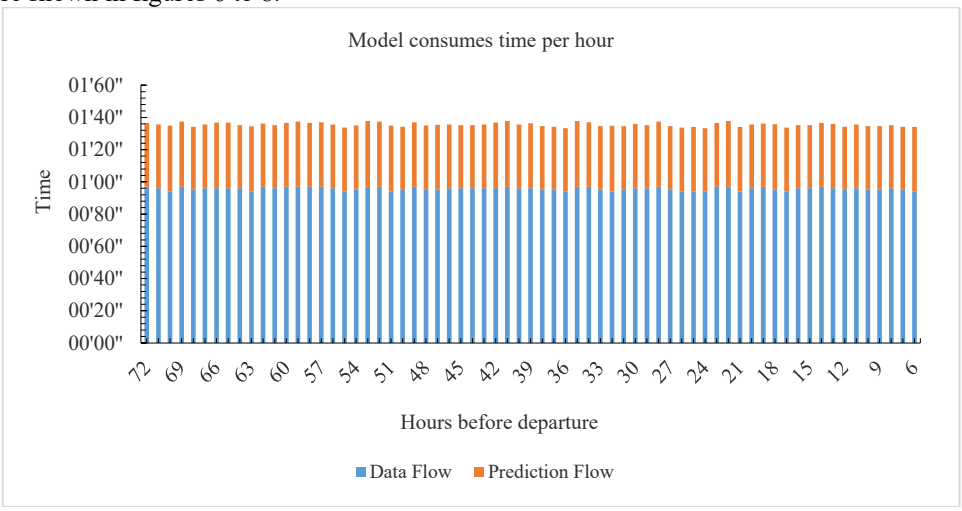

Fig. 5. The time consumption per hour of the model.

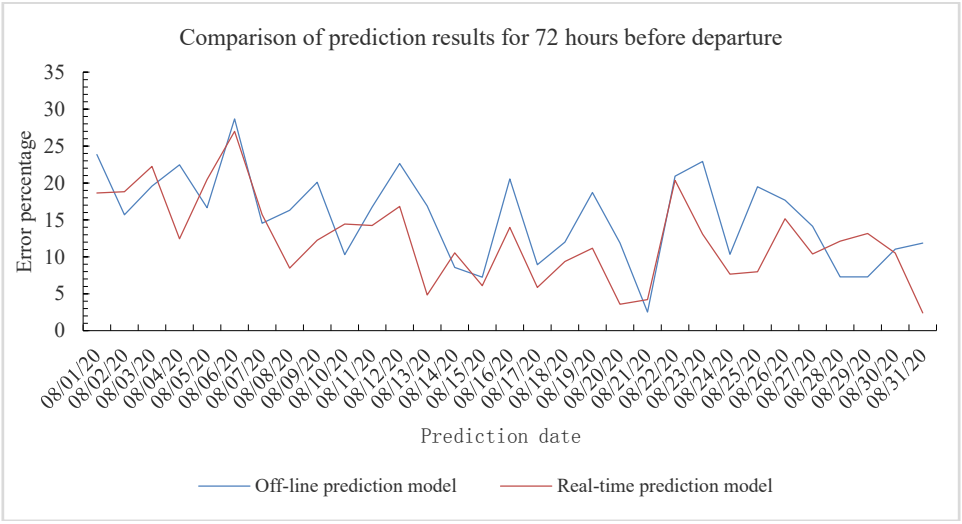

Fig. 6. Comparison of prediction results 72 hours before departure.

It can be seen from the results that under the same conditions, the prediction accuracy of the real-time prediction model is slightly better than that of the offline prediction model. At the same time, with the approaching of driving time, the real-time prediction results are closer to the real value.

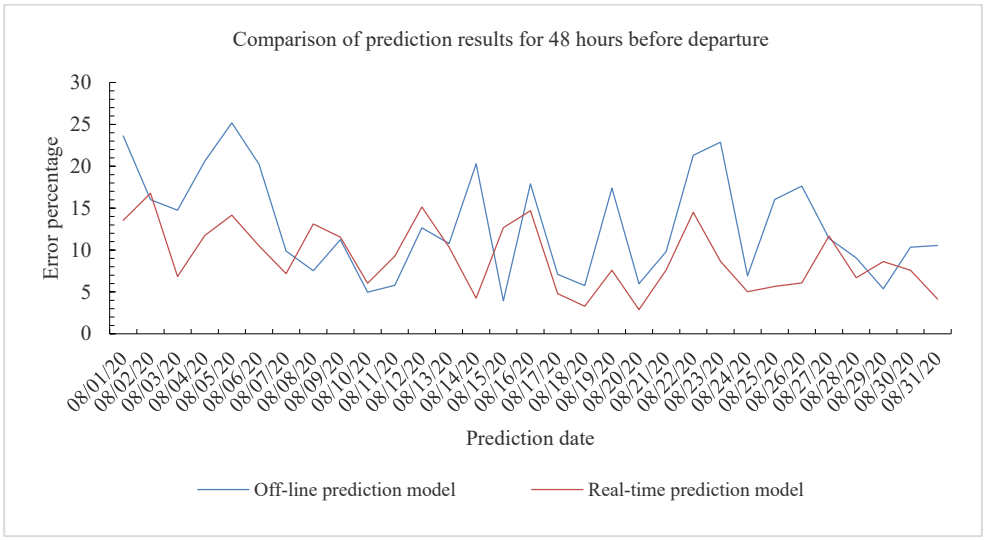

Fig. 7. Comparison of prediction results 48 hours before departure.

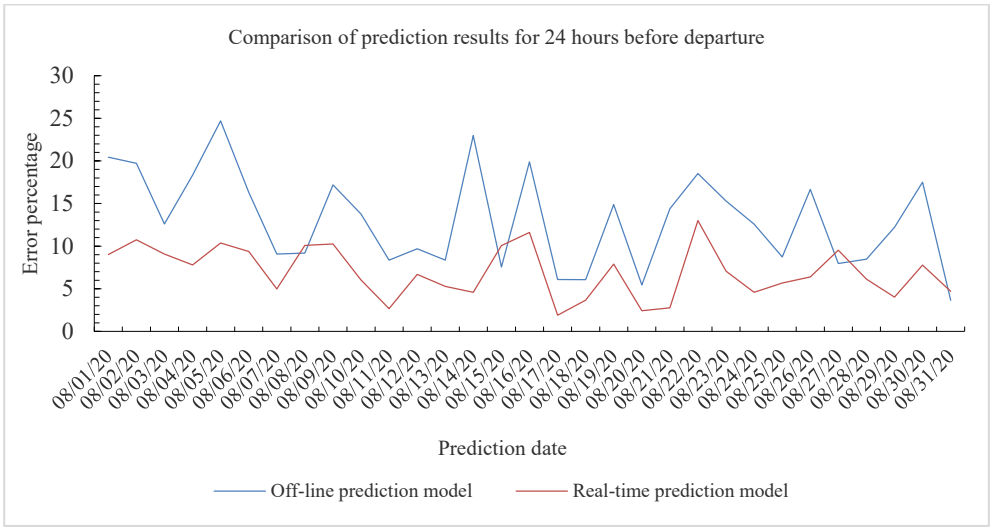

Fig. 8. Comparison of prediction results 24 hours before departure.

6 Conclusion and prospect

In this paper, the author introduces the real-time passenger flow forecasting model from the overall structure, data flow and forecasting process. According to the evaluation of the model performance and results, it can be seen that, the real-time prediction model has the feasibility and decision support ability to be applied to actual business scenarios. With the development and improvement of Internet Ticketing, extended services and e-ticket systems, In the next step of the study, the author will consider the real-time passenger flow forecast for different components of passenger flow or ticket demand and other multi business scenarios [10].

Fund projects: Youth Project of scientific research and development fund of China Academy of Railway Sciences Group Corporation (2019YJ120), Scientific research and development project of China Railway Group Corporation (K2019X022)

References

1. LI Jie,PENG Qi-yuan, YANG Yu-xiang. Passenger Flow Prediction for Guangzhou-Zhuhai Intercity Railway[J].JOURNAL OF SOUTHWEST JIAOTONG UNIVERSITY,2020(1):41-51.Based on SARIMA Model.
2. LI Li-hui, ZHU Jian-sheng, XU Yan,etl. Study on Forecast Methods of Passenger Flow Spatial Distribution for Beijing-Shanghai High-speed Railway[J].RAILWAY TRANSPORT AND ECONOMY, 2017, 039(006):32-36.
3. ZHANG Jin,XU Jun-xiang,GUO Jing-ni. Forecasting method of induced passenger flow
for Sichuan-Tibet railway based on improved MD model[J]. Journal of Southeast University (English Edition), 2020, v.36(01):101-109.
4. SONG Wen-bo, ZHAO Peng, LI Bo. High-speed Railway Ticket Allocation Considering Dynamic Ticket Booking Demand[J].JOURNAL OF THE CHINA RAILWAY SOCIETY,2019,041(009):20-27.
5. XU Sheng-bo.Real-time Prediction of Subway Passenger Travel Destinations Based on Automated Fare Collection Data[J].Journal of Transportation Engineering and Information,2019, 017(002):81-90.
6. HUAN Ning, XIE Qiao, YE Hong-xia,etl. Real-time Forecasting of Urban Rail Transit Ridership at the Station Level Based on Improved KNN Algorithm[J].Journal of Transportation Systems Engineering and Information Technology, 2018, 18(05):121-128.
7. YAO Enjian, ZHOU Wenhua, ZHANG Yongsheng. Real-time Forecasting of Entrance and Exit Passenger Folw for Newly Opened Station of Urban Rail Transit at Initial Stage[J]. China Railway Science ,2018,39(02):119-127.
8. Zaharia M, Chowdhury M, Franklin M, Shenker S, Stoica I.Spark: Cluster Computing with Working Sets.HotCloud 2010.2010.
9. Friedman J. Additive logistic regression: a statistical view of boosting[J]. Ann. Statist, 2000, 28.
10. YE Yu-ling, WANG Yi-shi. Research on Travel Model Choice Behavior in ShangHai-GuangZhou Transport Corridor[J].JOURNAL OF THE CHINA RAILWAY SOCIETY,2010,32(4):13-17.