Received 25 April 2024
Revised 3 June 2024
Accepted 4 June 2024

Short-term train arrival delay prediction: a data-driven approach

Qingyun Fu
Postgraduate Department, China Academy of Railway Sciences, Beijing, China; Signal and Communication Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing, China and Traffic Management Laboratory for High-Speed Railway, National Engineering Research Center of System Technology for High-Speed Railway and Urban Rail Transit, Beijing, China
Shuxin Ding and Tao Zhang
Signal and Communication Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing, China and Traffic Management Laboratory for High-Speed Railway, National Engineering Research Center of System Technology for High-Speed Railway and Urban Rail Transit, Beijing, China
Rongsheng Wang
Scientific and Technological Information Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing, China
Ping Hu
Postgraduate Department, China Academy of Railway Sciences, Beijing, China and Yibin Track, Signal and Communication Depot, China Railway Chengdu Group Co., Ltd, Chengdu, China, and
Cunlai Pu
School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China

Abstract

Purpose – To optimize train operations, dispatchers currently rely on experience for quick adjustments when delays occur. However, delay predictions often involve imprecise shifts based on known delay times. Real-time and accurate train delay predictions, facilitated by data-driven neural network models, can significantly reduce dispatcher stress and improve adjustment plans. Leveraging current train operation data, these models enable swift and precise predictions, addressing challenges posed by train delays in high-speed rail networks during unforeseen events.

This work was supported in part by the National Natural Science Foundation of China under Grant 62203468, in part by the Technological Research and Development Program of China State Railway Group Co., Ltd. under Grant Q2023X011, in part by the Young Elite Scientist Sponsorship Program by China Association for Science and Technology (CAST) under Grant 2022QNRC001, in part by the Youth Talent Program Supported by China Railway Society, and in part by the Research Program of China Academy of Railway Sciences Corporation Limited under Grant 2023YJ112.

Railway Sciences
Vol. 3 No. 4, 2024
pp. 514-529
Emerald Publishing Limited
e-ISSN: 2755-0915
p-ISSN: 2755-0907
DOI 10.1108/RS-04-2024-0012

Design/methodology/approach – This paper proposes CBLA-net, a neural network architecture for predicting late arrival times. It combines CNN, Bi-LSTM, and attention mechanisms to extract features, handle time series data, and enhance information utilization. Trained on operational data from the Beijing-Tianjin line, it predicts the late arrival time of a target train at the next station using multidimensional input data from the target and preceding trains.

Findings – This study evaluates our model's predictive performance using two data approaches: one considering full data and another focusing only on late arrivals. Results show precise and rapid predictions. Training with full data achieves a MAE of approximately 0.54 minutes and a RMSE of 0.65 minutes, surpassing the model trained solely on delay data (MAE: is about 1.02 min, RMSE: is about 1.52 min). Despite superior overall performance with full data, the model excels at predicting delays exceeding 15 minutes when trained exclusively on late arrivals. For enhanced adaptability to real-world train operations, training with full data is recommended.

Originality/value – This paper introduces a novel neural network model, CBLA-net, for predicting train delay times. It innovatively compares and analyzes the model's performance using both full data and delay data formats. Additionally, the evaluation of the network's predictive capabilities considers different scenarios, providing a comprehensive demonstration of the model's predictive performance.

Keywords Train delay prediction, Intelligent dispatching command, Deep learning, Convolutional neural network, Long short-term memory, Attention mechanism

Paper type Research paper

1. Introduction

With China's impressive achievements in high-speed rail construction, high-speed railways excel in speed, safety, and comfort. However, external factors such as communication interruptions and severe weather causing train delays will severely affect the traffic management of railway operations (Huang, Wen, Fu, Peng, & Tang, 2020). Handling emergencies such as severe winds and foreign matter collisions in high-speed railways is a complex task that requires real-time, efficient, and safe management. For example, the parallel railway traffic management (RTM) system, through real-time interaction and closed-loop feedback between the physical RTM system and the actual RTM system, can dynamically evaluate and optimize rescheduling strategies, thereby improving the efficiency of emergency response (Zhou *et al.*, 2023).

Dispatchers need advanced dispatching strategies to enhance operational efficiency, making accurate train delay prediction crucial. Summarizing operational experiences, establishing effective delay prediction models, and studying delay propagation mechanisms are crucial steps in enabling swift responses to delays. This approach minimizes their adverse effects and ensures smooth high-speed rail operation.

Train delay prediction models can be broadly categorized into two main types: event-driven and data-driven (Spanninger, Trivella, Büchel, & Corman, 2022). The core idea behind event-driven approaches is to explicitly capture and model dependencies between events such as train arrivals, departures, and pass-throughs in the prediction function. It involves constructing a continuous training event chain or a network of dependent training events for the predicted time range. Representative event-driven models include Markov chains (Barta, Rizzoli, Salani, & Gambardella, 2012), graph models (Goverde, 2010), Bayesian networks (Zilko, Kurowicka, & Goverde, 2016), etc.

Data-driven approaches, on the other hand, primarily employ supervised learning. In this method, the input to the system includes historical observation data and actual values, with the actual values serving as learning labels. The system iteratively refines a predictive function, aiming to minimize the difference between the output and the actual values. This category encompasses techniques such as linear regression (Gorman, 2009), decision trees (Kecman & Goverde, 2015), random forests (Wang & Zhang, 2019), and neural networks (Oneto *et al.*, 2018). The increasing prominence of neural networks in this field is driven by their precision, simplicity, and real-time capabilities. Current neural network models for train delay prediction focus on effectively handling spatiotemporal sequence data and extracting features from multiple dimensions. In essence, the evolution of train delay prediction models emphasizes leveraging neural networks to handle complex spatiotemporal data and extract multidimensional features.

For example, Huang *et al.* (2020) developed a hybrid model that combines a three-dimensional convolutional neural network (CNN), long short-term memory (LSTM), and a fully connected neural network (FCNN), called CLF-Net. This innovative approach simultaneously considered static, temporal, and spatiotemporal data, providing a comprehensive method for predicting train delays. Other notable models, such as LLCF (Li, Huang, Wen, Jiang, & Rodrigues, 2022), with a one-dimensional convolutional neural network block for route-related variables, two LSTM networks for delay-related variables, and an FCNN block for environment-related variables, considered the detailed arrival/departure routes of trains and route conflicts. Heglund, Taleongpong, Hu, and Tran (2020) proposed a spatial-temporal graph convolutional network (STGCN) model, which employed a graph convolutional neural network, treating the routes traveled by trains as nodes and the stations at both ends of the route as edges. Node features represented the arrival delays through links, considering the impact of connections in the railway network on delay propagation. Additionally, Ding, Xu, Li, and Shi (2021) introduced a multi-layer time-series graph neural network (MTGNN) model, utilizing actual delays and infrastructure data of trains at previous stations, studying the prediction of delays caused by different reasons. Zhang *et al.* (2021) proposed a train spatio-temporal graph convolutional network (TSTGCN) model, which incorporated graph convolution with spatiotemporal attention mechanisms, taking as input recent time series, daily time series, and weekly time series to predict the cumulative effects of delays at each railway station. Xu, Li, and Ding (2022) proposed a dynamic spatio-temporal graph convolutional network (DB-STGCN) model, which employed a Bayesian combined graph convolutional network, handling variables related to the timetable, delay patterns, infrastructure, and weather. Dynamic causal relationships between features of train event delays were constructed, obtaining a feature causality graph as the input for graph convolution. The summary of each algorithm is presented in Table 1.

However, most models are trained using pure delay data without distinguishing the predictive performance for delays of different magnitudes. In the actual application, data includes a mixture of early arrivals and delays of various scales, necessitating further analysis and processing. Additionally, existing models typically consider the forward relationships of input time series, neglecting the bidirectional connections inherent in time sequences. Considering bidirectional relationships can better extract patterns between sequences.

In this paper, we propose a novel CBLA-net model consisting of CNN, bidirectional LSTM (Bi-LSTM), and an attention mechanism. The CNN is employed to extract feature information from the train operation data, forming a feature sequence for further processing. The bidirectional LSTM enhances the recognition capability of mutual relationships in train delay sequences, while the attention mechanism allows the model to differentiate the importance of information at different time steps for more accurate predictions. The main contributions of this study are in the following aspects:

(1) We proposed a novel network structure, CBLA-net, for predicting train arrival delays. The model integrates CNN, Bi-LSTM, and attention mechanisms, enabling it to extract spatiotemporal information from multiple trains' operations and their impact on delays.

(2) In terms of input data, we trained the model using both raw mixed early and delayed data and only extracted delayed data. We analyze the predictive performance for delays of different magnitudes.

Literature	Method	Input data	Characteristics
Huang, Spanninger, and Corman (2022)	CLF-Net (3DCNN, LSTM, FCNN)	Spatio-temporal features, timetable features (time-series), infrastructure (non-time-series)	It is the first time that static, temporal, and spatio-temporal data are simultaneously considered in a hybrid model
Li *et al.* (2022)	LLCF (CNN, two LSTM, FCNN)	Considers the arrival routes of predicted trains and route conflicts with forward trains	The detailed train arrival/departure routes are considered from a microscopic view in the proposed arrival delay prediction model
Heglund *et al.* (2020)	STGCN	A sequence of node features that are the arrival delay of trains passing through links	Consider the connections between elements in the rail network
Ding *et al.* (2021)	MTGNN	The actual delay and infrastructure data of trains at previous stations	Combines graph learning, graph convolution, and temporal convolution modules to predict train arrival delays under different causes
Zhang *et al.* (2021)	TSTGCN (SAtt, TAtt, GCN)	Recent time series, daily time series, weekly time series	Predict the total number of delayed trains in each railway station
Xu *et al.* (2022)	DB-STGCN (STGCN, DBN)	Timetable-related variables, delay pattern variables, infrastructure-related variables, and weather-related variables	Consider train delay patterns and dynamic interactions between train events, and study the dynamic causality of train delay propagation

Source(s): Author's own work

Table 1.
Recent literature review on neural networks in delay prediction

(3) We compared our proposed CBLA model with the CBL model, which consists of CNN and Bi-LSTM, and the CL model, which consists of CNN and LSTM. We found that the CBLA model has the best delay prediction performance, verifying that the Bi-LSTM and attention mechanisms in our proposed model contribute to improving the accuracy of delay prediction.

The remaining sections of this paper are organized as follows. Section 2 describes the train delay prediction problem. Section 3 introduces the overall structure of the proposed model and provides detailed descriptions of each module. In Section 4, we provide a detailed analysis of the numerical experimental results for the model under different performance metrics, including comparative experiments on the proposed model. Finally, in Section 5, we summarize the work presented in this paper and outline directions for future research.

2. Problem statement

Train delay prediction is an essential component of the railway system. In this paper, we employ a data-driven deep learning approach to forecast short-term (de Faverges, Russolillo, Picouleau, Merabet, & Houzel, 2018) train delays based on the train operational data at previous stations.

Short-term delay prediction is a model for predicting train delays considered from an operational level (Marković, Milinković, Tikhonov, & Schonfeld, 2015). It takes real-time data from train operations as input and predicts the arrival delay at upcoming stations online. This is crucial for real-time adjustments to dispatch plans.

The train delay prediction process conducted by our model is real-time. Once the train departs from the initial station, the prediction of arrival delay at the next station can be made using the operational information generated by the train at the preceding station. Therefore, the arrival delay time of any station except the starting station can be predicted. At the same time, our model considers the impact of multi-train operations and spatial variations on delays. The input information includes the operational details of both the target train and the preceding trains at previous stations, as well as the operational conditions at these stations. This makes the delay prediction information more comprehensive.

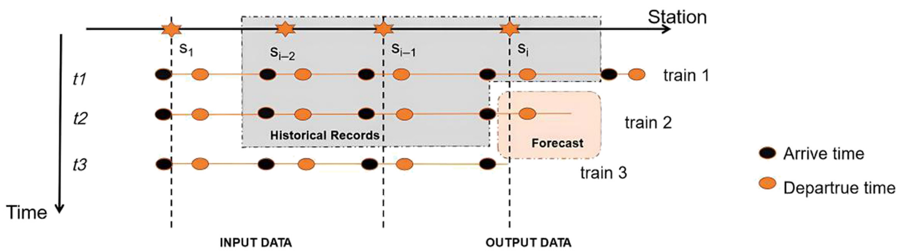
As illustrated in Figure 1, the target station for prediction is S_i, and the available information includes the operational details of train2 and its preceding train1 at S_{i-1} and S_{i-2}. This information encompasses arrival delay time, departure delay time, task type, and dwell interval. Utilizing these operational details from preceding stations as input data, the model predicts the arrival delay at the next station as the output.

3. Proposed model

3.1 Network architecture

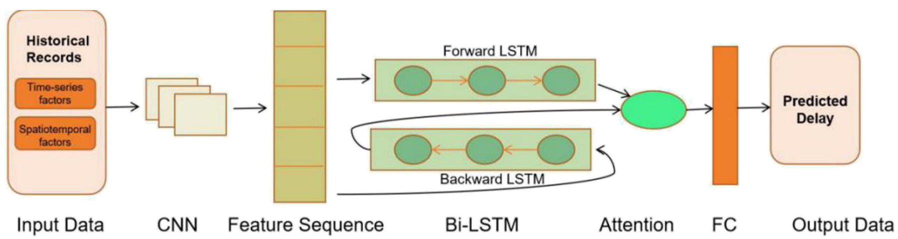
Based on the strong spatiotemporal correlation of train operation data, a neural network model named CBLA-net, which integrates CNN, Bi-LSTM, and attention mechanisms, is proposed for delay prediction. The research approach focuses on predicting the arrival delay of a target train at the destination station using the historical operation data of the target train and the preceding two trains at the preceding three stations. In other words, the model takes the historical operation data of the preceding stations as input and outputs the predicted delay time at the future station. The CBLA-net model structure is shown in Figure 2.

As shown in Figure 2, the historical data of trains containing both temporal and spatial features are input to the network. The CNN layer is employed for feature extraction. The CNN layer can capture spatial relationships between different feature values in the data,



Source(s): Authors' own work

Figure 1.
Model data processing principles



Source(s): Authors' own work

Figure 2.
The CBLA-net model structure

addressing the limitation of LSTM in capturing spatial components. Simultaneously, the extracted features still retain a temporal aspect. The sample data undergo convolution, pooling, and flattening operations in the CNN layer, resulting in a feature sequence composed of multiple feature maps, which is then input into the next-level network, Bi-LSTM. The Bi-LSTM network further learns temporal information from the feature sequence. Finally, the output vector from its hidden layer is fed into the attention layer. The attention layer computes the weighted average of the Bi-LSTM output vector, assigning weights to different time steps, thereby enhancing the influence of important time steps in LSTM and reducing the model's prediction errors. The output of the attention layer is trained through a fully connected (FC) layer, undergoes normalization, and produces the final prediction output.

3.2 CNN

CNN consists of multiple convolutional layers, pooling layers, and fully connected layers, exhibiting strong feature extraction capabilities (LeCun, Bottou, Bengio, & Haffner, 1998). By using convolutional kernels of different sizes, CNN can effectively extract local crucial information. Subsequently, through pooling layers, the input is compressed, reducing the size of the feature maps and simplifying the computational complexity of the network. Therefore, CNN is well-suited for processing and recognizing grid-structured data, such as the multidimensional data generated during the actual operation of trains.

3.3 Bi-LSTM

Bi-LSTM is formed by combining forward LSTM (Hochreiter & Schmidhuber, 1997) and backward LSTM (Schuster & Paliwal, 1997). LSTM is a specialized recurrent neural network unit that effectively addresses the issues of gradient vanishing and exploding. LSTM consists of memory units and control gates, enabling the network to better capture and remember long-term dependencies in the feature sequences of trains. Based on its functions, it can be divided into three main parts: input gate, forget gate, and output gate. These three gates are handled by gate functions using the sigmoid function, determining what information to input, forget, and output.

The formulas (1)–(6) precisely describe the working principles of LSTM. Here, f_t represents the output of the forget gate, determining what feature information of trains should be retained. It is a function of the input x_t and the previous time step's state h_{t-1}, determined by both the input and the previous state, with values ranging from 0 to 1. A value of 0 signifies complete forgetting, while 1 indicates complete retention. i_t represents the output of the input gate, determining which input information $\widetilde{C}_t$ will be updated into the model. Finally, o_t represents the output of the output gate, deciding what feature information of trains from the current time step will be passed on to the next time step.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\widetilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3}$$

$$C_t = f_t * C_{t-1} + i * \widetilde{C}_t \tag{4}$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

Bi-LSTM extends the unidirectional LSTM designed to better capture bidirectional dependencies in time-series data. The structure of Bi-LSTM is similar to that of unidirectional LSTM. However, it includes two sets of hidden states, one obtained from the forward propagation and the other from the backward propagation. These two sets of hidden states are concatenated or merged at each time step, providing a more comprehensive understanding of context information across the entire time sequence. Consequently, Bi-LSTM simultaneously considers information from both past and future time steps, contributing to a more holistic understanding of contextual relationships in the time-series data of train operations.

3.4 Attention

The attention mechanism addresses this issue by allowing the model to dynamically weight different parts of the input sequence when generating each output (Bahdanau, Cho, & Bengio, 2015). This flexibility enables the model to selectively focus on different parts of the input sequence, in this case, various aspects of the train's temporal and spatial features, rather than compressing all the information into a fixed vector.

In this paper, we compute attention weights by applying a fully connected layer to the input train operation data and obtaining a weighted output. This approach allows the model to dynamically adjust weights based on the content of the input sequence, enabling more focused attention on crucial information related to the current train operation.

4. Experiments and results

4.1 Data description and preprocessing

The dataset used in this study is from the Beijing-Tianjin high-speed railway in China, one of the busiest and most promising passenger high-speed railways with significant growth potential. Detailed records are available for each train operation, including train ID, stations, planned/actual arrival times, and other relevant data. We selected high-speed rail operation data for the Beijing South to Binhai over 10 months from October 2019 to August 2020 for late arrival prediction. The specific data format is shown in Table 2.

The station name column in the table, labeled as "BJNC, YZ, YL," represents abbreviations for each station. Specifically, "BJNC" stands for Beijing South Inter-City Station, "YZ" represents Yizhuang Station, and "YL" represents Yongle Station.

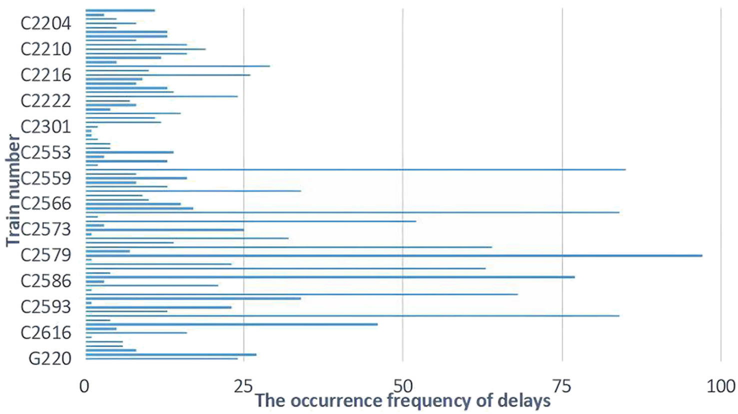
To better understand the patterns of delays, we conducted a statistical analysis of the delay frequency for trains along the route. The results are presented in Figure 3.

From Figure 3, it can be observed that the train named "C25XX" has a higher frequency of delays. Therefore, we selected all operational records of trains named "C25XX" as the

Table 2.
The example of train operation data

Date	Train	Station	Expected arrival	Expected depature	Actual arrival	Actual depature	Task	Arrival delay	Depature delay
2019-11-10	C2569	BJNC	10:29	10:29	10:29	10:29	1	False	False
2019-11-10	C2569	YZ	10:36	10:36	10:36	10:36	0	False	False
2019-11-10	C2569	YL	10:41	10:41	10:40	10:40	0	False	False
2019-11-10	C2571	BJNC	10:39	10:39	10:39	10:39	1	False	False
2019-11-10	C2571	YZ	10:46	10:46	10:46	10:46	0	False	False

Source(s): Author's own work



Source(s): Authors' own work

Figure 3. Frequency statistics of train delays

experimental dataset. Of these, 80% were used as the training set, 20% as the validation set, and the data from February 2020 were taken as the test set.

The experiment in this article utilizes historical data from this train and the preceding train at the first two stations as input to predict the delay duration of this train at the next station. The data from February 2020 is taken as the test set, while the remaining data from October 2019 to August 2020 is split into training and validation sets in an 8:2 ratio. The experiment is conducted in two ways: one using full data and the other using a dataset composed only of delayed data from the dataset. The specific data format is shown in Table 3.

The meanings of each data are as follows:

(1) Arrival late1: The delay time of the adjacent preceding train at a station.

(2) Departure late1: The arrival delay time of the adjacent preceding train at a station.

(3) Arrival late2: The departure delay time of the target train at a station.

(4) Departure late2: The arrival delay time of the target train at a station.

(5) Task: The type of task at a station.

(6) Sequence: The sorting order in the route.

In the experiment, we input the running data of the train in the first two stations (i.e. the data in the first two rows) into the network and predict the next station's arrival delay time for the target train, i.e. the "Arrival late2" in the next row.

The parameter settings of the model in the experiment are shown in Table 4.

Station	Arrival late1	Departure late1	Arrival late2	Departure late2	Task	Sequence
BJNC	0	0	−1	−1	1	1
YZ	0	1	0	1	0	2
YL	1	1	0	0	0	3

Source(s): Author's own work

Table 3. Input data format

4.2 Convergence analysis

Most delay studies train models using exclusively late arrival data. Since real-time train operation data comprises a mix of early and late arrivals, we trained the network using full data (including early and late arrival data). We conducted convergence comparisons between experiments using full data and those using only late arrival data.

In the full data experiment, the model is trained and tested using data from all time points. This means that the model can learn and consider features at different time points, including in non-delayed situations. Such experiments provide a comprehensive understanding of the entire dataset and evaluate the model's performance in various scenarios. Figures 4 and 5 depict the coefficient of determination (R^2) and loss curves during the model training process using the full data. The figures show that after 1,000 epochs of model training, R^2 approaches 1, and the loss gradually converges to around 0.3. The validation set exhibits a similar trend, indicating that the model fits the data effectively.

Figures 6 and 7 depict the coefficient of determination (R^2) and loss curves during the model training process using delay data. From the graphs, it can be observed that after 1,000 epochs of model training, both R^2 and loss gradually converge. The model losses and R^2 of the final training and validation set approach 0 and 1, respectively, indicating that the model performs well on the fitting effect when trained only with delay data.

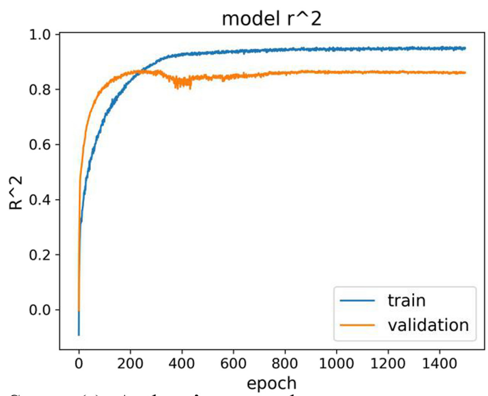
4.3 Performance results

To observe the predictive performance of the model, scatter plots are generated with the actual values on the horizontal axis and predicted values on the vertical axis, with a red line indicating the situation where predicted values are equal to the actual values.

Table 4.
Parameters of the model

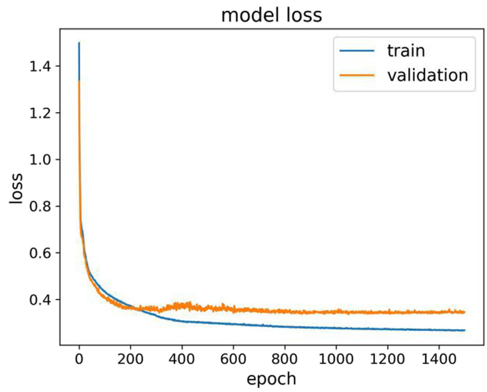
Parameters	Values
Optimizer	Adam
Epoch	1,500
Dropout rate	0.1
Input dimension	(2,5,64)
LSTM units	64
Batch size	64
Time steps	2

Source(s): Author's own work



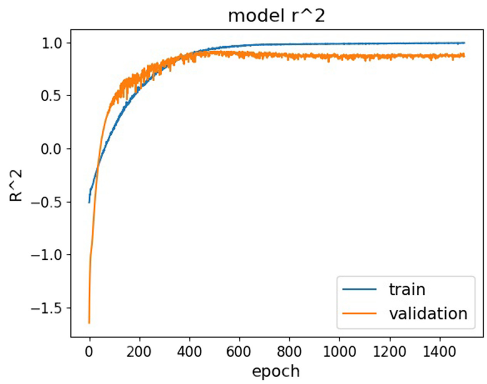
Source(s): Authors' own work

Figure 4.
R^2 curve for the full data experiment



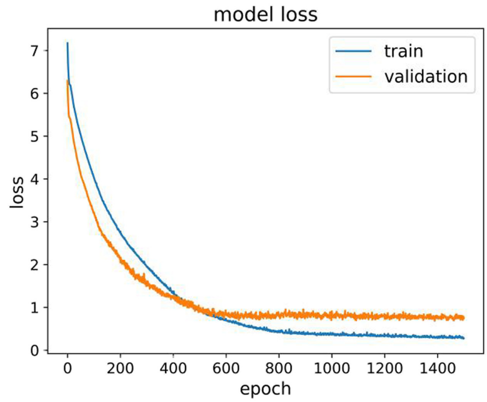
Source(s): Authors' own work

Figure 5.
Loss curve for the full data experiment



Source(s): Authors' own work

Figure 6.
R^2 curve for the delay data experiment

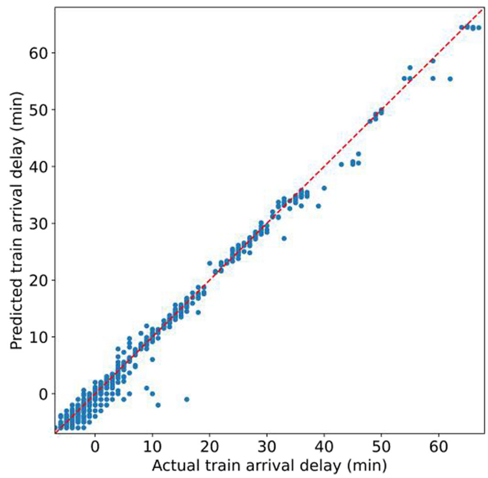


Source(s): Authors' own work

Figure 7.
Loss curve for the delay data experiment

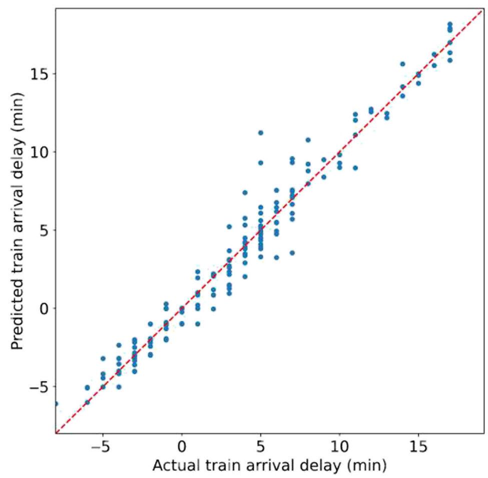
Figures 8 and 9 show scatter plots for predictions and actual values using the training and test sets in full data experiment, respectively. The figures show that the scatter points are mainly distributed near the equality line, indicating that the model accurately predicts the delay duration.

The experiment using only delay data restricts the model's training and testing data to include only delayed samples. The purpose of this experimental design is to focus on the predictive performance of the model specifically for delayed situations, ignoring information from other time points. Such experiments may emphasize the accuracy of the model in dealing with delayed situations, but they require more processing of the original data. Additionally, the continuity of the data is not as strong as in the original data. Due to the



Source(s): Authors' own work

Figure 8.
Scatter plot for training set using full data (predicted values vs actual values)

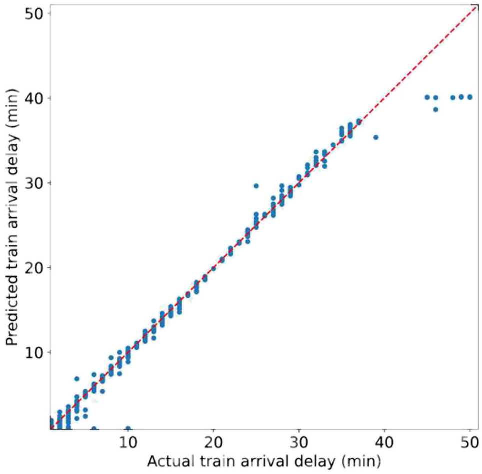


Source(s): Authors' own work

Figure 9.
Scatter plot for test set using full data (predicted values vs actual values)

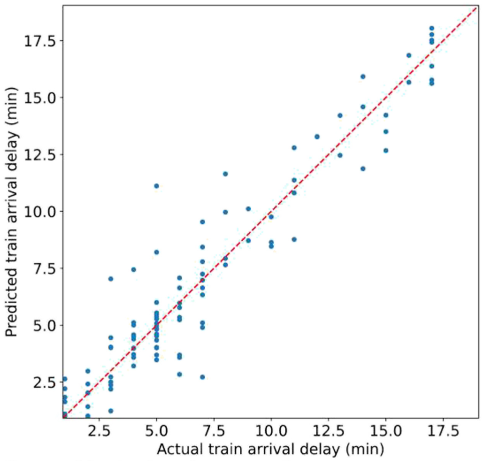
characteristics of time series model predictions, some isolated delayed data points need to be discarded, reducing the amount of data.

Figures 10 and 11 are scatter plots of predicted values versus actual values using the training and test sets in delay data experiment, respectively. The figures show that the scatter points in the training set plot are mainly distributed near the equality line, indicating accurate predictions of delay duration during model training. The scatter points in the test set plot are generally around the equality line. However, some individual data points deviate far from the equality line. It may be because the training set contains a limited number of delayed data points, and the model may not capture the comprehensive distribution and variations of delayed data. In this situation, the model may struggle to generalize well to a broader range of delayed situations in the test set.



Source(s): Authors' own work

Figure 10.
Scatter plot for training set using delay data (predicted values vs actual values)



Source(s): Authors' own work

Figure 11.
Scatter plot for test set using delay data (predicted values vs actual values)

The data from both experiments are categorized based on the delay minutes into “early and on time”, “slightly delayed”, “moderately delayed”, and “significantly delayed”. The Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are calculated for both the divided data and the entire dataset. The specific calculation formulas are (7) and (8):

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{7}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{8}$$

The results for the test set are shown in Table 5. The table shows that during the testing process, the experiment model using full data performs better in predicting slightly delayed data than the experiment model using delayed data. However, when predicting moderately delayed data, the experiment model using delayed data performs better. Overall, the full data experiment model exhibits better predictive performance across the entire dataset. Since real-world scenarios involve mixed data of various delay types, the model trained with full data has greater potential for practical applications.

To further validate the performance of the model, we compared the CBLA-net model with three other models: the CBL model, which consists of CNN and Bi-LSTM, and the CL model, which consists of CNN and LSTM. The experiments were conducted using the complete dataset, and the results are shown in Table 6, with the best value in italic.

As shown in Table 6, we observe that the performance of the CBL model is superior to that of the CL model, indicating that the bidirectional LSTM network contributes to enhanced

		Early and on time (<= 0 min)	Slightly delayed (0–15 min)	Moderately delayed (15–35 min)	Significantly delayed (>35 min)	Full state
Full data	MAE (min)	0.3327	0.7870	1.5134	\	0.5043
	RMS (min)	0.6077	1.1471	1.9194	\	0.6518
	Samples	239	153	12	\	404
Delay data	MAE (min)	\	0.9987	1.2093	\	1.0223
	RMS (min)	\	1.5473	1.3149	\	1.5231
	Samples	\	95	12	\	107

Source(s): Author’s own work

Table 5.
Comparison of the effectiveness of two experimental methods

Model	MAE (minute)	RMSE (minute)
CNN + Bi-LSTM + Attention (CBLA)	*0.504*	*0.652*
CNN + Bi-LSTM (CBL)	0.516	0.876
CNN + LSTM (CL)	0.519	0.925

Source(s): Author’s own work

Table 6.
Comparison of the effectiveness of two experimental methods

predictive performance. Additionally, the MAE and RMSE of the proposed CBLA-net is smaller than that of the CBL model without the attention mechanism, demonstrating the effectiveness of the attention mechanism in improving the predictive performance of the model.

5. Conclusion

This paper introduces a novel neural network model, CBLA-net, composed of CNN, Bi-LSTM, and attention mechanism. The model is applied to predict train delays by comprehensively considering the relationships in the propagation of train delays. Utilizing historical operation data of the target train and preceding trains at the previous station, the model forecasts the delay time of the target train at the next station. To evaluate the predictive performance of the model, it is applied in the delay prediction on the Beijing-Tianjin line. Predictive experiments are conducted under two data formats: full data and delay data, aiming to investigate the model's performance under different data structures. Besides, an analysis of prediction errors for delays of varying degrees is performed. The key conclusions are as follows:

(1) The model trained with full data exhibits superior overall performance compared to delay data. However, the predictive performance is better for pure late arrival data in medium to long delay cases. Analyzing the reasons for this phenomenon, on the one hand, the continuity of time in full data makes it easier for the model to grasp the temporal correlation of the data. On the other hand, with the same data sampling time, the overall sample size of full data is larger than that of delay data, covering a more diverse range of situations. In practical scenarios, train operation data typically include a mix of early and late arrivals. Using full data aligns better with real-world applications and reduces the complexity of data processing.

(2) Examining the prediction errors of delays at different scales reveals that the model trained with Beijing-Tianjin data performs better predicting small delays than large ones. This is because there are relatively fewer large-scale delays in the data, which is hard for the model to learn relevant patterns. In the future, expanding the distribution range of delay data could enhance the model's ability to predict delays of various magnitudes comprehensively.

(3) The proposed CBLA model will be compared with the CBL and CL models, respectively. Through experiments, it was found that both the CBL model lacking the Attention mechanism and the CL model without using Bi-LSTM performed poorly compared to the CBLA model. This demonstrates the effectiveness of the Attention mechanism and Bi-LSTM mechanism in the CBLA model.

References

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *In 3rd International Conference on Learning Representations*.

Barta, J., Rizzoli, A. E., Salani, M., & Gambardella, L. M. (2012). Statistical modelling of delays in a rail freight transportation network. *In Proceedings of the 2012 Winter Simulation Conference (WSC)* (pp. 1–12). IEEE.

de Faverges, M. M., Russolillo, G., Picouleau, C., Merabet, B., & Houzel, B. (2018). Estimating long-term delay risk with generalized linear models. *In 2018 21st International Conference on Intelligent Transportation Systems (ITSC)* (pp. 2911–2916). IEEE.

Ding, X., Xu, X., Li, J., & Shi, R. (2021). A train delays prediction model under different causes based on MTGNN Approach. *In 2021 IEEE International Intelligent Transportation Systems Conference (ITSC)* (pp. 2387–2392). IEEE.

Gorman, M. F. (2009). Statistical estimation of railroad congestion delay. *Transportation Research Part E: Logistics and Transportation Review*, *45*(3), 446–456. doi: 10.1016/j.tre.2008.08.004.

Goverde, R. M. (2010). A delay propagation algorithm for large-scale railway traffic networks. *Transportation Research Part C: Emerging Technologies*, *18*(3), 269–287. doi: 10.1016/j.trc.2010.01.002.

Heglund, J. S., Taleongpong, P., Hu, S., & Tran, H. T. (2020). Railway delay prediction with spatial-temporal graph convolutional networks. *In 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1–6). IEEE.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. doi: 10.1162/neco.1997.9.8.1735.

Huang, P., Wen, C., Fu, L., Peng, Q., & Tang, Y. (2020). A deep learning approach for multi-attribute data: A study of train delay prediction in railway systems. *Information Sciences*, *516*, 234–253. doi: 10.1016/j.ins.2019.12.053.

Huang, P., Spanninger, T., & Corman, F. (2022). Enhancing the understanding of train delays with delay evolution pattern discovery: A clustering and Bayesian network approach. *IEEE Transactions on Intelligent Transportation Systems*, *23*(9), 15367–15381. doi: 10.1109/tits.2022.3140386.

Kecman, P., & Goverde, R. M. (2015). Predictive modelling of running and dwell times in railway traffic. *Public Transport*, *7*(3), 295–319. doi: 10.1007/s12469-015-0106-7.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *In Proceedings of the IEEE* (Vol. 86, pp. 2278–2324). doi: 10.1109/5.726791.

Li, Z., Huang, P., Wen, C., Jiang, X., & Rodrigues, F. (2022). Prediction of train arrival delays considering route conflicts at multi-line stations. *Transportation Research Part C: Emerging Technologies*, *138*, 103606. doi: 10.1016/j.trc.2022.103606.

Marković, N., Milinković, S., Tikhonov, K. S., & Schonfeld, P. (2015). Analyzing passenger train arrival delays with support vector regression. *Transportation Research Part C: Emerging Technologies*, *56*, 251–262. doi: 10.1016/j.trc.2015.04.004.

Oneto, L., Fumeo, E., Clerico, G., Canepa, R., Papa, F., Dambra, C., . . . & Anguita, D. (2018). Train delay prediction systems: A big data analytics perspective. *Big Data Research*, *11*, 54–64. doi: 10.1016/j.bdr.2017.05.002.

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681. doi: 10.1109/78.650093.

Spanninger, T., Trivella, A., Büchel, B., & Corman, F. (2022). A review of train delay prediction approaches. *Journal of Rail Transport Planning and Management*, *22*, 100312. doi: 10.1016/j.jrtpm.2022.100312.

Wang, P., & Zhang, Q. P. (2019). Train delay analysis and prediction based on big data fusion. *Transportation Safety and Environment*, *1*(1), 79–88. doi: 10.1093/tse/tdy001.

Xu, X., Li, J., & Ding, X. (2022). Dynamic spatio-temporal graph convolutional network for railway train delay prediction using dynamic Bayesian network. SSRN 4175958. doi: 10.2139/ssrn.4175958.

Zhang, D., Peng, Y., Zhang, Y., Wu, D., Wang, H., & Zhang, H. (2021). Train time delay prediction for high-speed train dispatching based on spatio-temporal graph convolutional network. *IEEE Transactions on Intelligent Transportation Systems*, *23*(3), 2434–2444. doi: 10.1109/tits.2021.3097064.

Zhou, M., Xu, W., Liu, X., Zhang, Z., Dong, H., & Wen, D. (2023). ACP-based parallel railway traffic management for high-speed trains in case of emergencies. *IEEE Transactions on Intelligent Vehicles*, *8*(11), 4588–4598. doi: 10.1109/tiv.2023.3322045.

Zilko, A. A., Kurowicka, D., & Goverde, R. M. (2016). Modeling railway disruption lengths with Copula Bayesian networks. *Transportation Research Part C: Emerging Technologies*, *68*, 350–368. doi: 10.1016/j.trc.2016.04.018.

Corresponding author
Shuxin Ding can be contacted at: dingshuxin@rails.cn