# Leveraging Operational and Environmental Data for Train Delay Prediction via Deep Learning Models


Rama Krishna Eluri[1], Kotha Lahari[2], Appala Chandana Priya[3], Yarroju Rekha Sri[4], Dharmapuri Siri[5], Kandukuri Swarnalatha[6]

ramakrishnaephd7@gmail.com[1], laharikotta4@gmail.com[2], chandanapriya054@gmail.com[3], rekhasri5656@gmail.com[4], siri1686@grietcollege.com[5], swarna.kandukuri@gnits.ac.in[6]

Department of Computer Science and Engineering, Narasaraopeta Engineering College[1,2,3,4], Yellamanda Road, Narasaraopet – 522601, Andhra Pradesh, India[1,2,3,4].

Department of CSE, GRIET, Hyderabad, Telangana, India[5].Department of Electronics and Communication Engineering, G. Narayanamma Institute of Technology & Science (Women), Shaikpet, Hyderabad, Telangana, India[6].



***Abstract*—Accurate prediction of train delay is important to enhance rail operations, increase passenger satisfaction, and facilitate enhanced traffic management. This study presents a comparison of four deep learning models: Deep Neural Network (DNN), Convocational Neural Network (CNN), Long Short-Term Memory (LSTM), and Bidirectional LSTM (BiLSTM). We apply a dataset containing operating and contextual information to predict delays. The data set contains several attributes, including station-to-station distance, weather, weekday, time slots, train types, historical delays, and congestion degrees. These features were preprocessed and encoded for training. Every model leverages its strengths: DNNs handle non-linear data relationships, CNNs extract spatial-temporal features, and LSTM and BiLSTM are coded to capture long-term patterns.**

**In order to measure how well each model performed, various metrics of evaluation were employed: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Coefficient of Determination ($R^2$), and Symmetric Mean Absolute Percentage Error (sMAPE). The results indicate that the BiLSTM model performs the best in all cases, producing the best predictions.**

**This work emphasizes the extent to which BiLSTM networks are capable of learning intricate sequential dependencies, proving them fit for real-time forecasting of train delay. The findings emphasize the significance of deep learning algorithms in the design of intelligent railway systems and decision-making improvement.**



***Index Terms*—Railway Delay, Deep Learning, DNN, CNN, LSTM, BiLSTM, Time Series Forecasting**


## I. Introduction

Trains are pivotal to contemporary transportation as the backbone of passenger flow and cargo movement between countries and regions. They provide a cost-friendly, environmentally friendly, and cost-effective alternative to road and air traffic, particularly for high-capacity routes and long distances. But as cities expand and needs for transportation increase, railway systems are under pressure with regard to operations. Perhaps the most significant is the most common occurrence of delays in trains. These delays not only interfere with passenger timetables but also the entire transportation system.

Delays in train service can have a variety of adverse effects, ranging from missed connections, logistical problems, decreased network efficiency, and increased operating costs. For travelers, delays translate to longer journey times, overcrowded trains, and diminished public transport trust. For operators, delays can lead to fines, lost fuel, and reputational damage. Due to this reason, the accurate prediction of train delays has become essential for real-time decision-making, enhanced traffic management, and passenger information system improvement.

Part of the inspiration for this research comes from Malek Saharian and Stefan Voß, who investigated the application of open data and machine learning for delay prediction in rail networks. Their work demonstrated how data sets such as GTFS and weather APIs can be used to train models for precise transit prediction. We extend this work by using deep learning on a structured data set and comparing various models to determine the optimal solution for real-time train delay prediction.

Our goal is to assist in the development of smart transportation systems to enable flexible scheduling, passenger alerting, and traffic management with enhanced predictability.

Train delays are influenced by several factors, such as weather, station-to-station distance, route congestion, type of train (e.g., local, express, or superfast), day of the week, and time of day. Past delay trends also offer valuable information about chronic problems. The interconnectedness and complexity of these variables make delay forecasting difficult for conventional predictive models.

We develop and test these models with a comprehensive dataset containing both static and temporal features that are applicable to train operations. This dataset contains information such as the distances between stations, weather conditions, departure times, levels of congestion, and histories of delays. These diverse inputs train the models to make accurate delay forecasts.

Through the utilization of deep learning's pattern recognition capacity, this study seeks to develop a consistent prediction framework that enhances forecast accuracy. This will help in constructing sophisticated railway systems that can adapt to flexible scheduling, offer real-time notifications, and maximize resource utilization, finally enhancing the reliability and resiliency of public transport networks.

**The main contributions of this paper are as follows:**

- We carry out a detailed comparison of four deep learning models, DNN, CNN, LSTM, and BiLSTM, for predicting train delays using real-world railway data.
- We prepare and create a dataset with multiple features. This dataset includes time-related, operational, and environmental factors that affect train delays.
- We evaluate the models using several performance metrics, including Root Mean Square Error (RMSE), Mean Absolute Error (MAE), coefficient of determination ($R^2$), and Symmetric Mean Absolute Percentage Error (SMAPE). This approach helps us conduct a thorough assessment..
- Our findings indicate that the BiLSTM model always performs best in terms of prediction accuracy in all its metrics. This proves that it is suited well to predict sequential delays in dynamic transportation systems.

## II. Related Work

Deep learning has proven to be a robust solution within the realm of time-series prediction and transportation delay forecasting. By learning feature representations automatically and representing non-linear temporal relations, deep learning models have shown impressive performance relative to conventional methods.

Convolutional Neural Networks (CNN) have been employed in delay prediction tasks to extract local patterns and spatial features from input sequences. Although originally designed for image data, CNNs have proven effective in time series domains where patterns across temporal windows can influence the prediction of future delays. Their ability to reduce dimensionality while preserving critical patterns makes CNNs useful for preprocessing and feature extraction in transportation datasets.

Deep Neural Networks (DNN) are used extensively in situations where structured tabular input data is involved. DNNs can be applied to capture complicated feature interactions over multiple operation parameters like travel distance, station code, departure time, and day-of-week effects in train delay prediction. Although DNNs are not particularly good at capturing sequential dependencies, they can be used as powerful baseline architectures, particularly if the temporal aspect is not the dominant one or if it is covered by engineered features.

Long Short-Term Memory (LSTM) networks are particularly crafted to support sequential data and long-term dependencies. LSTMs have been used for modeling railway systems in order to identify temporal patterns of delays over time and stations. Through the use of gating mechanisms, LSTMs circumvent the vanishing gradient problem and retain information for large input sequences well, thus they are particularly apt for time-dependent railway delay modeling.

Bidirectional LSTM (BiLSTM) also takes the capabilities of basic LSTM further by processing the input sequence in both directions, i.e., forward and backward. Being bidirectional, the model is able to learn not just from the past but also from future context within the sequence. Experiments have proved that BiLSTM models are superior to unidirectional LSTM models for speech recognition, sentiment analysis, and delay forecasting because they have a deeper sequence understanding.

Previous research supports the efficacy of these four deep models for time-series delay prediction. Nevertheless, a detailed comparison of their performances on the same dataset under the same conditions is scarce. This paper fills this void by training and comparing CNN, DNN, LSTM, and BiLSTM models using a real-world railway delay dataset.

## III. Proposed Methodology

TABLE I
Description of Features Used in the Train Delay Dataset

| Feature Name | Description |
|---|---|
| Distance Between Stations (km) | The physical distance (in kilometers) between the train's current and next stop. |
| Weather Conditions | Describes the weather during the train's journey (e.g., Clear, Rainy, Foggy). |
| Day of the Week | The day on which the train journey occurs (e.g., Monday, Tuesday, etc.). |
| Time of Day | The time slot of the train schedule (e.g., Morning, Afternoon, Evening, Night). |
| Train Type | Category of train (e.g., Express, Superfast, Local). |
| Historical Delay (min) | Average past delay (in minutes) observed for similar trains on the route. |
| Route Congestion | Level of congestion or traffic on the rail route (e.g., Low, Medium, High). |

### A. Data Preprocessing

To ensure high model performance and consistency, the raw dataset underwent several preprocessing steps:

- **Handling Missing Values:** Null and inconsistent records were identified and removed.
- **Encoding Categorical Features:** Station names, train types, and day of journey were label-encoded.
- **Time Normalization:** Arrival and departure times were converted to numerical values in minutes.
- **Feature Scaling:** The numerical features were normalized using StandardScaler to enhance training efficiency.
- **Data Splitting:** The dataset was split into training and testing sets using an 80:20 ratio.

## Model Evaluation Metrics

To evaluate how well the proposed deep learning models predict train delays, we use several well-known statistical performance metrics. These indicators give an overall assessment of both the size of the errors and the consistency of the predictions.

### 1. Coefficient of Determination ($R^2$)

The $R^2$ score, or coefficient of determination, measures how much of the variance in the observed data can be predicted from the model's outputs. A higher value, nearer to 1, shows stronger explanatory power.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

*2. Mean Absolute Error (MAE)*

MAE measures the average absolute difference between the actual and predicted values. It is a simple way to see how far predictions differ from true observations, without looking at the direction of the error.

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

*3. Root Mean Square Error (RMSE)*

RMSE emphasizes larger errors due to the squaring operation and provides insight into the magnitude of the error. It is useful for penalizing large deviations more heavily.

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

*4. Symmetric Mean Absolute Percentage Error (sMAPE)*

The sMAPE metric calculates the average percentage error. It is adjusted to be symmetric and scale-independent. This makes it especially useful when handling different scales of prediction values.

$$sMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)/2}$$

*B. Deep Learning Models*

This study uses and compares four deep learning models, DNN, CNN, LSTM, and BiLSTM, to predict train delays using historical data.

**1) Deep Neural Network (DNN):** DNN uses several dense layers with ReLU activation to capture complex feature interactions. Dropout layers help reduce overfitting. While DNN does not explicitly manage sequence data, it acts as a solid baseline for modeling static feature relationships.

**2) Convolutional Neural Network (CNN):** CNN uses 1D convolutional layers to find spatial patterns in the input features. MaxPooling and flattening layers lower the dimensionality and send the data to dense layers for prediction. CNN works well for identifying localized trends in structured input. **3) Long Short-Term Memory (LSTM):** LSTM is built for sequence data and captures timing relationships in delay progression. It has memory cells and gating mechanisms that keep important time-based information. This makes it a good fit for modeling delay trends across stations and routes.

**4) Bidirectional LSTM (BiLSTM):** BiLSTM processes input in both forward and backward directions. This improves sequence learning by providing future context. It also boosts delay prediction accuracy by understanding bidirectional temporal relationships in the data.

All models are trained with the Adam optimizer and evaluated with MSE loss. Their performance is measured using RMSE, MAE, and $R^2$ metrics.

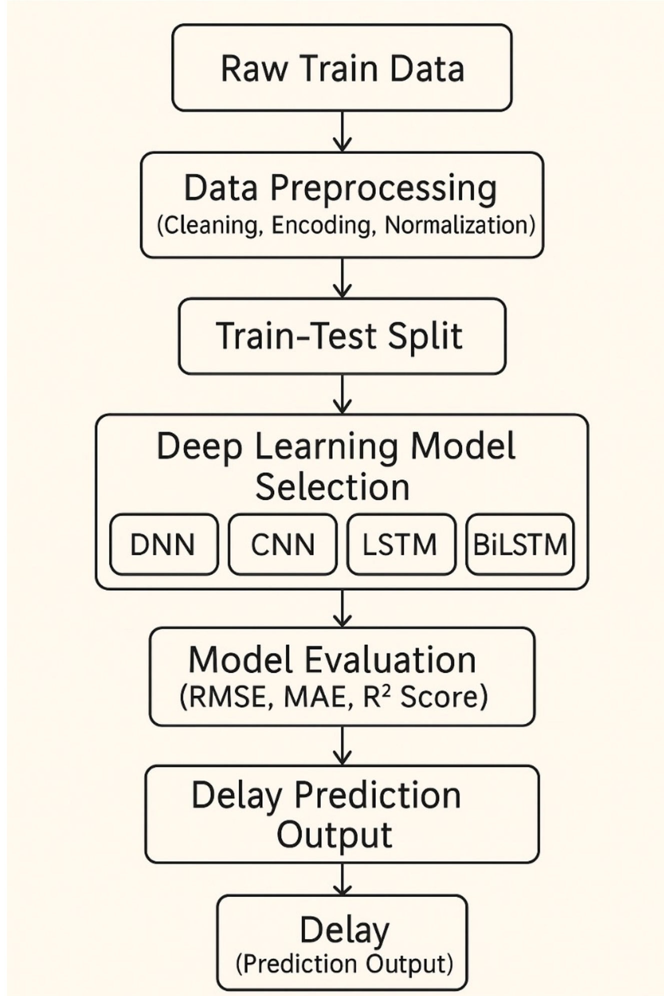


Fig. 1. Proposed Deep Learning Architecture for Train Delay Prediction

*C. Proposed System Architecture*

The suggested train delay prediction system architecture includes several stages, each of which adds to the correct modeling and forecasting of delays from past railway data. The whole pipeline is made to process raw input data, perform required preprocessing, train several deep learning models, and generate predictions that are assessed by standard performance metrics.

The process starts with the raw railway data ingestion containing features like station names, arrival and departure times, journey dates, total distance, and delay labels. Raw datasets always contain missing values, a non-uniform format for time, and categorical variables. Hence, an extensive preprocessing phase is used to achieve data quality and consistency. This involves transforming time fields to numeric types, encoding categorical features such as stations and train types, scaling numerical features through standardization, and addressing null or invalid entries.

After preprocessing, the dataset is divided into training and test subsets in a general 80:20 proportion to assess model

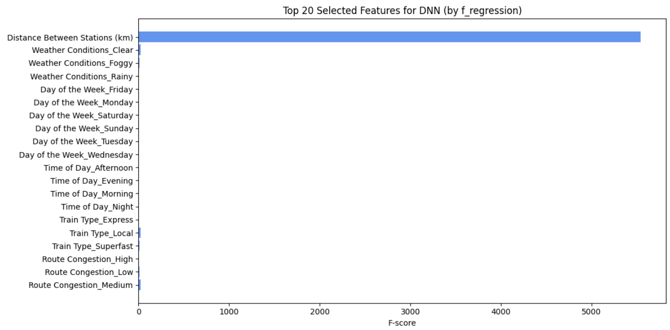


Fig. 2. Proposed Deep Learning Architecture of DNN

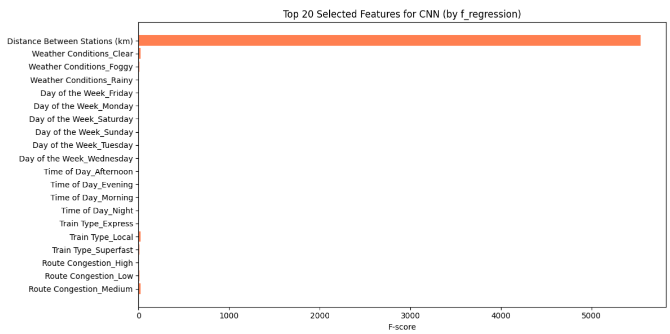


Fig. 3. Proposed Deep Learning Architecture for CNN

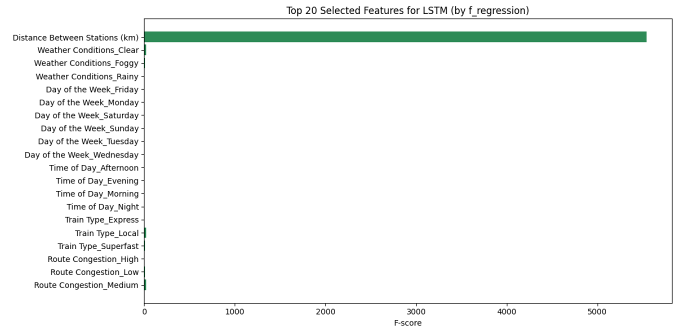


Fig. 4. Proposed Deep Learning Architecture for LSTM

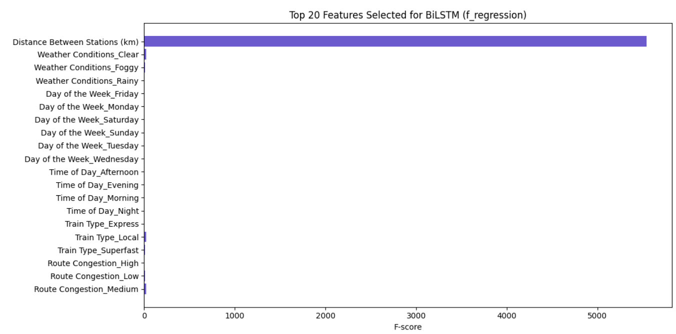


Fig. 5. Proposed Deep Learning Architecture for BILSTM

generalizability. The training set is input into four unique deep learning models — DNN, CNN, LSTM, and BiLSTM — each trained separately using the same input format to enable comparative evaluation. DNN captures rich feature interactions via dense layers; CNN is utilized to learn spatial patterns from the structured feature matrix; LSTM learns dependencies over long-term by exploiting memory cells, and BiLSTM extends temporal modeling by using both past and future contexts.

Every model is optimized with the Adam optimizer and Mean Squared Error (MSE) loss function. Regularization methods like dropout are used to prevent overfitting. Following training, the models are assessed against three critical performance measures: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination ($R^2$ score). These measures tell us about both absolute and relative performance at making predictions.

The design guarantees modularity and reusability, with the capability for future extension or integration with real-time railway control systems. Fig. 2–5 show the top 20 features for DNN, CNN, LSTM, and BiLSTM models in train delay prediction. "Distance Between Stations (km)" consistently ranks the highest. It is followed by weather, goods volume, and day of the week. All models emphasize the strong impact of spatial, weather, and time factors.

### *D. Experimental Results and Analysis*

The dataset used in this project includes historical train schedule data such as arrival time, departure time, day of journey, total travel distance, train number, and station code. The dataset is not from official Indian Railways, but it is inspired by Indian Railways for academic or project use and contains several categorical and numerical features that affect delay occurrences. The target variable indicates the delay in minutes at various stations. To test the effectiveness of the proposed deep learning models, experiments were conducted using the prepared railway delay dataset. The dataset was split into 80

All four deep learning models, DNN, CNN, LSTM, and BiLSTM, were trained on the same input structure to ensure a fair comparison. Model performance was assessed using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination ($R^2$ score).

TABLE II
PERFORMANCE COMPARISON OF DEEP LEARNING MODELS

| Model | $R^2$ (%) | MAE | RMSE | sMAPE (%) | CPU Time (s) |
|---|---|---|---|---|---|
| CNN | 94.44 | 29.68 | 49.14 | 70.86 | 32.9 |
| LSTM | 95.87 | 29.89 | 42.34 | 71.92 | 36.84 |
| DNN | 95.48 | 29.4 | 44.29 | 75.92 | 32.44 |
| BiLSTM | **96.17** | **28.42** | **40.8** | **69.17** | **316.24** |

The results demonstrate that BiLSTM achieved the best performance across all evaluation metrics, indicating its superiority in modeling bidirectional temporal dependencies.

Fig. 6 shows how different deep learning models (CNN, LSTM, DNN, BiLSTM) perform using metrics like $R^2$, MAE, RMSE, sMAPE, and CPU time. Fig. 7 compares actual and predicted delay values. The BiLSTM and LSTM models are closer to the true delay pattern.

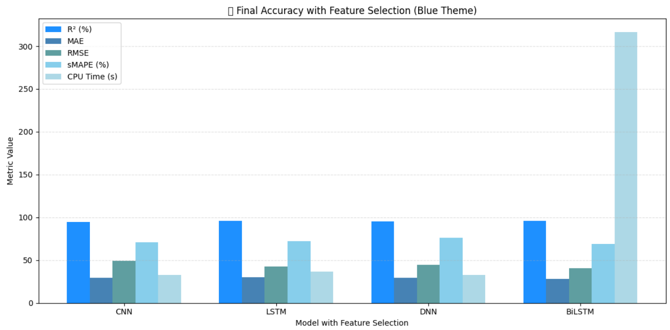


Fig. 6. $R^2$ , MAE , RMSE , sMAPE , CPU time of all algorithms

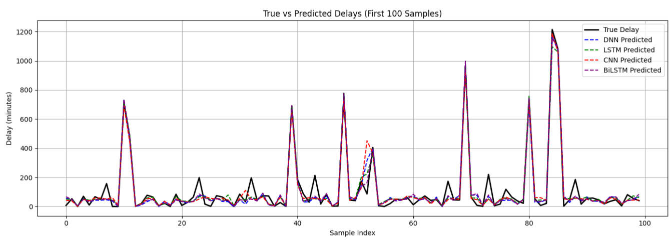


Fig. 7. Results for the different approches of Deep learning

LSTM also did well, highlighting the importance of sequence-based learning. CNN and DNN had moderate performance but struggled to capture long-term patterns in the delay data. These results suggest that models that can understand time context, especially in both directions, are more effective for predicting train delays.

### *E. Discussion*

The results of the performance analysis provide important insights into the relative strength and weakness of deep learning models embraced to make train delay prediction. Out of the four models taken into consideration, Bidirectional LSTM (BiLSTM) outperformed the others in all the evaluation metrics, including RMSE, MAE, and $R^2$ score. This spectacular performance is due to the reason that BiLSTM has access to both past and future contextual data of a time series, hence a superior viewpoint on temporal dependencies and delay propagation.

The base LSTM model similarly achieved competitive results, testifying to the strength of recurrent neural networks (RNNs) for time series forecasting tasks. Its capacity for long-range dependency learning in sequential data set it high, but it trailed just behind BiLSTM by lacking the reverse-time context.

Compared to that, CNN and DNN models, though making decent predictions, proved relatively less effective. The DNN, though effective at describing nonlinear feature interactions using dense layers, does not have mechanisms for retaining temporal memory. Likewise, the CNN model is naturally more suited to picking up local patterns and spatial hierarchies than global sequential dependencies. Consequently, both models could have failed to take full advantage of the dataset's temporal structure, which is absolutely essential to making correct forecasts of train delays.

The other significant factor to consider is the balance between accuracy and computation. The most accurate BiLSTM model came at the greatest cost of training time and memory consumption because of its dual-pass processing requirement. This aspect might be challenging in the context of deployment in real-time on devices with limited resources but is acceptable in contexts where prediction accuracy must be an utmost concern, like national railway control systems.

Interestingly, the predictive improvement was more noticeable for longer delays, which implies that deep learning models are exceptionally good at learning patterns that correspond to significant operation interruptions. This finding also implies that the models can possibly be used as early warning systems to support proactive actions to counteract cascading delays.

The uniform trend across all the evaluation metrics also validates the pipeline's robustness of preprocessing, which successfully converted raw operational data into structured input with meaning. Feature scaling, time normalization, and categorical encoding were key preprocessing operations, particularly for sensitive recurrent models to input distribution.

In a general sense, these results promote the applicability of deep learning, particularly sequential architectures such as LSTM and BiLSTM, towards delay forecasting in dynamic, high-dimensional, and multivariate settings like railway systems. The versatility of these models also widens the gate towards their integration with real-time sensors, traffic feeds, and weather inputs to transform into an overarching predictive system.

## IV. Conclusion

This research shows the potential and practicality of using deep learning techniques to predict train delays based on historical and operational railway data. We systematically implemented and evaluated four distinct deep learning models: Deep Neural Network (DNN), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Bidirectional LSTM (BiLSTM). Each model contributed uniquely to understanding and forecasting data patterns over time.

Our findings reveal that the BiLSTM model consistently produced better results across multiple evaluation metrics, including RMSE, MAE, and $R^2$ score. The BiLSTM's strength lies in its ability to learn from both past and future contexts within the sequence data, which greatly improves prediction performance for time-based systems like railway schedules. The CNN and LSTM models also performed well, demonstrating their effectiveness in modeling spatial and temporal correlations.

This work is significant for several reasons. First, it shows that deep learning is not only suitable but also very effective for large-scale transport forecasting problems. Unlike traditional machine learning methods, deep models remove the need for extensive feature engineering and offer automatic hierarchical representation learning. This is especially helpful in dynamic and noisy environments like railway systems.

Second, the proposed system is designed with modularity in mind. Each phase, from data preprocessing to model training and evaluation, can be easily expanded or integrated with more complex real-time systems. This makes the architecture flexible for both batch and streaming data scenarios, paving the way for real-world use in railway traffic control centers, mobile applications, or smart public transport dashboards.

From a societal perspective, accurate delay prediction leads to a better passenger experience, reduced congestion, efficient use of resources, and proactive incident management. It helps passengers make informed decisions and allows railway operators to optimize schedules, re-route traffic, and minimize delays.

Looking ahead, there are various opportunities for enhancement. Future research could include additional features like weather data, public holidays, accident reports, and infrastructure maintenance logs. Moreover, integrating attention mechanisms or Transformer-based models could further improve the modeling of temporal context. Reinforcement learning could be used for dynamic re-scheduling and adaptive prediction based on system feedback.

Furthermore, this framework could be applied to multi-modal transportation systems involving buses, flights, and metro services. As smart cities develop, implementing such predictive models on a large scale can lead to more sustainable, efficient, and intelligent transportation systems.

In summary, this study confirms the feasibility and effectiveness of deep learning for predicting railway delays and provides a scalable foundation for further research and development in intelligent transport systems. With ongoing improvements and real-time use, these systems can change how delays are anticipated, managed, and communicated across transportation networks.

## References


[1] K. Rautela, M. C. Trivedi, and P. P. Roy, "A BiLSTM based deep learning model for predicting train delays," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Glasgow, UK, Jul. 2020, pp. 1-7.

[2] R. K. Eluri, Y. G. Reddy, K. Valicharla, K. D. Prakash, and B. Sudheer, "Improving early detection of diabetic retinopathy: A hybrid deep learning model focused on lesion identification," in *Proc. 2024 First Int. Conf. Innovations in Commun., Electrical and Computer Eng. (ICICEC)*, Oct. 2024, pp. 1-7.

[3] L. Ge, M. Sarhani, S. Voß, and L. Xie, "Review of transit data sources: Potentials, challenges and complementarity," *Sustainability*, vol. 13, no. 20, p. 11450, 2021.

[4] Z. Li, C. Wen, R. Hu, C. Xu, P. Huang, and X. Jiang, "Near-term train delay prediction in the Dutch railways network," *Int. J. Rail Transportation*, vol. 9, no. 6, pp. 520-539, 2020.

[5] N. Marković, S. Milinković, K. S. Tikhonov, and P. Schonfeld, "Examining passenger train arrival delays with support vector regression," *Transportation Research Part C*, vol. 56, pp. 251-262, 2015.

[6] M. Sarhani and S. Voß, "On the effectiveness of SVM-based feature selection for transit delay prediction," 2021.

[7] M. Gargi, R. K. Eluri, O. P. Samantray, and K. Hajarathaiah, "Compact pyramidal dense mixed attention network for diabetic retinopathy severity prediction under deep learning," *Biomed. Signal Process. Control*, vol. 100, p. 106960, 2024.

[8] J. Wessel, J. Allen, and K. Watkins, "Integrating real-time data into GTFS, a public transit data standard," *Transportation Research Record*, vol. 2650, no. 1, pp. 110-118, 2017.

[9] P. Huang, C. Wen, L. Fu, et al., "Modeling train operation as sequences: A study of delay prediction with operation and weather data," *Transportation Research Part E*, vol. 141, p. 102022, 2020.

[10] M. Shoman, T. Laverty, and C. Andris, "Transit delay prediction using entity embeddings on heterogeneous data," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4151-4161, 2020.

[11] R. Al-Naim and Y. Lytkin, "Review and comparison of prediction algorithms for estimated time of arrival using geospatial transportation data," *Procedia Computer Science*, vol. 193, pp. 13-21, 2021.

[12] M. Sarhani and S. Voß, "Evaluation of regression algorithms for rail transit delay prediction using open data," *Public Transport*, vol. 15, no. 3, pp. 635-659, 2023. [Online]. Available: https://doi.org/10.1007/s12469-023-00300-9

[13] C. Wen, P. Huang, and L. Fu, "Predicting transit delays using graph attention networks," *Transportation Research Part C*, vol. 149, p. 103935, 2023.

[14] K. Hajarathaiah, N. B. Naidu, R. K. Eluri, and S. R. Naru, "Automatic recognition of traffic signs based on visual inspection," in *Advances in Electrical and Computer Technologies*, CRC Press, 2025, pp. 545-554.

[15] X. Wang, L. Chen, and Y. Zhao, "Comparative study of machine learning models for metro delay forecasting," *Expert Syst. Appl.*, vol. 216, p. 119419, 2023.