Journal of Theoretical and Applied Information Technology
31st March 2026. Vol.104. No.6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A DYNAMIC PRICING FRAMEWORK FOR TRAIN TICKETS USING MACHINE LEARNING PREDICTION AND RULE-BASED ADAPTATION

KIKI WIJAYA[1], YULYANI ARIFIN[2]

[1]Computer Science Department, BINUS Graduate Program, Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia
[2]Computer Science Department, BINUS Graduate Program, Doctor of Computer Science, Bina Nusantara University, Jakarta, Indonesia
E-mail: [1]kiki.wijaya@binus.ac.id, [2]yulyaniarifin@binus.ac.id

ABSTRACT

The railway transportation system in Indonesia has experienced a significant increase in demand, especially during holiday seasons, indicating the need for ticket price optimization to maximize revenue and balance passenger distribution. This study aims to develop a simple and efficient dynamic pricing model for train tickets, addressing the issues of static pricing and the confusing complexity of ticket subclasses for passengers and management. The methods employed include identifying key factors influencing ticket prices (booking time, route, service type, demand) and building a robust price prediction model using the XGBoost algorithm. Train ticket purchase transaction data from 2020 to 2025, including details like purchase time, route, ticket class, and schedule popularity, were utilized to generate accurate base prices. These base prices are then adjusted in real-time considering current demand and seat availability. Dynamic pricing simulations will evaluate price increases based on demand percentage and train occupancy rates. Model evaluation will use R-squared (R^2), Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) metrics to measure prediction accuracy. The results of this study are expected to contribute significantly to railway companies in optimizing ticket pricing strategies and improving operational efficiency.
Keywords: *Ticket Price Prediction, Dynamic Pricing, XGBoost, Railway, Machine Learning.*

1. INTRODUCTION

Railway transportation serves as a vital infrastructure supporting public mobility in Indonesia, particularly during peak periods such as the Mudik season and other national holidays. In 2022, passenger volume witnessed a significant 15% increase compared to the previous year, driven by factors such as comfort, punctuality, and competitive pricing [1]. This surge necessitates railway operators to optimize travel schedules and pricing strategies to maintain operational efficiency while enhancing profitability.

Current ticket pricing management faces substantial challenges due to highly volatile demand, which is influenced by booking lead times, travel classes, and specific routes. Rigid pricing systems that fail to account for market dynamics often lead to occupancy imbalances, where certain schedules experience overcapacity while others remain underutilized. This not only results in potential revenue loss but also creates complexity within the pricing structure, which traditionally relies on numerous and complicated subclasses.

To address these issues, a dynamic pricing approach powered by machine learning has emerged as a viable solution, allowing for real-time price adjustments based on demand, purchase timing, and seat availability. Previous studies indicate that dynamic pricing strategies can increase revenue by up to 9% [3]. In this mechanism, determining an accurate base price (H_0) is crucial and requires robust non-linear regression models capable of processing complex historical data [4].

This study focuses on evaluating and comparing several machine learning algorithms, namely Linear Regression, Random Forest, Decision Tree, and XGBoost. The XGBoost algorithm is hypothesized to be the most optimal model due to its efficiency in handling non-linear relationships between variables [2]. The ultimate objective of this research is to develop a superior base price prediction model to be integrated into a dynamic pricing framework. Through this approach, the pricing system can be

simplified by eliminating subclasses, leading to a more balanced passenger distribution and optimized corporate revenue.

2. RELATED WORK

This section discusses various relevant literature on dynamic pricing and price prediction, particularly in the context of ticket price prediction. This strategy has been widely applied across various industries, including transportation, to maximize revenue and enhance distribution efficiency. Several previous studies have made significant contributions to the understanding of ticket price prediction and the application of dynamic pricing. for instance, used linear regression to predict air ticket prices.

This research highlights how data preprocessing can improve model accuracy. Other studies [5] and [6] found that models such as Decision Tree and Random Forest Regression are effective in predicting ticket prices, achieving an R^2 value of 0.9713 and an accuracy of 81%, respectively. In the context of ensemble learning algorithms, research [7] specifically compared Random Forest and XGBoost. The results showed that XGBoost provided higher accuracy, at 85%, compared to Random Forest's 82% on the same dataset. These findings underscore the relevance of using XGBoost in this research. Meanwhile, other studies have focused on dynamic pricing strategies [8] explored this strategy using Reinforcement Learning, which was proven to adjust prices in real-time and has the potential to improve revenue management. [9] showed that dynamic pricing could increase the number of tickets sold by up to 6% and total revenue by 9%. [10] examined the impact of a low-cost carrier (LCC) entry on the ticket prices set by incumbent airlines. Their results suggest that the presence of an LCC forces incumbents to revise their management strategies. Furthermore, [11] presented an optimization approach to jointly learn demand as a function of price and dynamically set product prices to maximize revenue. Their research found that optimization-based methods can increase expected revenue regardless of the competitor's policy. Beyond the transportation sector, the application of machine learning has also proven effective in predicting price movements in the financial sector. [12], for example, used the Random Forest method to predict stock price movements on the Indonesia Stock Exchange (BEI) with 98% accuracy and an R^2 value of 0.94. Similarly, [13] also focused on stock price prediction using supervised learning algorithms to compare which one yields the best results. [14] used Artificial Neural Network and Random Forest techniques for stock price prediction, with evaluations showing their efficiency. The application of machine learning is also not limited to the transportation and financial sectors. [15] demonstrated its effectiveness in predicting property prices in Surabaya with 88% accuracy usinfg the Random Forest algorithm. A similar approach was used by Adetunji et al. [16], who employed Random Forest on the Boston housing dataset to predict price variance with a margin of error of ±5%. Interestingly. [17] also compared Random Forest and XGBoost models for house price prediction. Their results showed that the XGBoost algorithm achieved a higher R^2 score of 89%, indicating it is a more efficient and accurate model. [18] further reinforces this approach by using XGBoost regression to predict house prices. Furthermore, the use of machine learning extends to predicting the prices of used goods. [19] demonstrated that the XGBoost model can be used to accurately and efficiently predict the selling price of a used car. Meanwhile, [20] specifically compared the Random Forest and Decision Tree methods for used car price prediction, concluding that Random Forest had better accuracy (72.13%) than Decision Tree (67.21%). While the main focus of research is on technical and financial effectiveness, the impact on consumers is also an important consideration. [21] shows that dynamic pricing has a significant impact on customer behavior and perception. Their study reveals that customers can feel psychological pressure and lose trust if they feel manipulated by price changes. Collectively, all these studies demonstrate the significant potential of using machine learning, especially XGBoost and other ensemble models, to optimize pricing and make accurate predictions across various sectors, while also considering the strategic and consumer impacts.

Despite the advancements in price prediction using XGBoost and Random Forest as seen in various industries, there is still a significant gap in applying these models to a real-time adaptive framework within the Indonesian railway sector. Most existing research focuses solely on prediction accuracy without integrating a rule-based system that can handle sudden demand fluctuations and management policies. This research fills this gap by proposing a hybrid framework that combines the predictive power of XGBoost with a dynamic rule-based adaptation layer.

3. METHODOLOGY

This research methodology is designed to transform the railway ticket pricing system from a static subclass-based model into an adaptive dynamic pricing model.

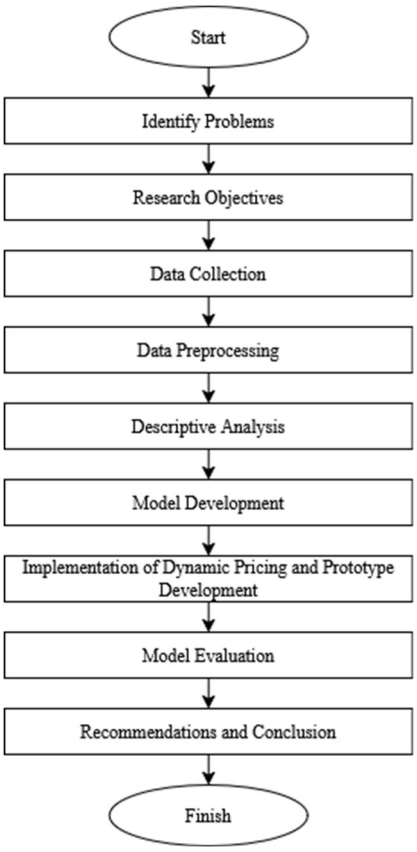

Figure 1: Methodology

3.1 Research Framework

The research is conducted through the following systematic stages:

- Introduction: Identification of problems regarding the rigidity of the static pricing system and the objective to simplify the system by eliminating subclasses.
- Literature Study: Reviewing dynamic pricing components consisting of the Base Price (H_0) based on historical data and Adjustment Factors (P,C,T) based on real-time conditions.
- Data Collection and Design: Integration of primary data (transactions), secondary data (market trends), and interviews (business logic).
- Comparative Model Construction: Training and comparing regression algorithms to determine H_0.
- Simulation and Evaluation: Implementation of the dynamic pricing formula and measuring model accuracy using statistical metrics.

3.2 Collection and Pre-processing

Data collection in this study is classified based on its primary sources: Primary Data and Secondary Data, supplemented by structured Interview techniques to extract business logic that is not available in raw transaction data.

1. Primary Data Primary data was obtained through direct extraction from the company's railway ticket transaction system. This data provides the key feature parameters for predicting the Base Price (H_0):
   - Temporal Data: Includes transaction timestamps (booking time) and actual travel dates (TRIPDATE).
   - Structural Data: Comprises travel routes (origin-destination), service categories (CLASS), and train identification numbers.
   - Occupancy Data: The seat occupancy rate at the time of departure.
   - Target Variable: The ticket price amount (AMOUNT).

   The data collection period was strategically designed to cover significant demand variations, specifically during peak season (busy periods) and off-peak season (normal periods), ensuring the dataset contains diverse pricing patterns and demand fluctuations.

2. Secondary Data Secondary data was gathered from data from previous periods. This data also includes information regarding travel trends, existing pricing policies, and external factors such as national holidays or long weekendsthat significantly influence passenger demand.

3. Interviews Structured interviews were conducted with relevant business units to gain conceptual and qualitative insights into the underlying pricing mechanisms. This information is vital for understanding the existing business rules before model construction, including:
   - Demand-Based Adjustment Logic: The rationale for price increases relative to spikes in market interest.
   - Occupancy-Based Adjustment Logic: How the percentage of occupied seats triggers changes in ticket pricing.
   - Time-Based Adjustment Logic: How the lead time between purchase and departure influences price volatility.

Data Pre-processing is performed through four crucial stages:

1. Data Cleaning: Removing missing values through row deletion.
2. Encoding: Converting categorical data (route, class, etc.) into numerical format using One-Hot Encoding.

3. Standardization: Equalizing the scale of all numerical variables.
4. Data Splitting: Dividing the dataset into 80% training data and 20% testing data.

3.3 Construction and Comparison of Predictive Models

The determination of the Base Price (H_0) is carried out using two approaches:

1. Operational Base Fare Reference: Calculated based on basic costs, profit, and Load Factor (LF) using the formula [22]:

$$\text{Base Fare} = \frac{((100\% + \text{Profit}) \times \text{Prime Cost})}{(\text{LF} \times \text{Capacity} \times \text{Route Length})} = \text{IDR/pnp km} \quad (1)$$

Where:

- Prime Cost: Sum of capital, operating, and maintenance/repair costs.
- Profit: Expected profit for business sustainability.
- Pnp.km (passenger-kilometer): Product of distance and number of passengers.
- LF (Load Factor): Proportion of passengers to carrying capacity.
- Capacity: Carrying capacity of the train.
- Route Length = Distance.

This calculation provides a logical initial ticket price based on operational factors. However, to improve prediction accuracy and incorporate additional variables like purchase time and train occupancy, the machine learning model will be utilized.

2. Comparative Machine Learning Model Construction: The construction of the predictive models follows a comparative approach to identify the most accurate algorithm for determining the ticket base fare. After calculating the operational base fare, the relevant data is further processed through a structured development lifecycle involving Linear Regression, Decision Tree, Random Forest, and XGBoost. The stages are defined as follows:

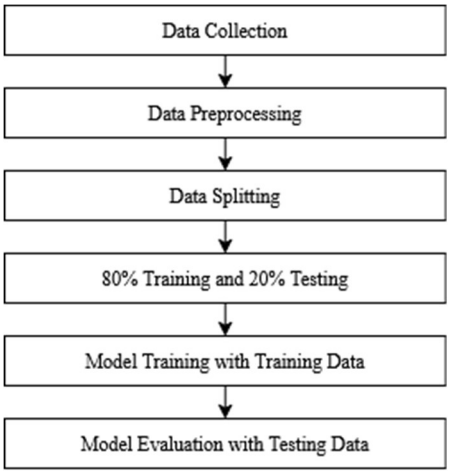

Figure 2: Model

- Data Collection and Pre-processing This initial stage focuses on ensuring data integrity. It includes handling missing values and outliers to prevent skewed results, as well as ensuring consistency in data formats. The data is then transformed into a numerical format, utilizing One-Hot Encoding for categorical variables and normalization where necessary to optimize algorithmic processing.
- Data Partitioning To ensure the model's ability to generalize to new data, the dataset is systematically split into two subsets: 80% for the training set, used to build and tune the models, and 20% for the testing set, reserved for evaluating final performance.
- Model Training During this phase, all four algorithms are trained to learn and map the complex relationships between the independent variables—which include TRAIN_NAME, ORIGIN, DESTINATION, SEAT, TRAIN_NUMBER, and AMOUNT and the dependent variable (target), which is the ticket price.
- Comparative Evaluation
  a. Linear Regression: Acts as the baseline model to identify linear correlations.
  b. Decision Tree: Used to capture non-linear patterns through hierarchical rule-based splitting.
  c. Random Forest: An ensemble method that improves accuracy by combining multiple decision trees to reduce variance.
  d. XGBoost: Specifically selected for its superior performance in handling complex, non-linear relationships and its efficiency in gradient boosting.

Journal of Theoretical and Applied Information Technology
31st March 2026. Vol.104. No.6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Once the training is complete, all models are subjected to evaluation using the test dataset to measure their generalization capabilities and to determine which model provides the most precise prediction for the dynamic pricing framework.

3.4 Dynamic Pricing Simulation

The best-selected model (H_0) is then integrated into a real-time price increase mechanism. The final price (H) is determined through the additive sum of three adjustment factors:

General Formula:

$$H=H_0\times(1+P+C+T) \quad (2)$$

Where the adjustment parameters are defined as follows:

1. Demand (P):
   Price Adjustment Formula Based on Percentage Demand Increase: Ticket prices will be adjusted using:

$$H = H_0 \times (1 + \frac{P}{100}) \quad (3)$$

   Where:
   - H = Adjusted ticket price.
   - H_0 = Base ticket price (initial price before adjustment).
   - P = Percentage price increase based on demand.

For illustration, here's a table showing calculations with base price (H_0), demand percentage (P).

TABLE 1. Demand percentage

Demand Percentage	Percentage Price Increase (P)
0% - 20%	0%
21% - 40%	5%
41% - 100%	10%

This table indicates that regardless of demand, the price increase (P) is capped at a maximum of 10% of the base price.

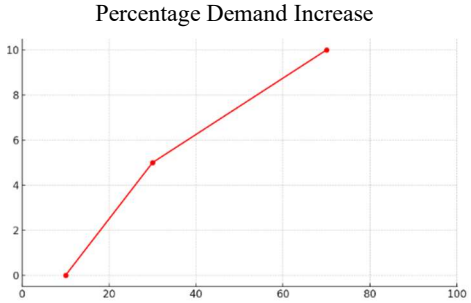

Figure 3. Demand percentage (P)

2. Occupancy (C):
   Price adjustments also consider train occupancy levels. The adjustment formula is:

$$H = H_0 \times (1 + \frac{C}{100}) \quad (4)$$

   Where:
   - H = Adjusted ticket price.
   - H_0 = Base ticket price.
   - C = Percentage of train occupancy.

For illustration, here's a table showing calculations with base price (H_0), occupancy percentage (C), and a maximum price increase of 8%:

TABLE 2. Occupancy percentage

Occupancy Percentage	Price Increase (C)
0% - 20%	0%
21% - 40%	4%
41% - 100%	8%

This table shows that for a given occupancy percentage (C), the price increase is capped at a maximum of 8% of the base price.

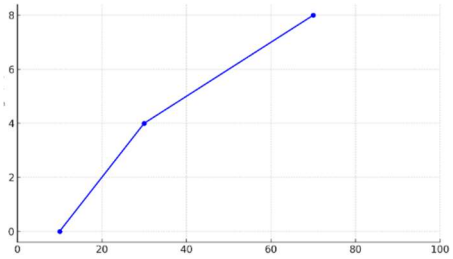

Figure 4. Occupancy percentage (C)

3. Purchase Time (T):
   To encourage early bookings, tickets will be cheaper when purchased far in advance, with prices increasing closer to departure. This adjustment is calculated as:

$$H = H_0 \times (1 + \frac{T}{100}) \quad (5)$$

   Where:
   - H = Adjusted ticket price.
   - H_0 = Base ticket price.
   - T = Percentage price increase based on purchase time.

Here's an illustrative table for purchase-time based price adjustments with a maximum increase of 7%:

TABLE 3. Price Adjustment Based on Purchase Time

Purchase Time (days before departure)	Price Increase (T)

≥ 20 days	0%
15 - 19 days	3.5%
≤ 14 days	7%

This adjustment aims to encourage earlier ticket purchases. Prices increase as departure time nears: 0% increase for purchases 20+ days out, 3.5% for 15-19 days, and 7% for 1-14 days before departure.

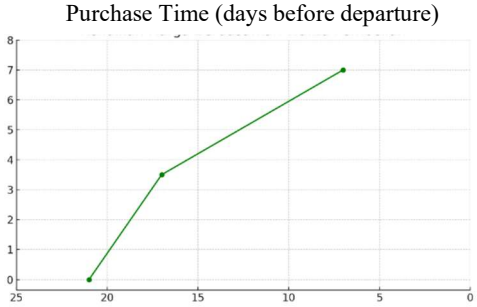

Figure 5. Price Adjustment Based on Purchase Time

3.5 Prototype Development for Dynamic Pricing Simulation

The dynamic pricing simulation prototype is designed to demonstrate the practical application of the research findings in a user-friendly and informative interface. This prototype serves as a functional bridge between the predictive models and operational decision-making. The development of this tool focuses on several key components:

- Real-Time Price Display: The system calculates and displays dynamic ticket prices that adjust automatically based on real-time inputs such as train occupancy levels, lead time to departure (purchase time), and market demand.
- Price Prediction Matrix: A structured table is provided to display predicted price points across various dimensions, including train categories and booking windows, allowing for a comprehensive overview of the pricing strategy.
- Dynamic Visualization: To enhance interpretability, the prototype incorporates a line chart that visualizes price fluctuations. This chart illustrates how ticket prices evolve over time or as demand intensity shifts, providing a clear visual representation of the dynamic pricing logic.
- User Interface Design: The interface is engineered to simplify complex backend calculations into an intuitive dashboard, enabling users to understand how the dynamic pricing system operates in a practical and measurable context.

3.6 Validation and Evaluation of Results

Model accuracy is strictly evaluated using three main metrics [23]:

1. R-squared (R^2): Measures how well the model explains the data variation.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (6)$$

Where:
- n: Total number of observations.
- yi: The actual value for the ith observation.
- y^i: The predicted value for the ith observation.
- $\bar{y}$: The mean (average) of all actual values.

2. Mean Absolute Error (MAE): Measures the average error in IDR.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (7)$$

Where:
- n: Total number of observations.
- yi: The actual value (real price in IDR).
- y^i: The predicted value generated by the model.
- |yi−y^i|: The absolute difference between the actual and predicted values.

3. Root Mean Squared Error (RMSE): Provides a higher penalty for significant errors to ensure model reliability during peak seasons.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (8)$$

Where:
- n: Total number of observations.
- yi: The actual value (real price in IDR).
- y^i: The predicted value generated by the model.
- $(yi−y^i)^2$: The squared difference between the actual and predicted values, which amplifies larger errors.

The final stage is the development of a Prototype that visualizes dynamic price changes in the form of a line chart to facilitate operational understanding of the new system without subclasses.

3.7 Recommendations and Conclusion

Based on the analysis and evaluation results, strategic recommendations are provided for future railway ticket price management. The company can optimize the implementation of dynamic pricing by

considering demand patterns influenced by seasonal factors, weekdays, and weekends.

To achieve further improvements in accuracy, it is suggested to utilize machine learning-based prediction methods that can capture more complex data patterns. The conclusion of this research emphasizes the critical importance of robust transaction data management to support data-driven pricing strategies. Ultimately, this approach is expected to enhance operational efficiency and increase customer satisfaction through more transparent and adaptive pricing.

4. RESULT

This chapter presents the research findings systematically, starting from data extraction to model implementation. The discussion focuses on the effectiveness of Machine Learning algorithms in determining the base fare and how dynamic variables influence the final ticket pricing. These stages aim to validate the transition from a static pricing system to a more optimal adaptive pricing model for both the company and the customers.

4.1 Data Collection

The data collection phase was conducted by extracting railway ticket transaction data from the company's internal system. The dataset includes comprehensive information such as booking and payment timestamps, travel routes (origin and destination), train categories, ticket prices, and occupancy levels at the time of departure.

Additionally, structured interviews were conducted with relevant business units to gain in-depth insights into the base fare calculation logic and recommendations for the dynamic pricing formulas. These formulas include:

- Demand-Based Price Adjustment Formula: Used to adjust ticket prices in response to increasing ticket demand for specific schedules.
- Occupancy-Based Price Adjustment Formula: Accounts for seat occupancy rates in determining ticket prices, allowing prices to fluctuate based on remaining capacity.
- Purchase Time-Based Price Adjustment Formula: Adjusts ticket prices based on the lead time of purchase; for instance, prices tend to increase as the departure date approaches.

The insights from these interviews provided critical qualitative foundations for developing the dynamic pricing analysis and prediction models. The data collection techniques are classified into two types:

- Primary Data: Obtained directly from the company's ticket transaction system. It covers essential parameters for analysis, such as transaction dates, train categories, routes, amounts paid, and occupancy rates. Data was collected across both peak seasons and off-peak seasons to provide a complete overview of demand patterns and price fluctuations.
- Secondary Data: Sourced from relevant external and internal documents, including company annual reports, market research, and historical supply-demand trends. This data provides a broader understanding of factors influencing price and demand, such as holiday seasons and travel policies.

4.2 Data Pre-processing

Data pre-processing involved a series of data management steps, including selection and preparation, to ensure the dataset was suitable for analysis and machine learning model construction. This stage is vital for ensuring the accuracy and relevance of the predictions.

To provide an initial overview of the data before cleaning and transformation, the following is a sample of the raw data extracted from the transaction system:

Table 4.1 Railway Departure Data: Bandung (BD) to Gambir (GMR)

BOOKING TIME	PAYMENT TIME	TRAIN NAME	ORIGIN	DESTINATION	TRIPDATE	BOOKING CODE	CUSTOMER NAME	AMOUNT
09-01-2025 15:52:27	09-01-2025 16:06:58	Argo Parahyangan	BD	GMR	2025-01-01	MY1792M	M DWIKY AFITO	220000
09-01-2025 15:52:27	09-01-2025 16:06:58	Papandayan	BD	GMR	2025-01-29	2CH7F61	YUSDI	175000
15-01-2025 17:28:26	15-01-2025 17:42:51	Argo Parahyangan	BD	GMR	2025-01-29	IHE7BBQ	HERY PRABOWO	165000
13-01-2025 15:29:03	13-01-2025 15:31:37	Argo Parahyangan	BD	GMR	2025-01-01	IHE7BBQ	HAMZAH	220000
22-01-2025 16:10:41	22-01-2025 16:12:31	Papandayan	BD	GMR	2025-01-01	OBJ7ILK	RIZKY SETIAWAN	175000
22-01-2025 16:10:41	22-01-2025 16:12:31	Papandayan	BD	GMR	2025-01-06	SVX7VD7	FADLI WITULAR	220000
22-01-2025 15:01:35	16-11-2024 16:37:30	Papandayan	BD	GMR	2025-01-04	HRU7UYA	SUHARDIMAN	165000
16-11-2024 16:33:05	16-11-2024 16:37:30	Argo Parahyangan	BD	GMR	2025-01-04	CK17BD4	A DURACHIM	175000
16-11-2024 16:33:05	16-11-2024 16:37:30	Argo Parahyangan	BD	GMR	2025-01-04	51V7HOB	Andra Olvia	175000

4.2.1 Data Selection

The data used in this study was sourced from railway ticket sales reports. To construct an effective prediction model, specific key attributes were selected to represent the variables most influential to ticket pricing. These attributes include:

- Train Name (TRAIN_NAME)
- Origin (ORIGIN)
- Destination (DESTINATION)
- Travel Date (TRIPDATE)
- Seat Information (SEAT)
- Train Number (TRAIN_NUMBER)
- Ticket Price (AMOUNT)

All selected columns are considered critical for predicting the fare accurately. The data was managed within a MySQL database before being extracted, cleaned, and processed to meet the requirements of the analytical model.

Table 4.2 Sample Data After Selection (Raw Data)

TRAIN NAME	ORIGIN	DESTINATION	TRIPDATE	SEAT	TRAIN NUMBER	AMOUNT
Argo Parahyangan	BD	GMR	2025-01-01	EKS-5/8A	45	220000
Papandayan	BD	GMR	2025-01-29	EKS-5/8B	49	175000
Argo Parahyangan	BD	GMR	2025-01-29	EKS-4/5A	45	165000
Argo Parahyangan	BD	GMR	2025-01-01	EKS-4/5B	45	220000
Papandayan	BD	GMR	2025-01-01	EKS-4/7D	49	175000
Papandayan	BD	GMR	2025-01-06	EKS-4/8B	49	220000
Papandayan	BD	GMR	2025-01-04	EKS-4/8C	49	165000
Argo Parahyangan	BD	GMR	2025-01-04	EKS-4/9C	45	175000
Argo Parahyangan	BD	GMR	2025-01-04	EKS-4/13C	45	220000

4.2.2 Data Cleaning

The first step in data processing involves cleaning the dataset to remove inconsistencies, specifically addressing missing values and formatting errors.

1. Handling Missing Values Records containing null or empty values in crucial columns—such as TRAIN_NAME, ORIGIN, DESTINATION, SEAT, TRAIN_NUMBER, and AMOUNT—are removed using the row deletion technique. Incomplete data can cause significant distortions in the analysis and degrade the predictive quality of the model.

Table 4.3 Illustration of Missing Values Identification

No	TRAIN NAME	ORIGIN	DESTINATION	TRIPDATE	SEAT	TRAIN NUMBER	AMOUNT	Remarks
1	Argo Parahyangan	BD	GMR	2025-01-01	EKS-5/8A	45	220000	Complete Data
A	Papandayan	BD	GMR	2025-01-29	EKS-5/8B	49	(Empty)	Row Deleted (Missing AMOUNT)
2	Argo Parahyangan	BD	GMR	2025-01-29	EKS-4/5A	45	175000	Complete Data
3	Argo Parahyangan	BD	GMR	2025-01-01	EKS-4/5B	45	165000	Complete Data
B	(Kosong)	BD	GMR	2025-01-01	EKS-4/7D	49	220000	Row Deleted (Missing TRAIN NAME)
4	Papandayan	BD	GMR	2025-01-06	EKS-4/8B	49	175000	Complete Data
5	Papandayan	BD	GMR	2025-01-04	EKS-4/8C	49	220000	Complete Data
6	Argo Parahyangan	BD	GMR	2025-01-04	EKS-4/9C	45	165000	Complete Data
7	Argo Parahyangan	BD	GMR	2025-01-04	EKS-4/13C	45	175000	Complete Data

After performing row deletion, only complete records remain, as shown in the table below:

Table 4.4 Dataset After Removing Missing Values

TRAIN NAME	ORIGIN	DESTINATION	TRIPDATE	SEAT	TRAIN NUMBER	AMOUNT
Argo Parahyangan	BD	GMR	2025-01-01	EKS-5/8A	45	220000
Argo Parahyangan	BD	GMR	2025-01-29	EKS-4/5A	49	175000
Argo Parahyangan	BD	GMR	2025-01-01	EKS-4/5B	45	165000
Papandayan	BD	GMR	2025-01-06	EKS-4/8B	45	220000
Papandayan	BD	GMR	2025-01-04	EKS-4/8C	49	175000
Argo Parahyangan	BD	GMR	2025-01-04	EKS-4/9C	49	220000
Argo Parahyangan	BD	GMR	2025-01-04	EKS-4/13C	49	165000

2. Data Format Conversion Following the removal of missing values, the consistency of the data formats was verified, particularly for the TRIPDATE column. Raw data often arrives in

non-uniform formats (e.g., YYYY-MM-DD, DD/MM/YYYY, or MM-DD-YY). All date entries were converted into a standardized datetime type. This step is essential to ensure that time-based features can be correctly extracted and interpreted by the machine learning algorithms.

Table 4.5 Sample Data Before Format Conversion (Inconsistent TRIPDATE)

TRAIN NAME	ORIGIN	DESTINATION	TRIPDATE (Inconsistent Format)	SEAT	TRAIN NUMBER	AMOUNT
Argo Parahyangan	BD	GMR	2025-01-01	EKS-5/8A	45	220000
Argo Parahyangan	BD	GMR	29/01/2025	EKS-4/5A	49	175000
Papandayan	BD	GMR	01-01-25	EKS-4/5B	45	165000
Papandayan	BD	GMR	2025-01-06	EKS-4/8B	45	220000
Argo Parahyangan	BD	GMR	04/01/25	EKS-4/8C	49	175000

Data After Format Conversion (Uniform Date Data Type): All date formats above have been successfully converted and standardized into the Date data type (e.g., in the standard YYYY-MM-DD format).

Table 4.6 Sample Data After Conversion (Standardized Date Format)

TRAIN NAME	ORIGIN	DESTINATION	TRIPDATE (Standardized Date Format)	SEAT	TRAIN NUMBER	AMOUNT
Argo Parahyangan	BD	GMR	2025-01-01	EKS-5/8A	45	220000
Argo Parahyangan	BD	GMR	2025-01-29	EKS-4/5A	49	175000
Papandayan	BD	GMR	2025-01-01	EKS-4/5B	45	165000
Papandayan	BD	GMR	2025-01-06	EKS-4/8B	45	220000
Argo Parahyangan	BD	GMR	2025-01-04	EKS-4/8C	49	175000

4.2.3 Feature Engineering

In this research, several Feature Engineering techniques were applied to expand the dataset and transform categorical variables into a format compatible with machine learning models.

1. Service Class Extraction (SEAT) The SEAT column was processed to extract a new feature named CLASS. Specifically, the seat identifier (e.g., EKS-5/8A) was split at the hyphen, and only the first part—representing the service class—was retained. This CLASS feature is a primary determinant of the ticket's base fare.

Table 4.7 Sample Data Following CLASS Extraction

TRAIN NAME	ORIGIN	DESTINATION	TRIP DATE	SEAT	TRAIN NUMBER	AMOUNT	CLASS (New)
Argo Parahyangan	BD	GMR	2025-01-01	EKS-5/8A	45	220000	EKS
Argo Parahyangan	BD	GMR	2025-01-29	EKS-4/5A	49	175000	EKS
Papandayan	BD	GMR	2025-01-01	EKS-4/5B	45	165000	EKS
Papandayan	BD	GMR	2025-01-06	EKS-4/8B	45	220000	EKS
Argo Parahyangan	BD	GMR	2025-01-04	EKS-4/8C	49	175000	EKS

The extraction of the CLASS feature allows the model to distinguish between service levels (e.g., Executive, Business, Economy). Since ticket pricing is heavily dependent on the service category, this new variable serves as a critical categorical input for the prediction model.

2. Time-Based and Demand Feature Extraction To capture the influence of temporal patterns and market demand on ticket pricing, four distinct features were extracted from the TRIPDATE column:
   - DAY: Identifies the specific day of the month of travel.
   - MONTH: Provides monthly information to capture seasonal trends.
   - DAY_OF_WEEK: Represents the specific day of the week (e.g., Monday, Tuesday).
   - IS_WEEKEND: A binary indicator to identify weekend departures.

Table 4.8 Sample Data Following Time-Based Feature Extraction

TRAIN NAME	TRIP DATE	SEAT	TRAIN NUMBER	AMOUNT	DAY	MONTH	DAY OF WEEK	IS WEEKEND
Argo Parahyangan	2025-01-01	EKS-5/8A	45	220000	1	1	2	0

Argo Parahyangan	2025-01-29	EKS-4/5A	49	175000	29	1	2	0
Papandayan	2025-01-01	EKS-4/5B	45	165000	1	1	0	0
Papandayan	2025-01-06	EKS-4/8B	45	220000	6	1	5	1
Argo Parahyangan	2025-01-04	EKS-4/8C	49	175000	4	1	3	0

By breaking down the TRIPDATE into these specific components, the model can capture daily and seasonal pricing patterns, as well as specific price surges associated with weekend demand.

3. One-Hot Encoding To ensure compatibility with machine learning algorithms, One-Hot Encoding was performed on five expanded categorical columns: TRAIN_NAME, ORIGIN, DESTINATION, CLASS, and TRAIN_NUMBER. This technique converts categorical values into a binary format (0 or 1).

Table 4.9 Sample Data After Full Feature Engineering

TRAIN NAME	ORIGIN	DESTINATION	CLASS	TRAIN NUMBER	DAY	MONTH	IS WEEKEND
0	1	1	1	0	1	1	0
0	1	1	1	0	29	1	0
1	1	1	1	1	4	2	0
1	1	1	1	1	6	3	1

One-Hot Encoding transforms unique categorical values into separate binary columns. This prevents the model from incorrectly assuming a mathematical order or ranking between categories and allows it to assign specific weights to each unique category. Following this process, the dataset is entirely numerical and ready for model training.

4.2.4 Data Partitioning

Once the data cleaning and feature extraction phases were completed, the dataset was partitioned into two primary subsets: the training set and the test set. This division was executed using an 80:20 ratio, where 80% of the data was allocated for model training, and the remaining 20% was reserved for evaluating the model's predictive accuracy.

This partitioning strategy is essential to ensure that the constructed model does not merely "memorize" the training data (overfitting). Instead, it aims to verify the model's ability to generalize effectively when presented with new, unseen data. By evaluating the model on a separate test set, the reliability and robustness of the price predictions can be objectively measured before the system is implemented in a production environment.

4.3 Descriptive Analysis

A descriptive analysis was conducted to understand the fundamental characteristics of the dataset. The study utilizes a railway ticket transaction dataset comprising primary features: TRAIN_NAME, ORIGIN, DESTINATION, CLASS, TRAIN_NUMBER, and the target variable, AMOUNT.

The TRAIN_NAME feature identifies the specific train service; ORIGIN and DESTINATION indicate the route stations; and CLASS—extracted from the SEAT column—serves as the primary determinant of the base fare (H_0). The TRAIN_NUMBER provides identification for specific service schedules. Furthermore, the TRIPDATE column was decomposed into temporal features, including DAY, MONTH, DAY_OF_WEEK, and a binary demand indicator, IS_WEEKEND.

The distribution of ticket prices (AMOUNT) exhibits high variance, indicating that fares are significantly influenced by structural factors (such as class and train service) as well as daily and seasonal demand fluctuations. To support the subsequent modeling phase, critical categorical features were transformed via One-Hot Encoding, ensuring the dataset is fully compatible with machine learning requirements.

4.4 Prediction Model Comparison

In this study, a comparative approach was adopted using four distinct machine learning algorithms to determine the most precise model for predicting ticket prices. The algorithms compared are Linear Regression, Decision Tree, Random Forest, and XGBoost.

The construction and comparison process followed a rigorous pipeline:

1. Preprocessing: Final numerical transformation and scaling.
2. Data Splitting: Partitioning the dataset into training and testing sets (80:20).
3. Comparative Training: Each of the four models was trained on the same dataset to ensure a fair performance comparison.
4. Metric Evaluation: Each model was evaluated based on its ability to handle the non-linear complexities of railway pricing.

The inclusion of the Decision Tree alongside Random Forest and XGBoost allows for an analysis of how hierarchical rule-based splitting compares to ensemble methods. While Linear Regression serves as the baseline, the non-linear models (Decision Tree, Random Forest, and XGBoost) are expected to

better capture the fluctuations caused by occupancy and purchase timing.

4.4.1 Linear Regression

The Linear Regression model was employed as the baseline for predicting railway ticket prices based on the available features. This model assumes a linear relationship between the input features and the target variable (AMOUNT). The training process was conducted after the data was normalized using StandardScaler to ensure all features contributed equally to the model.

The training and evaluation results of the Linear Regression model on the test set are as follows:

- Mean Absolute Error (MAE): 15,485.58.
- Root Mean Squared Error (RMSE): 21,739.04.
- R-squared Score (R^2): 0.6117.

An R^2 value of 0.6117 indicates that the Linear Regression model is capable of explaining approximately 61.17% of the total variance in railway ticket prices. The MAE value suggests that the average prediction error of the ticket price is 15,485.58.

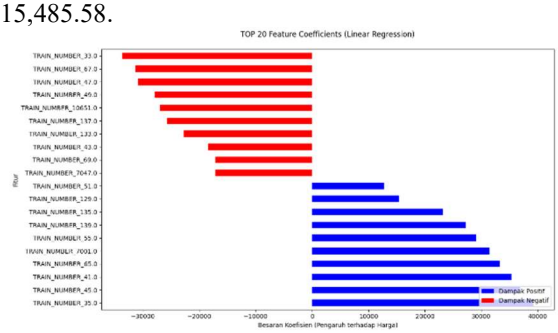

Figure 6. Feature Impact Visualization (Coefficients)

This plot displays the coefficients of the top 20 features that most significantly influence ticket prices according to the Linear Regression model. These coefficients indicate the magnitude and direction of each feature's impact on the fare. In this case, the results are heavily dominated by the Train Number (TRAIN_NUMBER) feature, highlighting its primary importance within the linear modeling framework:

- Red bars represent train numbers with negative coefficients, meaning these specific services tend to lower the ticket price. Examples include TRAIN_NUMBER_33.0 and TRAIN_NUMBER_67.0.
- Blue bars represent train numbers with positive coefficients, indicating that these services tend to increase the ticket price. Examples include TRAIN_NUMBER_35.0 and TRAIN_NUMBER_45.0. The black dashed line at 0 serves as the boundary between positive and negative impacts.

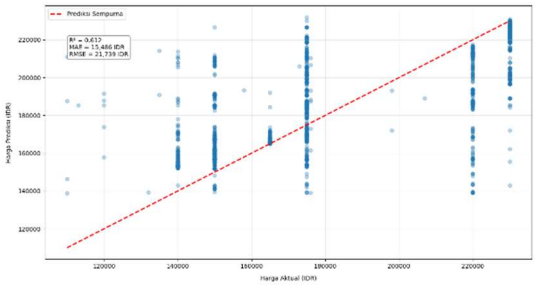

Figure 7. Actual vs. Predicted Price Comparison Visualization

This plot compares the actual ticket prices (X-axis) with the prices predicted by the Linear Regression model (Y-axis):

- Red Dashed Line: Represents the "Perfect Prediction" line, where the actual price exactly equals the predicted price.
- Blue Points: Represent individual ticket observations.
- Interpretation: The concentration of blue points around the Perfect Prediction line visually confirms the R^2 value of 0.612. While the model demonstrates moderate predictive capability, the dispersion of points—particularly in certain price ranges—indicates that there are variations in the data that the linear assumption cannot fully capture.

4.4.2 Random Forest

Random Forest is an ensemble learning algorithm based on decision trees, designed to improve prediction accuracy by aggregating the "voting" results of multiple decision trees. The evaluation results for the Random Forest model show a significant improvement over the Linear Regression baseline:

- Mean Absolute Error (MAE): 10,095.63.
- Root Mean Squared Error (RMSE): 16,984.13
- R-squared Score (R^2): 0.7823

The R^2 value of 0.7823 indicates that the Random Forest model can explain approximately 78.23% of the variations in ticket pricing, which is a substantial increase compared to the Linear Regression model (R^2=0.6117). An MAE of 10,095.63 means the average prediction error is only around 10,000.00, confirming a significant leap in predictive accuracy.

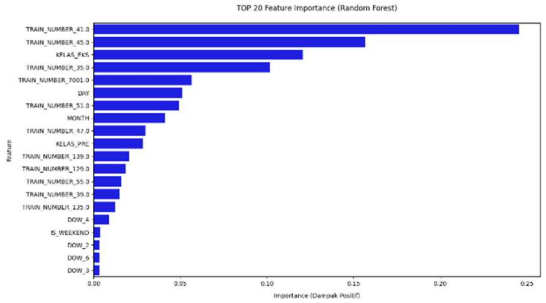

Figure 8. Feature Importance Visualization (Random Forest)

This plot illustrates the most influential features in determining ticket prices according to the Random Forest model:

- Dominant Train Numbers: In this model, specific train identification numbers (e.g., TRAIN_NUMBER_41.0 and TRAIN_NUMBER_45.0) are the most dominant factors influencing the prediction, outperforming all other variables.
- Service Class and Temporal Factors: The service class feature (CLASS_EKS) also ranks highly in importance. Temporal factors such as DAY and MONTH appear in the middle of the ranking. This suggests that the Random Forest model relies heavily on the specific identity of the train service to predict fares.
- Low Impact Features: Features such as the day of the week (DOW_4, DOW_6) and the IS_WEEKEND binary flag have a relatively smaller impact within this model's logic.

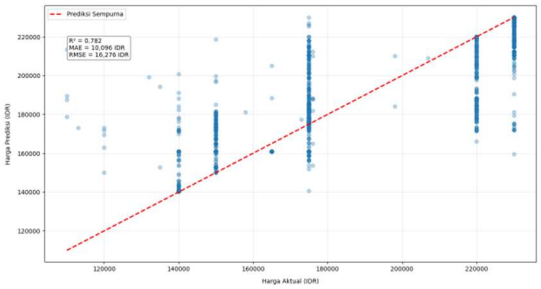

Figure 9. Actual vs. Predicted Price Comparison Visualization (Random Forest)

This plot compares the actual ticket prices (X-axis) with the prices predicted by the Random Forest model (Y-axis):

- Red Dashed Line: Represents the "Perfect Prediction" line.
- Blue Points: Represent the model's individual predictions.
- Interpretation: It is evident that the blue points are concentrated much more tightly along the ideal diagonal line compared to the previous model. This visual clustering confirms the high accuracy of the model (consistent with the R^2 score) and demonstrates a superior quality of prediction over Linear Regression.

4.4.3 Decision Tree

The Decision Tree model was implemented as a machine learning algorithm to predict ticket prices by recursively partitioning the data based on available features until a price decision is reached. This model is particularly useful for understanding the underlying decision rules governing ticket pricing. The evaluation results of the Decision Tree model on the test set demonstrate strong performance, slightly below the Random Forest model:

- Mean Absolute Error (MAE): 10,081.44.
- Root Mean Squared Error (RMSE): 16,984.13.
- R-squared Score (R^2): 0.7630

An R^2 value of 0.7630 indicates that this model is capable of explaining approximately 76.30% of the total variance in ticket prices, performing significantly better than the Linear Regression baseline. The MAE (10,081.44), which is remarkably similar to the Random Forest result, and the RMSE (16,984.13) suggest a stable average prediction accuracy.

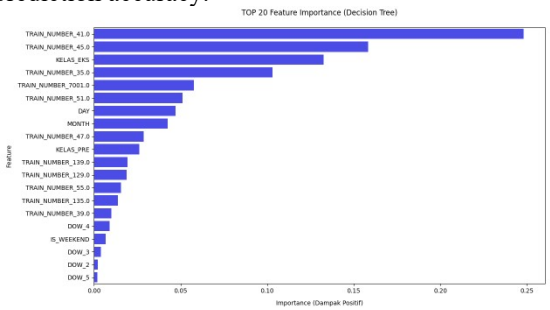

Figure 10. Feature Importance Visualization (Decision Tree)

This plot illustrates the most significant features in determining ticket prices according to the Decision Tree model:

- Dominance of Train Numbers: Consistent with the Random Forest findings, the Train Number features (specifically TRAIN_NUMBER_41.0 and TRAIN_NUMBER_45.0) are the most dominant and decisive factors influencing price predictions.
- Service Class Influence: The Executive Class feature (CLASS_EKS) also ranks high,

Journal of Theoretical and Applied Information Technology
31st March 2026. Vol.104. No.6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

confirming the substantial impact of service levels on pricing.

- Moderate Temporal Factors: Time-based factors such as DAY and MONTH hold moderate importance. This indicates that while the Decision Tree is a non-linear model, it relies heavily on specific train identities and service classes to perform data splits.

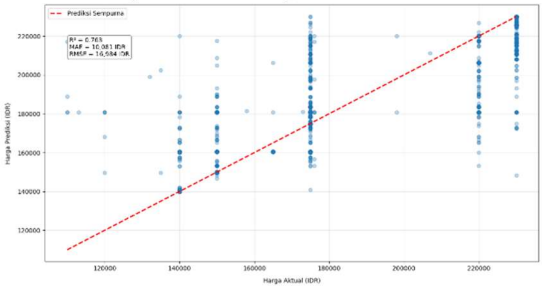

Figure 11. Actual vs. Predicted Price Comparison Visualization (Decision Tree)

This plot compares the actual ticket prices (X-axis) with the prices estimated by the Decision Tree model (Y-axis):

- Red Dashed Line: Represents the ideal line of perfect prediction.
- Blue Points: Represent the model's individual prediction results.
- Interpretation: The blue points are seen to be tightly concentrated along the ideal line. Visually, the prediction quality is similar to that of the Random Forest, proving the Decision Tree's ability to capture non-linear patterns. However, the remaining dispersion of points highlights the inherent accuracy limits of a single-tree structure.

4.4.4 XGBoost

XGBoost is a gradient boosting algorithm designed for high efficiency and performance in complex prediction tasks. Following the training and testing phases, the XGBoost model demonstrated superior performance across all evaluation metrics:

- Mean Absolute Error (MAE): 7,804.49.
- Root Mean Squared Error (RMSE): 14,840.00.
- R-squared Score (R^2): 0.8190.

An value of 0.8190 indicates that the XGBoost model is capable of explaining approximately 81.90% of the total variance in railway ticket prices. Furthermore, the significantly lower MAE (7,804.49) suggests that the average prediction error is only approximately 7,800.00. Combined with an RMSE of 14,840.00, these results establish this model as highly accurate and the most reliable among the tested algorithms.

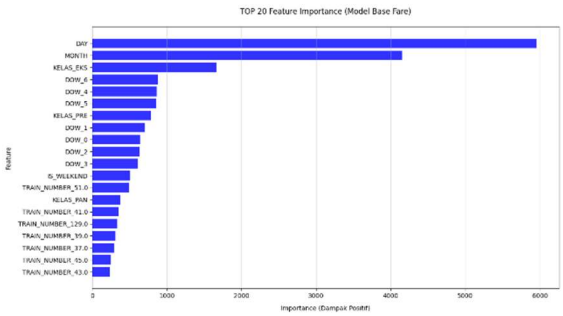

Figure 12. Feature Importance Visualization (XGBoost)

This plot illustrates the feature importance for the XGBoost model, measured by the 'F Score' or 'Weight' (the frequency with which a feature is used to split nodes across the trees). The hierarchy of importance reveals key insights:

- Temporal and Service Class Priority: Unlike the previous models, the most influential features here are temporal factors (DAY and MONTH) and the Service Class (e.g., CLASS_EKS for Executive). This indicates that the model identifies the purchase timing and the chosen month as the primary drivers of ticket price fluctuations.
- Departure Day Sensitivity: The day of the week (DOW_0 through DOW_6) and the IS_WEEKEND status also rank highly. This reinforces the conclusion that ticket pricing is highly dynamic; prices on weekdays differ significantly from those on weekends or public holidays.
- Reduced Dominance of Train Number: While the TRAIN_NUMBER feature is still present, its influence is markedly smaller compared to temporal and class factors. This suggests that XGBoost successfully captures complex interactions between multiple variables rather than relying solely on the train's identity.

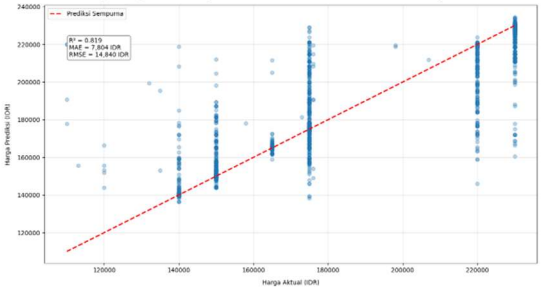

Figure 13. Actual vs. Predicted Price Comparison Visualization (XGBoost)

Journal of Theoretical and Applied Information Technology
31st March 2026. Vol.104. No.6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

This plot displays the comparison between actual prices and the prices predicted by the XGBoost model:

- Red Dashed Line: Represents the "Perfect Prediction" line.
- Blue Points: Represent the paired actual and predicted price observations.
- Interpretation: Visually, the blue points are very tightly clustered and focused around the Perfect Prediction line, particularly within the higher price ranges. This concentration validates the R^2 of 0.819, RMSE of 14,840.00, and MAE of 7,804.49. These conditions demonstrate that the XGBoost model possesses robust predictive capabilities and is highly suitable for modeling the complexities of railway ticket pricing.

4.5 Model Evaluation and Performance Comparison

The performance of the models was evaluated using three primary metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the R-squared (R^2) Score. Additionally, a 5-fold cross-validation process was implemented to measure the stability of the models when presented with new data. The following table summarizes the evaluation results for all tested algorithms:

Table 4.10 Model Evaluation Results

Algorithm	MAE	RMSE	R^2 Score
Linear Regression	15,485.58	21,739.04	0.6117
Random Forest	10,095.63	16,275.70	0.7823
Decision Tree	10,081.44	16,984.13	0.7630
XGBoost	7.804,49	14,839.60	0,8190

The evaluation results indicate that XGBoost achieved the best performance compared to the other algorithms, both in terms of accuracy (lowest MAE and RMSE) and explanatory power (highest R^2 Score). Linear Regression showed the lowest performance, which is likely due to the dataset's non-linear characteristics not aligning with the assumption of linearity [24].

4.6 Interpretation of Prediction Results

To test the models' ability to predict on new data, a sample input was used with the following parameters:

```
sample_input = {
    'TRAIN_NAME': 'ARGO PARAHYANGAN',
    'TRAIN_NUMBER': '45',
    'ORIGIN': 'BD',
    'DESTINATION': 'GMR',
    'TANGGAL_BERANGKAT': '2025-09-19',
    'KELAS': 'EKS'
}
```

Figure 14. Model Input Sample

The prediction results from the four models based on this specific input data are presented below:

Table 4.11 Prediction Results for Base Ticket Fare

Algorithm	Ticket Price Prediction (IDR)
Linear Regression	190,855.82
Random Forest	160,985.56
Decision Tree	160,520.00
XGBoost	185,936.80

XGBoost provided a ticket price prediction that most closely aligns with expectations based on historical observations. This strengthens the finding that the XGBoost model is more reliable for fare prediction within the context of the dataset used in this study.

4.7 Dynamic Pricing Simulation Based on the Prediction Model

To optimize revenue and seat utilization, a dynamic pricing system was applied by adjusting ticket prices based on three primary factors:

- Customer demand percentage (P)
- Train occupancy rate (C)
- Purchase lead time (T)

The base fare (H_0) is derived from the prediction result of the best-performing model (XGBoost).
General Price Adjustment Formula:

$$H=H_0\times(1+P+C+T) \quad (9)$$

Where:

- H = Adjusted ticket price.
- H_0 = Predicted base ticket price.
- P = Price increase based on demand percentage.
- C = Price increase based on train occupancy rate.
- T = Price increase based on purchase lead time.

The application of this formula requires a granular breakdown of each adjustment factor (P, C, and T). Each factor is determined by specific thresholds and business logic designed to respond to market conditions while maintaining competitive fare

levels. The following sections detail the simulation parameters for each component:

1. Price Adjustment Based on Demand (P)
   Ticket prices are adjusted according to the demand volume for a specific trip. As demand increases, the price adjusts by up to a maximum of 10%.

Table 4.12 Simulation of Price Increase P

Demand Percentage	Percentage Price Increase (P)
0% - 20%	0%
21% - 40%	5%
41% - 100%	10%

2. Price Adjustment Based on Train Occupancy (C)
   Fares are also adjusted based on seat availability. Higher occupancy leads to a price increase of up to 8%.

Table 4.13 Simulation of Price Increase C

Occupancy Percentage	Price Increase (C)
0% - 20%	0%
21% - 40%	4%
41% - 100%	8%

3. Price Adjustment Based on Purchase Timing (T)
   This strategy encourages early booking. The closer the purchase is to the departure date, the higher the ticket price.

Table 4.14 Simulation of Price Increase T

Purchase Time (days before departure)	Price Increase (T)
≥ 20 days	0%
15 - 19 days	3.5%
≤ 14 days	7%

4. Final Price Calculation Simulation
   This table demonstrates the simulation of railway ticket prices using the dynamic pricing model. The base fare (H_0) is taken from the XGBoost prediction (IDR 185,936).

Table 4.15 Price Increase Scenarios

Skenario	Adjustment Factors	Total Increase	Calculation	Final Price
1. Maximum Increase	- Demand - Occupancy Rate - Purchase Timing	- P = 10% - C = 8% - T = 7%	H = 185,936 × (1+0.10+0.08+0.07) = 185,936 ×1.25 = Rp232.421	IDR 232.421
2. Moderate Increase	- Demand - Occupancy Rate - Purchase Timing	-P = 5% -C = 4% -T = 3.5%	H = 185,936 × (1+0.05+0.04+0.035) = 185,936 ×1.125 = Rp209,178	IDR 209,178
3. No Increase	- Demand - Occupancy Rate - Purchase Timing	- P = 0% - C = 0% - T = 0%	H = 185,936×1 = Rp185,936	IDR 185,936

4.8 Prototype Development

Web-based prototype was developed to demonstrate the dynamic pricing system and visually validate the research findings in an interactive manner. The prototype consists of three integrated core modules, reflecting the workflow from base fare prediction to the application and monitoring of dynamic policies.

1. Prediction Page (Base Fare) This page is designed for users to input travel details (Train Name, Train Number, Origin, Destination, Departure Date, and Class) as feature inputs for the prediction model. It enables users to obtain the Base Fare (H_0) generated by the best-performing model (XGBoost) before any dynamic adjustments are applied.
   - Input: Travel time, service class, and route details.
   - Output: Predicted Base Fare and a saved prediction history.

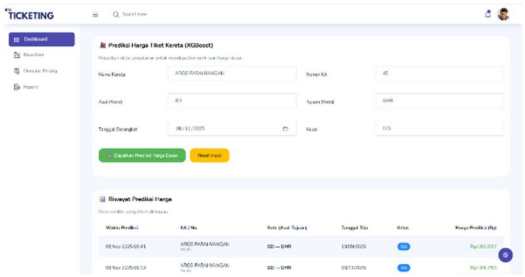

Figure 15. Prediction Input Sample

2. Dynamic Pricing Simulation Page This module serves as a testing tool where dynamic pricing policies are applied to the previously predicted Base Fare (H_0). Users can select H_0 data and apply the policy parameters P, C, and T (Demand Percentage, Occupancy, and Purchase Timing).
   - Input: Adjustable parameters for P, C, and T.
   - Output: Calculation of the total price increase percentage (P+C+T) and the resulting Final Price (H).

Journal of Theoretical and Applied Information Technology
31st March 2026. Vol.104. No.6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

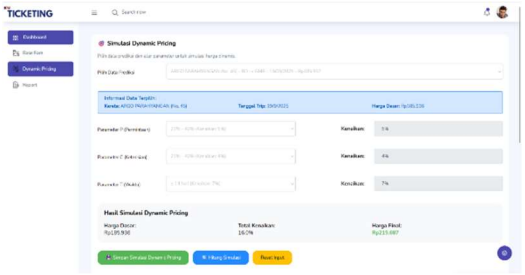

Figure 16. Dynamic Pricing Simulation

3. Monitoring Dashboard The dashboard provides a comprehensive visualization to monitor the performance and impact of the simulated dynamic pricing policies. It tracks key performance indicators such as Total Passengers, Total Income, and Ticket Sales Trends per class.
   - Performance Visualization: Displays graphs of sales trends and ticket distribution across different classes.
   - Monitoring Table: Presents a comparison between the Static Base Fare, the Percentage Increase (%), and the Final Dynamic Price based on the prediction time. This validates the system's ability to adjust ticket prices in real-time according to established policies.

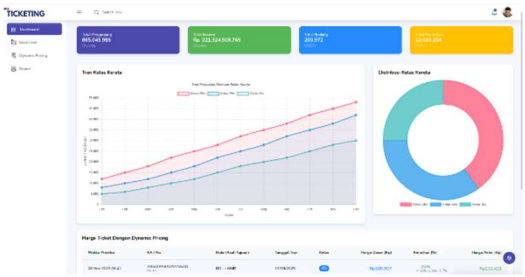

Figure 17. Monitoring Dashboard

4.9 Research Limitations and Critique

While the proposed framework demonstrates high accuracy with an R2 score of 0.819, this study acknowledges several limitations. The dataset used primarily reflects historical purchasing patterns and does not fully account for external real-time shocks, such as extreme weather disruptions or sudden policy changes from competing transportation modes (e.g., airline discounts). Additionally, the rule-based parameters currently rely on fixed thresholds which may require periodic recalibration as passenger behavior evolves over time. Future studies should consider integrating real-time social media sentiment or competitor price crawling to further refine the adaptation rules.

5. CONCLUSION

Based on the results of the research conducted, the following conclusions are drawn:

1. XGBoost Model Performance: The XGBoost model provides the best performance compared to Linear Regression and Random Forest models in predicting train ticket prices. This is demonstrated by an R^2 value of 0.8190, an MAE of 7.845, and an RMSE of 14.839, which are lower than the other models.
2. Base Price Accuracy: The XGBoost-based predictive model is capable of providing an accurate base price (H_0) to be used as the foundation for a dynamic pricing scheme, which incorporates three critical variables: demand level, train occupancy, and purchase timing.
3. Dynamic Pricing Implementation: The application of dynamic pricing with a maximum increase limit of 25% from the base price allows for flexible and responsive price adjustments to market conditions, while encouraging early ticket purchases without disadvantaging consumers.
4. Elimination of Subclasses: By implementing predictive modeling and dynamic pricing, the train ticketing system can eliminate the need for subclasses, which were previously used for price variation. This simplifies the tariff structure and provides price transparency to customers.
5. System Potential: This system demonstrates significant potential for revenue optimization, operational efficiency, and enhancing the customer experience within a data-driven transportation ecosystem.

The novelty of this research lies in the successful integration of a machine learning-based base price prediction with a rule-based adaptation mechanism tailored for train ticket subclasses. This framework adds significantly to the existing body of knowledge by demonstrating how high-accuracy prediction models can be made operationally flexible. The impact of this work provides a strategic advantage for train operators to optimize revenue and occupancy while maintaining price transparency for passengers, offering a scalable solution for dynamic pricing in emerging transportation markets like Indonesia.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to Bina Nusantara University for the academic support and facilities provided throughout this research. Special appreciation is also extended

to PT Kereta Api Pariwisata for their invaluable support in providing the datasets necessary for this study.

DATA AVAILABILITY STATEMENT

The datasets generated and analyzed during the current study are not publicly available due to commercial confidentiality and data privacy agreements with PT Kereta Api Pariwisata. However, the data are available from the corresponding author upon reasonable request for academic and research purposes, subject to approval.

AUTHOR CONTRIBUTIONS

Regarding author contributions, Kiki Wijaya was responsible for the conceptualization, methodology, software implementation, investigation, data curation, and the writing of the original draft, including visualization. Yulyani Arifin provided essential supervision, validation, and formal analysis, as well as managing the project administration and contributing to the critical review and editing of the final manuscript. All authors have read and approved the final version of this paper.

REFERENCES:

[1] Muhammad Sjahid Akbar, D. A. (2024). EID AL-FITR INFLUENCES THE NUMBER OF TRAIN PASSENGERS ON THE SUMATRA ISLAND (CALENDAR VARIATIONS TIME SERIES MODEL). BAREKENG, 2191–2202.

[2] Francesco Branda, F. M. (2020). Ticket Sales Prediction and Dynamic Pricing Strategies in Public Transport. Big Data and Cognitive Computing, 2.

[3] Kumar, A. (2023). Airline Price Prediction Using XGBoost Hyper-parameter Tuning. Communications in Computer and Information Science , 240.

[4] Liu, J. (2023). Feature correlation analysis and comparison of machine learning models for air ticket price prediction. International Conference on Machine Learning and Automation, 271.

[5] Vincent, J. R. (2023). Using Different Machine Learning Algorithms to Predict the Prices of Flight Tickets. Journal Of Student Research, 2.

[6] Janhvi Mukane, S. P. (2022). Aircraft Ticket Price prediction using Machine Learning. Journal for Research in Applied Science & Engineering Technology (IJRASET), 300.

[7] Lakshmi, J. V. (2024). An Ensemble Learning Method to Predict Airline Ticket Price Using Machine Learning. International Journal of Advanced Research in Science Communication and Technology , 294.

[8] Gursale, A. (2024). A Study on Dynamic Pricing in the Airline Industry Using Reinforcement Learning Analyzing the Impact of Reinforcement Learning on Airline Pricing Strategies. International Journal of Innovative Science and Research Technology, 895.

[9] Francesco Branda, F. M. (2020). Ticket Sales Prediction and Dynamic Pricing Strategies in Public Transport. Big Data and Cognitive Computing, 2.

[10] Rafael Varella, J. F. (2020). Dynamic pricing and market segmentation responses to low-cost carrier entry. Transportation Research Part E Logistics and Transportation, 2.

[11] Perakis, D. B. (2020). Dynamic Pricing: A Learning Approach. Springer Science and Business Media, Inc, 46.

[12] Lubis, M. A. (2025). Using the Random Forest Method in Predicting Stock Price. Data Science, Information Technology, and Data Analytics.

[13] R S Abirami, K. M. (2022). STOCK MARKET PRICE PREDICTION USING RANDOM FOREST AND SUPPORT VECTOR MACHINE. International Journal of Innovative Research in Technology.

[14] Mehar Vijh a, D. C. (2020). Stock Closing Price Prediction using Machine Learning Techniques. International Conference on Computational Intelligence and Data Science (ICCIDS 2019).

[15] Rinabi Tanamal, N. M. (2023). House Price Prediction Model Using Random Forest in Surabaya City. TEM Journal.

[16] Abigail Bola Adetunji, O. N. (2022). House Price Prediction using Random Forest Machine Learning Technique. ScienceDirect.

[17] Li, H. (2023). House Price Prediction and Analysis Based on Random Forest and XGBoost Models. GEFHR.

[18] .J. Avanijaa, G. S. (2021). Prediction of House Price Using XGBoost Regression Algorithm. Turkish Journal of Computer and Mathematics Education.

[19] Qian, T. (2023). Used Car Price Prediction by Using XGBoost. FIBA.

[20] Purwa Hasan Putra, A. (2023). Random forest and decision tree algorithms for car price prediction. Jurnal Matematika Dan Ilmu Pengetahuan Alam LLDikti Wilayah 1 (JUMPA).

[21] Gonzalez, D. A. (2024). How does dynamic pricing affect airline customers. Research Archive of Rising Scholars.
[22] INDONESIA, M. P. (2018). PEDOMAN TATA CARA PERHITUNGAN DAN PENETAPAN TARIF ANGKUTAN ORANG DENGAN KERETA API. MENTERI PERHUBUNGAN REPUBLIK INDONESIA.
[23] Lubis, M. A. (2025). Using the Random Forest Method in Predicting Stock Price. Data Science, Information Technology, and Data Analytics.
[24] Lin, W. A.-S. (2023). Deep-Learning-Powered GRU Model for Flight Ticket Fare Forecasting. applied science.