Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

June 2026. Vol. 5. No. 3; e-ISSN: 2808-4519

Implementation of the Gradient Boosting Algorithm for Palm Oil Price Prediction

Wilbert Fernando[1*], Hendri[2], Robby Wijaya[3]

[1, 2, 3]*Informatics Engineering*

[1, 2, 3]*STMIK TIME, Medan, Indonesia*

wilbertfernando711@gmail.com[1*], h3ndr1wu@gmail.com[2], robbyhuang89@gmail.com[2]

Abstract

The price of palm oil is highly volatile due to the influence of global market dynamics, trade policies, and climate change, creating uncertainty for industry players in decision-making. This research aims to implement the XGBoost (Extreme Gradient Boosting) algorithm, optimized using Grid Search Cross-Validation, to predict palm oil prices. The dataset used is the Palm Oil Futures Historical Data.csv obtained from Kaggle, consisting of nine features. Data preprocessing is performed using StandardScaler for normalization, followed by model training with hyperparameter tuning. The system is built as a web-based application separating the frontend using PHP and Flask as the Backend API. Testing on 105 test data points yielded an MAE of 43.97, RMSE of 65.14, and R^2 of 91.82%, demonstrating the model's strong ability to explain palm oil price variation. Based on the results, the XGBoost algorithm is suitable as a decision-support tool for commodity price prediction, achieving high accuracy consistent with standard criteria for commodity price forecasting and capable of handling large datasets.

Keywords: *Price Prediction, Palm Oil, XGBoost Algorithm, Hyperparameter Tuning, Machine Learning*

1. Introduction

The palm oil sector is a commodity with high economic value and serves as one of Indonesia's primary sources of foreign exchange. The agricultural sector ranks third as a major contributor to Gross Domestic Product (GDP), with its contribution continuing to increase. Therefore, the agricultural sector plays a very important role in the Indonesian economy [1]. Palm oil prices in the international market are highly dependent on volatile market dynamics, including tariff policies, climate change, rainfall, and the dominance of private companies in plantation ownership [2]. This price instability presents a challenge for industry players in decision-making, making the need for an accurate price prediction system critical for mitigating economic uncertainty risks.

Malaysia is one of the world's main producers and exporters in global trade. Due to Malaysia's dominance, the palm oil commodity market price uses the Malaysian Ringgit (MYR) as currency and is used as a benchmark in international trade. Systematic management of historical data can provide a more realistic picture of price movements [3]. However, insufficient algorithm optimization also affects accuracy, preventing the achievement of optimal results. Advances in Machine Learning technology enable the development of more accurate price prediction models with the ability to identify hidden patterns in historical data. Given the high price volatility, the need for prediction systems is crucial, as incorrect decisions can cause financial losses for all industry players.

The Gradient Boosting algorithm, with a focus on XGBoost implementation, was selected because it excels at handling non-linear data patterns, offers high computational efficiency, and includes regularization techniques to prevent overfitting [4]. The combination of XGBoost with optimization techniques can produce better performance than standard models for forecasting data [5]. To improve model performance, hyperparameter optimization using the Grid Search Cross-Validation method is applied to find the most suitable parameter combination for the model [6]. Based on this problem background, this study aims to implement an optimized XGBoost algorithm in a web-based system, with research objectives of applying the XGBoost algorithm by determining hyperparameters using Grid Search and enabling it to serve as a decision-support tool in the price prediction process.

Based on this background, the researcher is interested in conducting a study on palm oil price prediction with the title: "IMPLEMENTATION OF THE GRADIENT BOOSTING ALGORITHM FOR PALM OIL PRICE PREDICTION"

2. Theoretical Review

2.1 Palm Oil

Palm oil traded in international markets originates from Crude Palm Oil (CPO). As an important commodity in trade with high economic value, palm oil can produce several processed products, one of which is RBD Olein, which serves as a staple food for the community without economic disparity [7]. Products traded in international markets have an impact on futures trading, which functions as a financial instrument that helps business actors protect themselves from the risk of price changes while also serving as a means of open price formation. High levels of price volatility can trigger economic crises [8].

2.2 Prediction

Prediction is a structured process for estimating the value of something likely to occur in the future by utilizing historical data. Prediction results are not always exact; they represent only the closest possible estimate. In this context, prediction is performed to estimate the next closing price [9].

2.3 Related Work

Prior research on cooking oil price prediction using Multiple Linear Regression showed that this approach is effective for linearly patterned data, but is unable to capture non-linear relationships and complex interactions [10].

In contrast, a comparative study of C4.5, Random Forest, and Gradient Boosting algorithms for plantation commodity price classification showed that these algorithms excel at handling complex datasets and yield good accuracy [11].

Study [1] conducted palm oil price prediction was performed using Linear Regression and Random Forest algorithms, where Linear Regression outperformed Random Forest in two of the three evaluated scenarios. This study demonstrates that machine learning methods are applicable to commodity price forecasting, though they capability is limited in handling complex non-linear patterns.

Study [12] applied XGBoost regression to forecast biomass gasification system parameters using R^2 and RMSE evaluation, it achieved R^2 values exceeding 0.9 for multiple targets. Which highlights the strength of XGBoost in handling complex system problems.

Study [13] evaluated LSTM and XGBoost models for forecasting Crude Palm Oil (CPO) production at a palm oil mill. LSTM attained 93.7% accuracy with an RMSE of 21.04, while XGBoost showed improved performance after hyperparameter tuning with an RMSE of 22.17 and an accuracy of 92.8%.

Unlike Study [14] compared five algorithms ARIMA, LSTM, SVR, Prophet, and XGBoost in an automated agricultural commodity price prediction. XGBoost yielded the lowest RMSE and MAE values among them, confirming its superiority in commodity price prediction.

Other Research [15] used the Gradient Boosted Trees Regression method to predict premium rice prices in 2024, yielding an RMSE of 0.0473 and R^2 of 0.9047. These results indicate that gradient boosting algorithms have high accuracy in predicting food commodity prices.

3. Method

This research uses a quantitative approach, integrating historical data analysis with machine learning-based predictive modeling for crude palm oil price forecasting. The research follows a sequential methodology, encompassing problem identification, data collection, preprocessing, model training and optimization, evaluation, result visualization, and web-based system deployment, as illustrated in Figure 1 below.

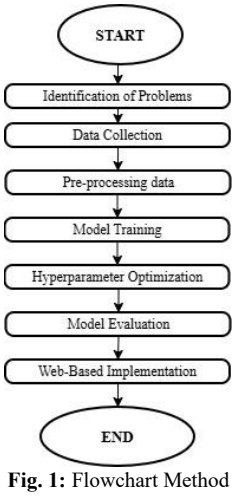

Fig. 1: Flowchart Method

3.1 Data Collection

This research uses a quantitative method based on historical data and prediction using a Machine Learning model. The data source is

obtained from the Kaggle website, with the file named "Palm Oil Futures Historical Data.csv" containing a total of 833 historical price records. Following removal of non-predictive identifiers, seven input features remain, as summarized in Table 1 below.

Table. 1: Palm Oil Futures Dataset Features

Features	Type	Description
Date	Date	The trading or recording date
Price	Numeric	The closing price
Open	Numeric	The opening price of the contract
High	Numeric	The highest price reached during the trading session.
Low	Numeric	The lowest price reached during the trading session.
Vol	Integer	The trading volume
Chance	Float	The decimal representation of the daily price

3.2 Data Preprocessing

Preprocessing is performed to prepare the data before model training. The first step is converting the 'Vol' and 'Change' features to integer and float data types respectively, then handling outliers using the IQR method on all numerical features with a limit of 1.5 × IQR. This process is shown in figure 2 below.

```
original_len = len(df)
for col in ['Price', 'Open', 'High', 'Low', 'Vol_num']:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    df = df[(df[col] >= lower_bound) & (df[col] <= upper_bound)]
```

Fig. 2: Handling Outliers

The second step is performing feature engineering to generate derived features including trend indicators, volatility indicators, momentum indicators, comparison features, and volume indicators. All derived features are calculated based on historical data. The feature engineering process is shown in figure 3 below.

```
# indikator tren
df['MA_7'] = df['Price'].rolling(window=7).mean()
df['MA_14'] = df['Price'].rolling(window=14).mean()

# fitur pembanding
df['Price_lag_7'] = df['Price'].shift(7)

# indikator volatilitas
df['Price_std_7'] = df['Price'].rolling(window=7).std()
df['High_Low_Range'] = df['High'] - df['Low']
df['Price_Range_Pct'] = (df['High'] - df['Low']) / df['Open'] * 100

# indikator momentum
df['Price_Change_7d'] = df['Price'].diff(7)

# indikator volume
df['Vol_MA_7'] = df['Vol_num'].rolling(window=7).mean()
df['Vol_Ratio'] = df['Vol_num'] / df['Vol_MA_7'].replace(0, np.nan)
```

Fig. 3: Feature Engineering

The next step is normalizing features using StandardScaler so that each feature has a distribution with mean = 0 and standard deviation = 1. Data splitting divides the data into training and test sets at an 80% to 20% ratio, as shown in figure 4 below.

```
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, shuffle=False
)
```

Fig. 4: Split Data

3.3 Model Training

The model is built by initializing it with initial default parameters such as objective='reg:squarederror' and random_state=42. Feature engineering is then performed on the original data to generate new features before training the model on data normalized with StandardScaler. The default parameter is shown in figure 5 below.

```
xgb_model = XGBRegressor(
    random_state=42,
    objective='reg:squarederror',
    n_jobs=-1,
    tree_method='hist',
    min_child_weight=8,
    gamma=0.2,
    colsample_bytree=0.7
)
```

Fig. 5: Default Parameter

3.4 Hyperparameter Optimization

The hyperparameter tuning process uses the Grid Search method to find the best combination. The hyperparameters used in this algorithm include n_estimators with values between 80 and 150, learning_rate with values between 0.05 and 0.1, max_depth with values between 2 and 7, subsample with values between 0.05 and 0.1, and reg_alpha and reg_lambda with values between 18 and 25. These parameters are shown in the figure 6 below.

```
param_grid = {
    'n_estimators': [90, 80],
    'learning_rate': [0.05],
    'max_depth': [3],
    'subsample': [0.6],
    'reg_alpha': [30, 32],
    'reg_lambda': [25, 28]
}
```

Fig. 6: Combination Hyperparameter

The final model from the best hyperparameter combination is then retrained using all training data and tested on the 20% split test data.

3.5 Model Evaluation

The trained model is evaluated using MAE, RMSE, and R^2 metrics to measure model performance. Adapted from the work of D. Chicco, M. J. Warrens, and G. Jurman, a model is considered acceptable when standard criteria are met: MAE in the range of 30 to 60, RMSE in the range of 50 to 100, and R^2 above 0.8, but capped below 0.91 to avoid overfitting [17].

Model performance demonstrates prediction accuracy through three primary key metrics, with the main objective of this model being to minimize prediction errors.

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

$$RMSE = \sqrt{\square} \quad (2)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\square} \quad (3)$$

3.6 Web-Based Implementation

The prediction system with the optimized XGBoost model is deployed into a web-based application built with four separate architectures: Machine Learning Model (.pkl), Database (MySQL), Frontend (PHP), and Backend (Flask API). The system is designed with a simple UI/UX. After the integration process, the system is tested comprehensively to ensure the model can function accurately and stably before displaying the prediction output.

4. Result

The steps described in Chapter 3 were carried out in a structured manner, from data collection to system implementation.

4.1 Data Preprocessing Results

This process was performed on the "Palm Oil Futures Historical Data.csv" dataset with a total of 833 records. The feature engineering process generated 9 additional derived features. After this preprocessing, 524 records remained ready for further research and testing , as shown in the figure 7 below.

```
Data Cleaning
   - Penanganan outlier dengan IQR method:
     Outlier removed: 273 rows
     ✅ Dataset size: 560 rows

   Normalisasi Fitur & Feature Engineering
   - Creating engineered features...
   ✅ Features created: 524 rows remaining

   ✅ Features created: 524 rows remaining

    Split Data (80% Train, 20% Test)
   Train size : (419, 14)
   Test size  : (105, 14)
```

Fig. 7: Output Preprocessing Data

4.2 Model Training

Initial parameters were set on the XGBoost model before performing hyperparameter tuning. Initial configuration is required to reduce the risk of overfitting while accelerating training.\

4.3 Hyperparameter Optimization Results

The optimal parameter combination was selected because it produced the best cross-validation score with an optimal balance between prediction accuracy and model generalization capability. The optimization process resulted in the best parameter combination as determined by Grid Search Cross-Validation results are shown in the figure 8 below.

HASIL GRID SEARCH - BEST PARAMETERS:
- learning_rate: 0.05
- max_depth: 3
- n_estimators: 90
- reg_alpha: 30
- reg_lambda: 25
- subsample: 0.6

Fig. 8: Output Grid Search

4.4 Model Testing Results

Model testing was performed on 105 split test data points. The evaluation results are shown in Table 1. The R^2 value of 91.82% is consistent with prior research on "The Importance of Input Features Applied to Artificial Intelligence Models for Biomass Gasification Systems", which obtained R^2 above 0.9 using XGBoost in complex regression systems [14].

Table 2: Comparison of Standard Criteria and Model Results

Metric	Result	Standard
MAE	43.97	Good (30 – 60)
RMSE	65.14	Good (50 – 100)
R^2	91.82%	High (> 0.85)

The results in Table 2W show that all three metrics fall within the accepted standard criteria for commodity price prediction: MAE of 43.97 falls within the good range (30-60), RMSE of 65.14 is within the acceptable range (50-100), and R^2 of 91.82% exceeds the 0.85 threshold, indicating strong model performance.

4.5 System Implementation

The following describes the UI design of the web-based Palm Oil price prediction system that has been built:

1. Guest Dashboard and User Dashboard

The dashboard for the public (guests) serves as the landing page when users first access the system, displaying the latest Palm Oil price, a New Prediction button (which redirects to the Login page upon clicking), and a six-month price trend graph, as shown in Figure 9.

Fig. 9: Guest Dashboard (Landing Page)

This page will appears when users have successfully logged in with valid credentials.Once the credentials are entered correctly, the authenticated dashboard presents similar content plus additional metrics—latest Palm Oil price, total predictions, average confidence—along with the New Prediction button. The main dashboard is shown in figure 10.

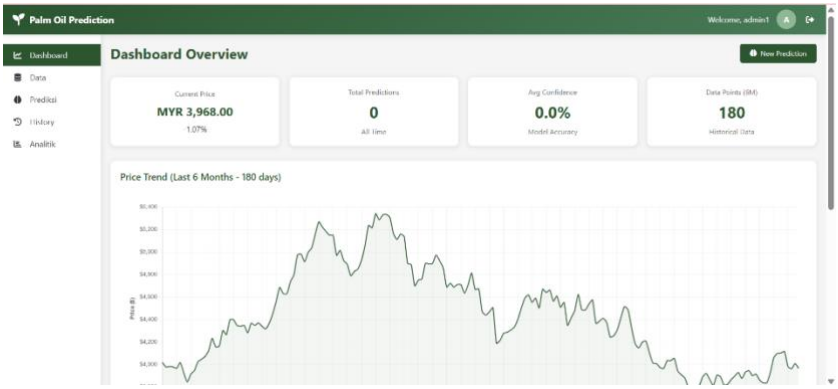

Fig. 10: Dashboard Page

2. Login Page

The page for users to log in is supported by PHP session management for user authentication. On this page, users input a username and password already stored in the database through the UserModel before being redirected to the user dashboard. The Login Page is shown in figure 11.

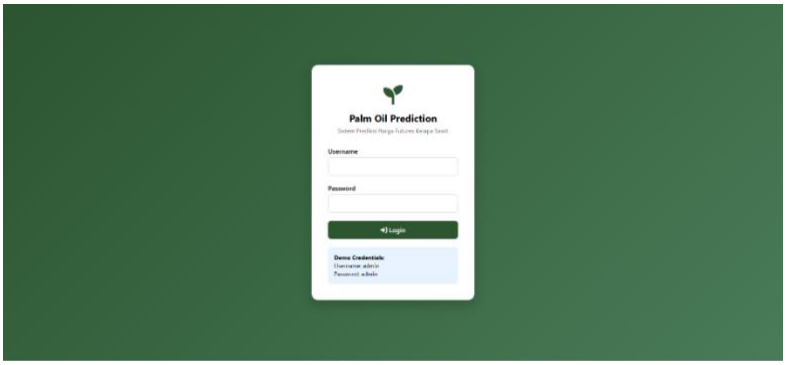

Fig. 11: Login Page

3. Data Page

This page presents historical price information in a table format retrieved from the database through the DataModel, accessible to both the public and logged-in users. The data page is shown in figure 12.

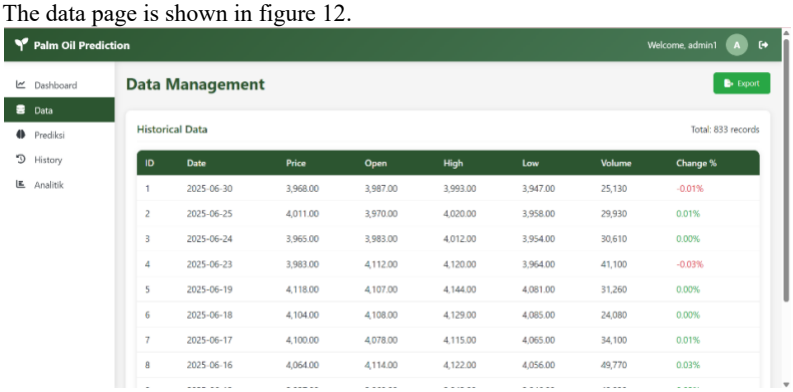

Fig. 12: Data Page

4. Prediction Page

This page provides an input form for users to perform predictions. The form requires users to fill in the Open, High, and Low columns. The prediction page is shown in figure 13.

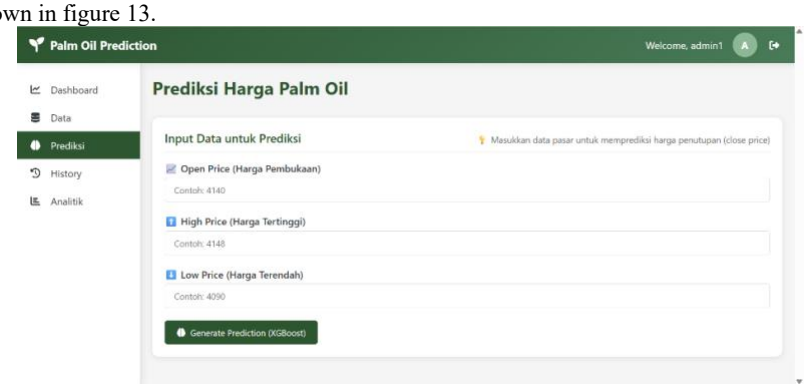

Fig. 13: Prediction Page

The input data is sent to the Flask API for processing through validation, feature engineering, and normalization stages before the XGBoost model displays the prediction value along with a confidence score. The Prediction results Page can be viewed in the figure 14 below.

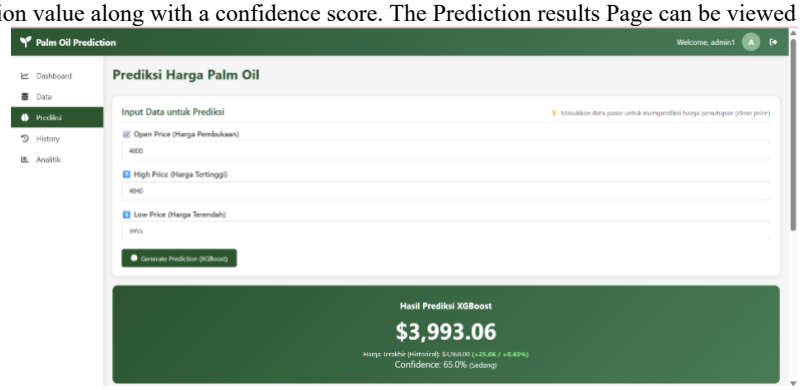

Fig. 14: Prediction Results Page

5. History Page

This page presents complete details of each prediction stored in the system as prediction history statistics. This page is only accessible to users who have logged in with credentials. The history page is shown in figure 15.

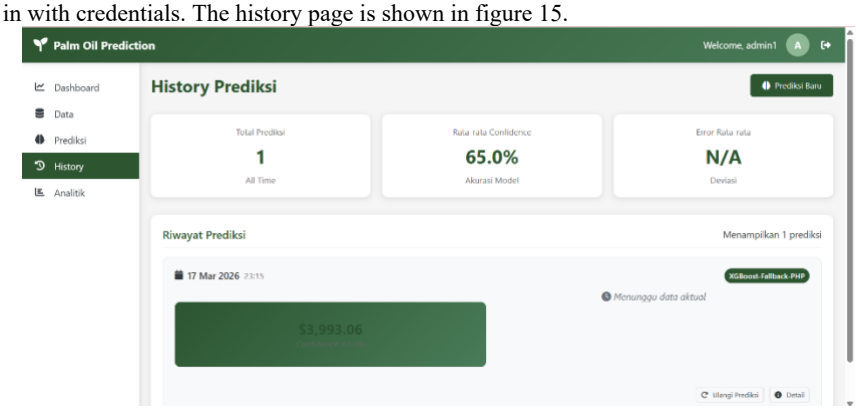

Fig. 15: History Page

6. Analytics Page

A page for the public (guests) to view a line chart visualization rendered using Chart.js comparing actual and predicted prices, along with model performance evaluation metric values. The analytics page is shown in figure 16 below.

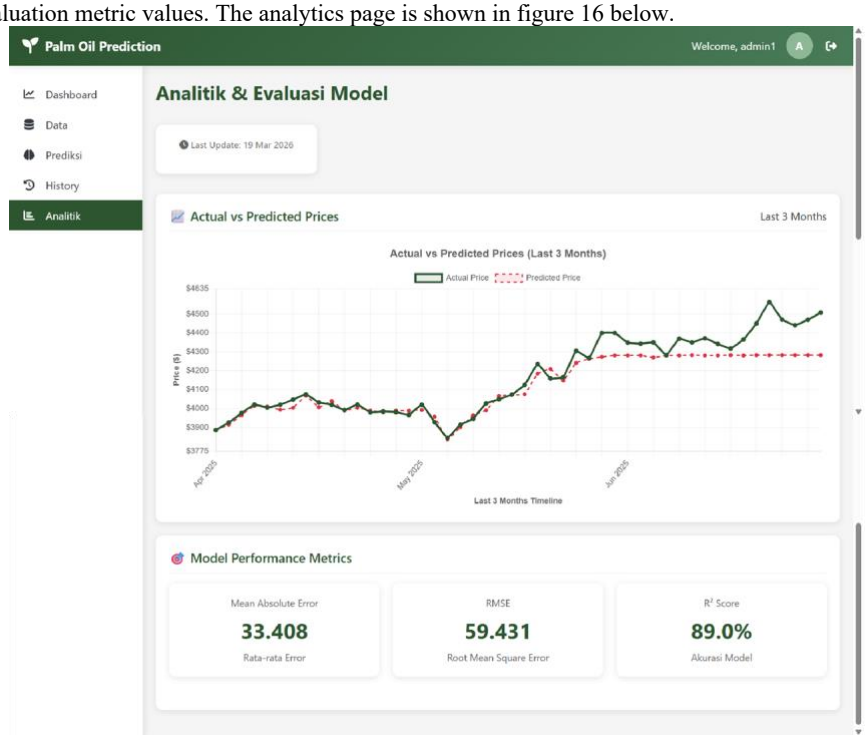

Fig. 16: Analytics Page

5. Conclusion

Based on the analysis and discussion conducted, it can be concluded that the XGBoost algorithm is suitable for predicting prices using features derived from historical data. The web-based system was built using a PHP frontend, Flask API backend, and MySQL database.

Hyperparameter optimization using Grid Search Cross-Validation was proven to improve model performance. The evaluation results—MAE of 43.974, RMSE of 65.145, and R^2 of 91.82%—demonstrate good accuracy consistent with standard commodity price prediction criteria.

Future research is recommended to develop the system in mobile form to enhance accessibility, integrate real-time data through commodity market APIs, enrich the model by adding external variables such as trade policies, weather, and geopolitics, and explore deep learning models such as LSTM for handling long-term predictions.

Acknowledgement

The author expresses gratitude and praise to Almighty God for His blessings and assistance, enabling the completion of this thesis as one of the requirements to fulfill the Bachelor of Informatics Engineering program at STMIK TIME.

The author extends sincere thanks to the thesis advisors, institution leadership, all lecturers, parents and family, as well as colleagues who have provided support and assistance throughout the thesis preparation process.

References

[1] Y. I. M. E. U. Supriyanto, "Prediksi Harga Minyak Kelapa Sawit Menggunakan Linear Regression Dan Random," *J. Ilm. Wahana Pendidik.*, vol. 8, no. 7, pp. 178–185, 2022, doi: 10.5281/zenodo.6559603.

[2] S. D. Oktarina, R. Nurkhoiry, R. Amalia, I. Pradiko, and S. Rahutomo, "Dampak Ketidakpastian Covid-19, Iklim, Dan Kompleksitas Lainnya Pada Industri Kelapa Sawit," *War. Pus. Penelit. Kelapa Sawit*, vol. 27, no. 2, pp. 70–77, 2022, doi: 10.22302/iopri.war.warta.v27i2.83.

[3] F. Suroso, G. M. Rahmah, and D. R. A. Permana, "Implementasi Sistem Peramalan Kebutuhan Spare Part Mobil Dengan WMA," *J. Teknol. dan Manaj.*, vol. 21, no. 2, pp. 113–122, 2023, doi: 10.52330/jtm.v21i2.136.

[4] Z. I. Bimawan, T. Astuti, and P. Arsi, "Comparison of Random Forest, K-Nearest Neighbor, Decision Tree, and Xgboost Algorithms for Detecting Stunting in Toddlers," *J. Tek. Inform.*, vol. 5, no. 6, pp. 1599–1607, 2024, doi: 10.52436/1.jutif.2024.5.6.2629.

[5] R. Siringoringo, R. Perangin Angin, and B. Rumahorbo, "Model Klasifikasi Genetic-Xgboost Dengan T-Distributed Stochastic Neighbor Embedding Pada Peramalan Pasar," *J. TIMES*, vol. 11, no. 1, pp. 30–36, 2022, doi: 10.51351/jtm.11.1.2022672.

[6] M. Fajri and A. Primajaya, "Komparasi Teknik Hyperparameter Optimization pada SVM untuk Permasalahan Klasifikasi dengan Menggunakan Grid Search dan Random Search," *J. Appl. Informatics Comput.*, vol. 7, no. 1, pp. 14–19, 2023, doi: 10.30871/jaic.v7i1.5004.

[7] N. K. Tri Yulianto, Rhevi HS Putri, "ANALISIS PENGARUH HARGA CPO (CRUDE PALM OIL) DUNIA DAN PRODUKSI CPO (CRUDE PALM OIL) INDONESIA TERHADAP FLUKTUASI HARGA MINYAK GORENG CURAH INDONESIA," *J. Cakrawala Ilm.*, vol. 2, no. 2, pp. 741–748, 2022.

[8] F. Ekonomi and U. Tidar, "Analisis Hubungan Volatilitas Harga Crude Palm Oil, Volume Ekspor dan Nilai Tukar Indonesia," vol. 6, pp. 42–53, 2023.

[9] F. A. Lase, K. S. Zai, J. Berkat, and I. Jaya, "Analysis Of Raw Material Inventory Forecasting Using The Time Series Method In Achieving Profit In The Integrated Training And Skills Development Business In Gunungsitoli Analisis Peramalan Persediaan Bahan Baku Menggunakan Metode Time Series Dalam Mencap," vol. 4, no. 2, pp. 765–778, 2025.

[10] R. Fadianty and S. Sriani, "Penerapan Data Mining dengan Algoritma Regresi Linear Berganda Untuk Memprediksi Omset Penjualan Minyak Goreng," *Build. Informatics, Technol. Sci.*, vol. 6, no. 2, pp. 1191–1200, 2024, doi: 10.47065/bits.v6i2.5951.

[11] E. Ismanto and M. Novalia, "Komparasi Kinerja Algoritma C4.5, Random Forest, dan Gradient Boosting untuk Klasifikasi Komoditas," *Techno.Com*, vol. 20, no. 3, pp. 400–410, 2021, doi: 10.33633/tc.v20i3.4576.

[12] H. T. Wen, H. Y. Wu, and K. C. Liao, "Using XGBoost Regression to Analyze the Importance of Input Features Applied to an Artificial Intelligence Model for the Biomass Gasification System," *Inventions*, vol. 7, no. 4, 2022, doi: 10.3390/inventions7040126.

[13] K. Aqbar and R. A. Supomo, "Performance Analysis of LSTM and XGBoost Models Optimization in Forecasting Crude Palm Oil (CPO) Production at Palm Oil Mill (POM)," *Int. J. Comput. Appl.*, vol. 185, no. 17, pp. 37–44, 2023, doi: 10.5120/ijca2023922890.

[14] Z. Chen *et al.*, "Automated Agriculture Commodity Price Prediction System with Machine Learning Techniques," vol. 6, no. 2, 2021.

[15] M. Andriyani, S. Nurwilda, D. Zatusiva Haq, and D. Candra Rini Novitasari, "Prediksi Harga Beras Premium Tahun 2024 Menggunakan Metode Gradient Boosted Trees Regression," *J. Teknol. Inf. J. Keilmuan dan Apl. Bid. Tek. Inform.*, vol. 18, no. 2, pp. 75–84, 2024, [Online]. Available: https://doi.org/10.47111/JTIAvailableonlineathttps://e-journal.upr.ac.id/index.php/JTI