KOMPUTASI: JURNAL ILMIAH ILMU KOMPUTER DAN MATEMATIKA
VOL. 23 (2) (2026), 113-X125 p-ISSN: 693-7554, e-ISSN:2654-3990

Neural Network Based Multiple Linear Regression Model for Financial Profit Prediction

Sandy Budi Wirawan[1], Asep Denih[2], Syarif Hidayatullah[3]

[1,2,3] Department of Computer Science, Faculty of Mathematics and Natural Science, Pakuan University, Bogor, West Java, 16143, Indonesia

Abstract

Financial forecasting grounded in operational efficiency is crucial for any business. CV. Surya Cipta Estetica Mandiri still records their finances manually which results in inaccurate forecasting. This project uses a web-based Enterprise Resource Planning (ERP) system to create a profit prediction model based on a Single Layer Perceptron with Multiple Linear Regression. Deep learning models with many layers carry the risk of overfitting, especially with linear data. Unlike other models, the proposed SLP model has a simplified architecture and will not contain a hidden layer. Additionally, this will both make the model compute in less time and maintain the financial audits' transparency. The dataset consists of 3,127 daily transactions from January 2021 to April 2025. The model was optimized using the Adam algorithm with 80% of the data used for training and 20% for validation (hold-out). The model was able to achieve an R^2 of 0.9988, MAPE of 1.03%, MAE of 258,080, RMSE of 893,381, and an overall accuracy of 98.97%. We were also able to create a model that outperformed the standard OLS baseline when dealing with outliers. This research combines traditional regression with neural networks and creates a measure-based financial planning framework from proactive to reactive for the first time.

Keywords: *Neural Network, Machine Learning, Multiple Linear Regression, Predictive Model, Profit Prediction, Financial System*

1. Introduction

Financial forecasting plays a central role in supporting strategic planning, operational control, and resource allocation within modern organizations. As business environments become increasingly dynamic and competitive, companies are required to adopt data-driven approaches to improve accuracy in financial decision-making. Profit prediction, in particular, is essential because profit fluctuation is influenced by numerous factors such as operational efficiency, consumer behavior, and cost management. Without reliable forecasting tools, companies may face challenges in planning future operations, assessing financial stability, and identifying potential risks [1].

CV. Surya Cipta Estetica Mandiri currently manages its financial data manually, causing limitations in data analysis, inefficiencies in record-keeping, and inaccuracies in profit estimation. Manual processing also restricts the company's ability to identify financial patterns or anticipate fluctuations based on historical data. These constraints highlight the need for an automated and intelligent predictive system capable of generating accurate financial insights to support managerial decisions [2].

Machine learning methods have increasingly been utilized to enhance predictive accuracy in various business domains. Neural Networks, in particular, are known for their capability to learn relationships between variables through iterative weight optimization, making them suitable for regression and forecasting tasks [3]. Previous studies have successfully applied linear regression and Neural Network techniques for forecasting inventory movement, sales prediction, commodity pricing, and used-car price estimation [1], [4], [5], [6]. However, most research either relies solely on traditional regression models or uses limited datasets, resulting in reduced predictive robustness [7].

This study proposes a profit prediction model that integrates Multiple Linear Regression with a Single Layer Perceptron (SLP), providing a Neural Network-based optimization framework to enhance regression performance. The SLP functions as a linear neuron that adjusts weights iteratively based on the error gradient, enabling it to capture linear relationships more effectively compared to conventional regression [8]. The model is trained using 3,127 daily transaction records collected between January 2021 and April

*Corresponding author. *E-mail address: sandybudiwirawan@gmaill.com*
Received: 06 December 2025, Accepted: 25 July 2026 and available online 31 July 2026
DOI: https://doi.org/10.33751/komputasi.v23i2.68

2025, consisting of variables such as date, income, expense, and transaction category.

To ensure data quality, preprocessing steps include handling missing values, encoding categorical variables using label encoding, applying feature scaling using RobustScaler, and dividing the dataset into an 80:20 train–test split. The model is developed using the TensorFlow Sequential API and trained with 2,501 data points, while 626 data points are used for evaluation. The performance of the model is assessed using several evaluation metrics, including R^2, RMSE, MAE, and MAPE, to ensure completeness and reliability of the predictive output. The results demonstrate that the model performs exceptionally well, achieving an R^2 of 0.9988 and a MAPE of 1.03%, indicating highly accurate predictions [9].

The main scientific contribution of this study is to combine the ease of interpretation of the conventional Multiple Linear Regression (MLR) model with the optimization capabilities of the Single Layer Perceptron (SLP). While most current financial prediction systems still rely on deep learning models whose decision-flows are difficult to analyze (black-box) or traditional static regression models, this approach expressly formulates financial linear relationships such as the direct influence of income and expenses on net income into a more adaptive architecture. The strategy of combining neural network-based optimization methods with linear mathematical functions has proven to be effective in minimizing error rates, in line with findings in the development of previous adaptive predictive models [10].

In addition, this research contributes in the form of a scalable framework by integrating direct models into web-based *Enterprise Resource Planning* (ERP) systems. This integration results in practical, *real-time*, and data-driven decision support tools to facilitate the operational dynamics of Small and Medium Enterprises (SMEs). It also answers the practical needs of industry that are often overlooked in *purely theoretical machine learning research*.

The objective of this study is to provide a practical, accurate, and scalable predictive tool that supports financial planning and enhances decision-making within business environments.

2. Methods

In the development of this system, the authors applied the Waterfall method, which is part of the SDLC (Software Development Life Cycle) framework. The Waterfall model is a systematic and sequential approach to system development [11], allowing each phase to be executed in a structured and orderly manner. This approach ensures that the development process progresses smoothly while minimizing the possibility of errors at each stage. The Waterfall model consists of six phases Analysis, Design, Development, Testing, Implementation, and Maintenance-as illustrated in Figure 1.

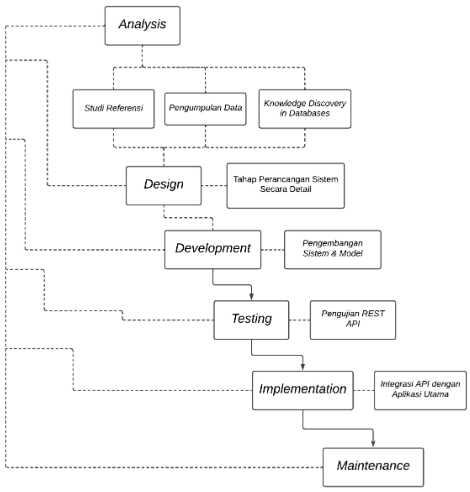


Figure 1. System Development Life Cycle (SDLC) using the Waterfall model

2.1 Analysis

The Analysis phase focuses on identifying system requirements and understanding the characteristics of the financial data used for machine learning-based profit prediction. This phase consists of three main activities: reference study, data collection, and analytical processing through the Knowledge Discovery in Databases (KDD) framework.

A. Reference Study

The reference study was conducted to explore relevant theories, models, and technologies related to profit prediction using Neural Networks and Multiple Linear Regression. Scientific journals, books, and credible literature sources were reviewed to strengthen the theoretical foundation and ensure the selection of an appropriate modeling approach.

B. Data Collection

Financial transaction data were collected through direct observation at CV. Surya Cipta Estetica Mandiri. The dataset contains 3,127 daily records from January 2021 to April 2025, stored in Excel

(.xlsx) format. Variables include date, transaction category, total income, total expense, and profit. A summary of the dataset is presented in Table 1.

Table 1. Dataset Description

Variable Name	Data Type	Description
Date	Datetime	Timestamp of the transaction
Category	String	Business or transaction category
Total Income	Float / Integer	Total incoming revenue
Total Expense	Float / Integer	Total outgoing expense
Profit	Float / Integer	Net profit (income - expense)

C. Knowledge Discovery in Databases (KDD)

The KDD framework is applied to analyze and prepare the dataset systematically. The process consists of five stages:

1) Data Selection
   In this stage, relevant attributes such as date, transaction category, income, and expense are selected to support the predictive modeling process. This selection ensures that only variables that directly contribute to profit calculation are included in the analysis.
2) Preprocessing
   This phase involves cleaning the data to address missing values, inconsistencies, and formatting issues, ensuring that the dataset is suitable for further analytical and modeling processes.
3) Transformation
   Data transformation is performed to convert the dataset into a format compatible with machine learning algorithms. This includes applying Label Encoding to categorical variables and using RobustScaler to normalize numerical features.
4) Data Mining
   At this stage, the modeling process is carried out using a Multiple Linear Regression algorithm based on a Single Layer Perceptron (SLP) to predict profit values based on historical patterns.
5) Interpretation and Evaluation
   The prediction results are evaluated using metrics such as MAE, RMSE, and R^2 to ensure that the model performs accurately and reliably within operational requirements.

The complete KDD process is illustrated in Figure 2.

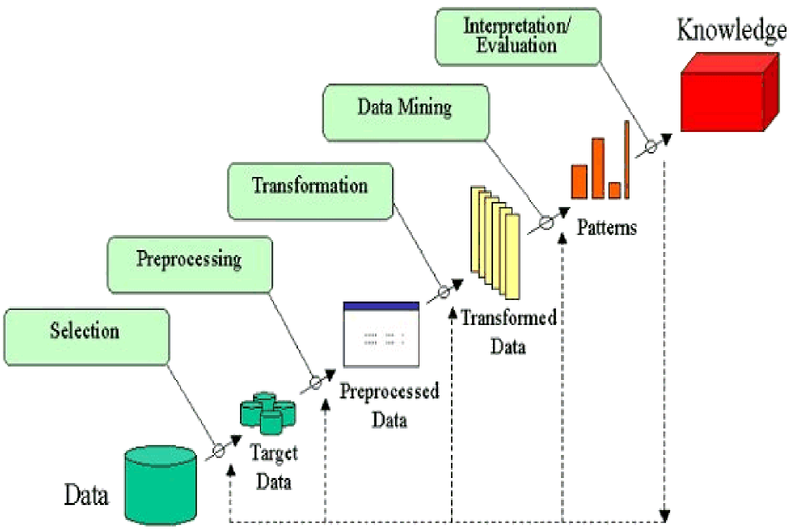


Figure 2. Knowledge Discovery in Database (KDD) process

A. Dataset Selection

The dataset consists of 3,127 daily financial transaction records from January 2021 to April 2025, obtained from CV. Surya Cipta Estetica Mandiri. Each record contains the following variables:

Table 2. Dataset Summary

Characteristics	Description
Total Records	3,127 daily financial transactions
Time Range	January 2021 - April 2025
Input Variables	Date, Category, Total Income, Total Expense
Output Variable	Profit
Training Data	2,501 records (79.98%)
Testing Data	626 records (20.02%)

The dataset represents real-world business operations, ensuring that the model is evaluated on realistic financial behavior.

B. Data Preprocessing and Transformation

Data preprocessing was conducted to ensure data quality and to enhance model performance. The following steps were applied:

Table 3. Data Preprocessing Steps

Preprocessing Step	Description	
Missing Value Handling	Inspection and imputation of incomplete records	
Label Encoding	Converts categorical transaction types to numeric values	
Feature Scaling	RobustScaler applied to reduce effect of outliers	
Data Split	80% Training	20% Testing

These preprocessing procedures ensure that the input features are suitable for Neural Network training.

C. Data Mining

In the data mining stage, the neural network architecture design is tightly adjusted to function purely as a Multiple Linear Regression model. Therefore, this architecture is built specifically as a Single Layer Perceptron (SLP) without using a hidden layer. Specifically, the input layer consists of several neurons that represent the predictor variables of the extraction results (such as nominal income, expenses, and transaction categories). All of these input neurons connect directly to a single output neuron that uses a linear activation function. The decision to eliminate hidden layers and non-linear activation functions was deliberate. This approach ensures that the model does not create complex representations of abstractions, but rather purely optimizes linear weights using the Adam optimizer. Thus, the final results of the model training can be extracted and read directly as the coefficients of regression equations, maintaining the transparency of the logic (explainability) required in financial system audits.

Given that the model is trained using thousands of daily transaction data points, overfitting mitigation strategies are the main focus in the testing phase. The separation of datasets with an 80:20 ratio (training and testing) is used as empirical validation (hold-out validation) to measure the model's generalization ability against previously unseen data. During training, the loss curves on the training and test data are monitored simultaneously at each epoch. The convergence that goes hand in hand between the two curves, without any wide divergence, confirms that the model manages to learn the basic patterns of transaction variance without simply memorizing noise or data isolation.

A Sequential Neural Network architecture was implemented using TensorFlow to enhance the capability of Multiple Linear Regression. The model consists of Table 4.

Table 4. Model Training Configuration

Parameter	Value
Architecture	Sequential Neural Network
Input Layer	5 neurons
Hidden Layers	2 hidden layers (ReLU activation)
Output Layer	1 neuron (Linear activation)
Optimizer	Adam
Loss Function	Mean Squared Error (MSE)
Epochs	100
Batch Size	32

The training process used 100 epochs and a batch size of 32, allowing the model to learn optimal regression parameters.

D. Model Evaluation

Model evaluation was conducted using four standard regression metrics:

Table 5. Model Evaluation Metrics

Metric	Result
R^2	0.9988
MAPE	1.03%
MAE	258,080
RMSE	893,381
Overall Accuracy	98.97%

These metrics measure accuracy, error magnitude, and prediction consistency. Visualization techniques-including loss curves, actual-predicted comparison, and residual plots-were also used to interpret the model's performance.

R^2 (Coefficient of Determination) is an evaluation metric used to measure how well a regression model fits the data. Its value ranges from negative infinity to 1, where 1 represents a perfect fit. A higher R^2 indicates that the model can explain a larger portion of the variance in the target variable. The formula for R^2 is shown in Eq. (1).

$$R^2 = 1\frac{\Sigma(y-\hat{y})}{\Sigma(y-y')} \tag{1}$$

RMSE (Root Mean Squared Error) measures the magnitude of prediction errors produced by the model. RMSE values range from 0 to positive infinity, where a smaller RMSE indicates that the predicted values are closer to the actual values. The RMSE formula is presented in Eq. (2).

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(X_i - Y_i)^2} \tag{2}$$

MAPE (Mean Absolute Percentage Error) represents the average percentage error between predicted values and actual values. The MAPE value ranges from 0 to positive infinity, with smaller values indicating better model accuracy. Eq. (3) shows the MAPE formula.

$$MAPE = \frac{1}{m}\sum_{i=1}^{m}\left|\frac{Y_i - X_i}{Y_i}\right| \times 100\% \tag{3}$$

Interpretation of MAPE values is shown in Table 6.

Table 6. Interpretation of MAPE Values

MAPE (%)	Interpretation
< 10	Very accurate prediction
10 – 20	Good prediction
20 – 50	Reasonable prediction
> 50	Inaccurate prediction

2.2 Design

The Design phase outlines the overall structure of the system and describes how the components interact to support the profit prediction process. This stage provides a conceptual view of the system architecture to ensure that all functional elements of the application are clearly defined before development begins. The general system architecture used in this study is shown in Figure 3.

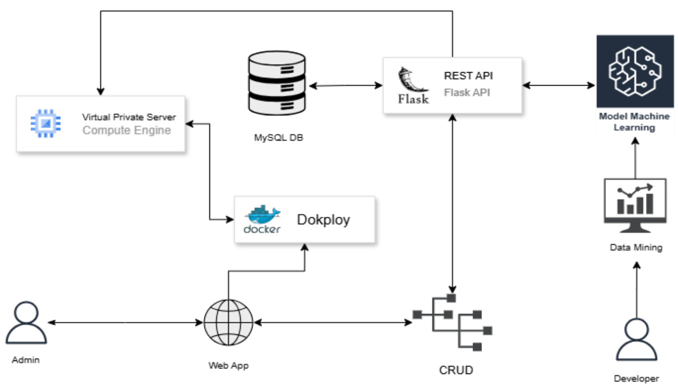


Figure 3. System architecture of the proposed financial prediction system

The detailed system design further specifies the internal components, data flow, and functional interactions within the system. This includes defining modules responsible for data input, preprocessing, model execution, and result visualization.

A Data Flow Diagram (DFD) is used to illustrate how data moves through the system and how each process transforms the data into meaningful output. The DFD describes the flow from user input to prediction results, ensuring a clear representation of data processing activities. The DFD used in this study is presented in Figure 4.

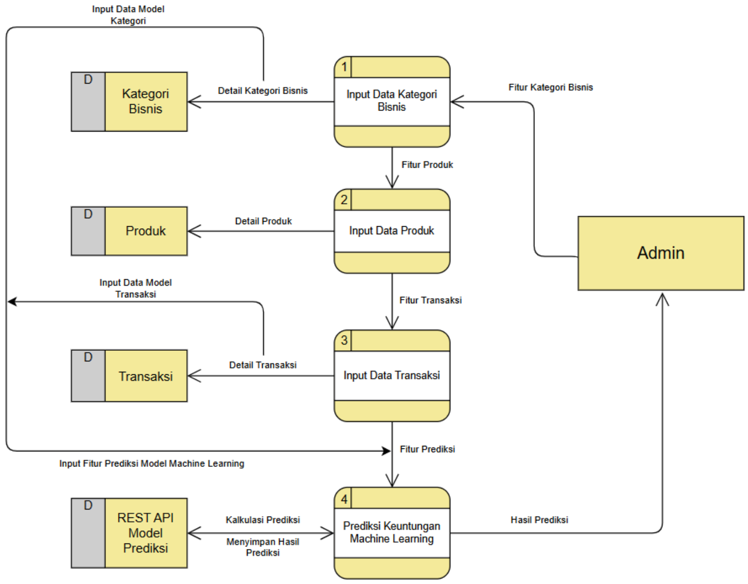


Figure 4. Data Flow Diagram (DFD) of the financial prediction system

2.3 Development

The Development phase focuses on implementing the system according to the design specifications. This stage includes coding the application modules, integrating the user interface with the database, and connecting the system to the machine learning model used for profit prediction. The development workflow and interactions between system components are illustrated in Figure 5.

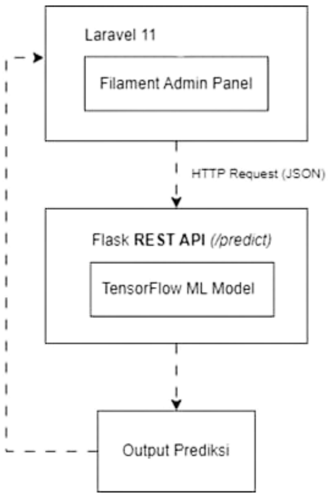


Figure 5. Development architecture diagram

2.4 Testing

The testing phase is conducted to ensure that all system components function correctly and consistently after the machine learning model has been fully integrated. The primary objective of this stage is to validate the accuracy of the profit prediction model and evaluate the stability of the REST API service implemented using the Flask framework. The testing workflow is illustrated in Figure 6.

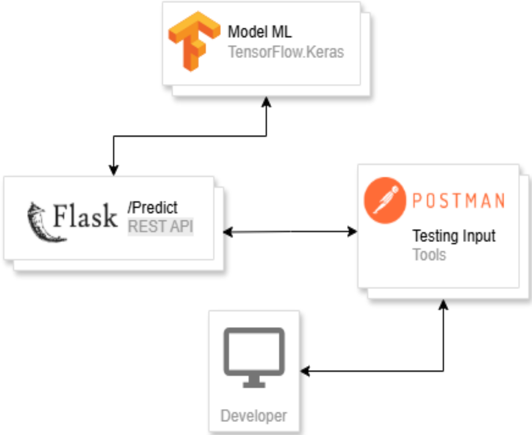


Figure 6. Testing Scenario

During testing, input data is submitted to the predict endpoint using Postman. The API processes the received data and forwards it to the machine learning model, which generates the prediction output. The resulting prediction is then examined to validate its accuracy and reliability. This process ensures that the model inference pipeline-including data transmission, API response handling, and system integration-operates without errors.

2.5 Implementation and Maintenance

The implementation stage involves deploying and integrating all major system components, including the backend, frontend, and the Neural Network–based Multiple Linear Regression model used for profit prediction. The overall implementation architecture is shown in Figure 7.

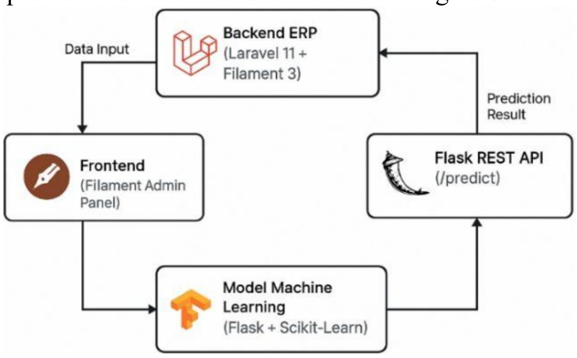


Figure 7. System Implementation Diagram

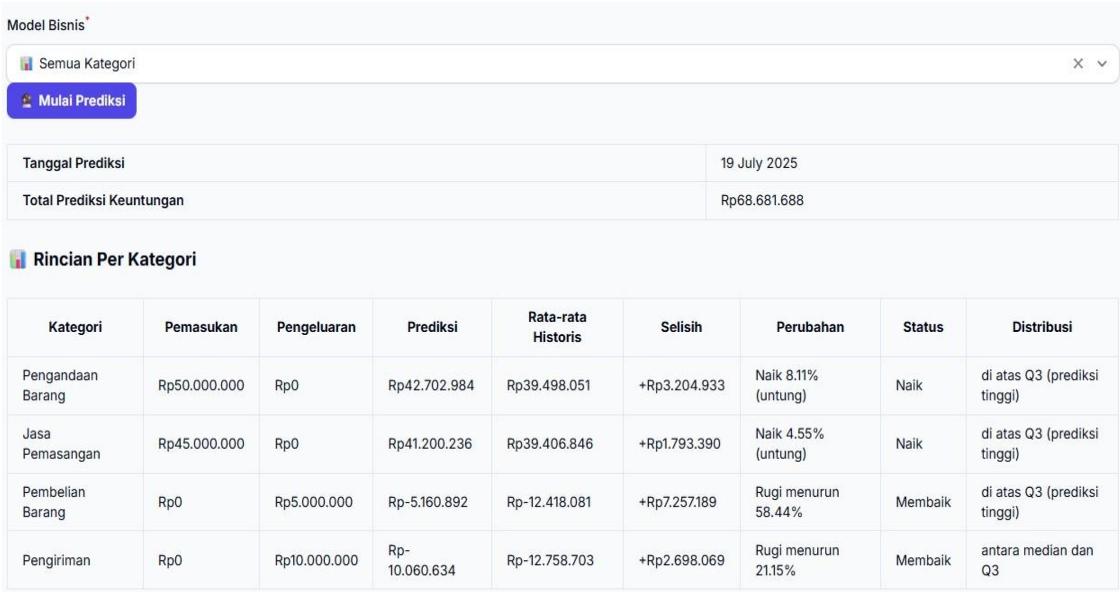
During implementation, the backend is developed using Laravel 11 combined with the Filament 3 plugin to support administrative interface management. The machine learning model is hosted via Flask and accessed using REST API endpoints. The frontend interacts with the backend and API services to deliver real-time prediction functionality within the ERP system.

Following deployment, the system enters the maintenance phase. Maintenance activities are carried out to ensure ongoing system performance and include bug fixes, periodic updates, monitoring of server resources, and improvements in model performance when new data becomes available. This continuous maintenance process ensures that the financial prediction system remains stable, accurate, and aligned with business operational needs.

3. Result and Discussion

This section presents the experimental results obtained from the developed predictive system and provides an in-depth discussion regarding model performance, data behavior, and the implications of the findings. The analysis focuses on the computational aspects of the Multiple Linear Regression-based Neural Network model, including preprocessing, training outcomes, evaluation metrics, and interpretative visualizations.

3.1 Experimental Results



Model Bisnis*

Semua Kategori

Mulai Prediksi

Tanggal Prediksi	19 July 2025
Total Prediksi Keuntungan	Rp68.681.688

Rincian Per Kategori

Kategori	Pemasukan	Pengeluaran	Prediksi	Rata-rata Historis	Selisih	Perubahan	Status	Distribusi
Pengandaan Barang	Rp50.000.000	Rp0	Rp42.702.984	Rp39.498.051	+Rp3.204.933	Naik 8.11% (untung)	Naik	di atas Q3 (prediksi tinggi)
Jasa Pemasangan	Rp45.000.000	Rp0	Rp41.200.236	Rp39.406.846	+Rp1.793.390	Naik 4.55% (untung)	Naik	di atas Q3 (prediksi tinggi)
Pembelian Barang	Rp0	Rp5.000.000	Rp-5.160.892	Rp-12.418.081	+Rp7.257.189	Rugi menurun 58.44%	Membaik	di atas Q3 (prediksi tinggi)
Pengiriman	Rp0	Rp10.000.000	Rp-10.060.634	Rp-12.758.703	+Rp2.698.069	Rugi menurun 21.15%	Membaik	antara median dan Q3

Figure 8. ERP System Prediction

The implemented prediction module generates daily profit estimations using historical records of income, expenses, and transaction categories. Figure 8 displays the prediction interface, which includes the total estimated profit and a detailed per-category analysis.

The system automatically computes:

1) Predicted values
2) Historical averages
3) Absolute differences
4) Percentage deviations
5) Status classification (increase/decrease)
6) Distribution analysis (quartile-based evaluation)

These outputs demonstrate that the system is capable of summarizing financial patterns and providing insights relevant to operational and strategic decision-making.

3.2 Model Performance Evaluation

The proposed predictive model uses the Single-Layer Perceptron (SLP) architecture to perform multiple linear regression with a linear activation function. Before being evaluated, the dataset first goes through the preprocessing and feature scaling stages and is then divided into training data and test data with an 80:20 ratio. A summary of the results of the performance test of this model is presented in Table 7 and Table 8.

Given that this model is trained using thousands of daily transaction records, overfitting mitigation strategies are the main focus of the evaluation phase. The 80:20 ratio of the dataset serves as an empirical validation (hold-out validation) to measure the model's generalization ability against previously unstudied data. During training, the loss curves for both training and test data are monitored simultaneously at each epoch. The harmonious convergence between the two curves, in the absence of wide divergence, confirms that the model has successfully learned the basic patterns of transaction variance, rather than simply memorizing noise or data outliers.

To strengthen the robustness analysis, we compared the Single-Layer Perceptron (SLP) with the standard Multiple Linear Regression (MLR) model based on Ordinary Least Squares (OLS) as a baseline. Although both methods rely in linear assumptions, SLP architectures optimized with Adam's algorithm show greater adaptability when handling large volumes in fluctuating financial data. The evaluation results showed that the SLP maintained a Mean Absolute Percentage Error (MAPE) of 1.03%. This performance significantly outperforms static MLR models, which tend to remain highly sensitive to residual effects of outlying data even after applying RobustScaler.

Table 7. Training and Testing Accuracy

Dataset	Accuracy	MAE
Training	99.01%	248,599
Testing	98.81%	295,960

The similarity between training and testing accuracy indicates stable model generalization with no overfitting. The low MAE further confirms that prediction errors fall within an acceptable operational margin.

Table 8. Error-Based Evaluation Metrics (10 Test Samples)

Metric	Value
R^2 Score	0.9997
MAE	277,687
MSE	1.69×10^{11}
RMSE	411,612
MAPE	1.26%
Accuracy	98.74%

The high R^2 and low MAPE indicate that the model successfully captures almost all variance in profit values, producing consistent predictions even under diverse transaction conditions.

3.3 Interpretative Analysis

This subsection provides a detailed interpretation of the model's behavioral characteristics by examining training dynamics, residual patterns, correlation structures, linear regression coefficients, and comparative predictive accuracy. These analyses support the reliability, interpretability, and practical applicability of the proposed Multiple Linear Regression-based Neural Network model.

A. Model Training Interpretation

The training curves for loss and MAE (Figure 9) show rapid convergence during the first 20 epochs for both the training and validation datasets.

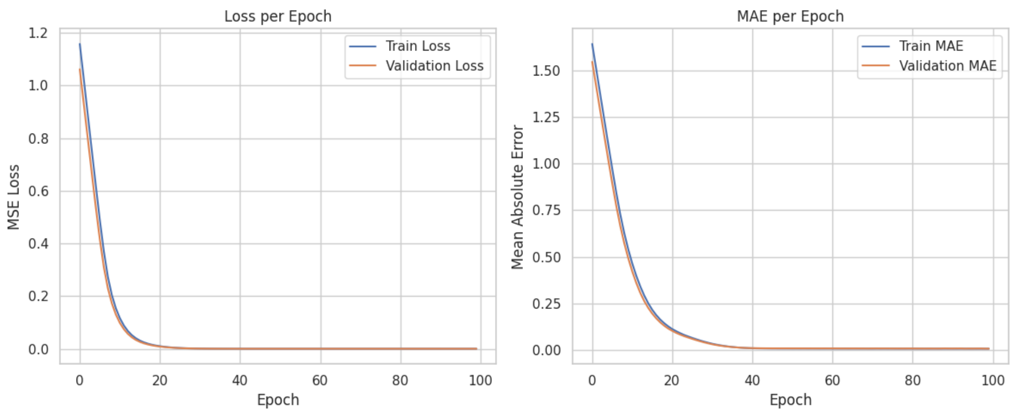


Figure 9. Training Curves for Loss and MAE

The close alignment between the two curves indicates that the model does not experience overfitting or underfitting. The combination of the Huber loss function and Adam optimizer contributes to robust convergence, particularly in the presence of financial outliers that naturally occur in transaction datasets.

B. Residual Distribution

Residual plots for both the training and testing sets are presented in Figure 10.

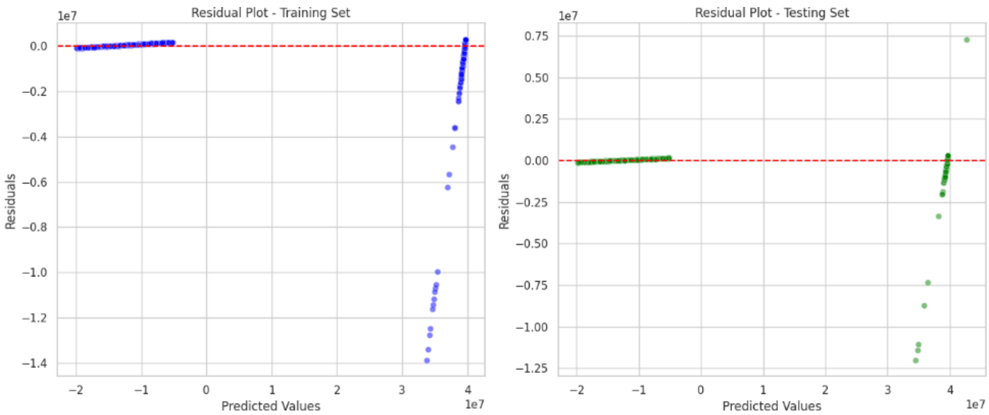


Figure 10. Residual Distribution for Training and Testing Sets

Most residuals cluster around zero, demonstrating unbiased prediction behavior. A small number of larger deviations are observed in samples with unusually high profit values, suggesting mild sensitivity to financial outliers. This behavior is typical in datasets with wide transactional variance. Despite these deviations, the overall residual pattern remains stable, confirming consistent generalization performance.

C. Correlation Analysis

The correlation matrix for the selected features is displayed in Figure 11.

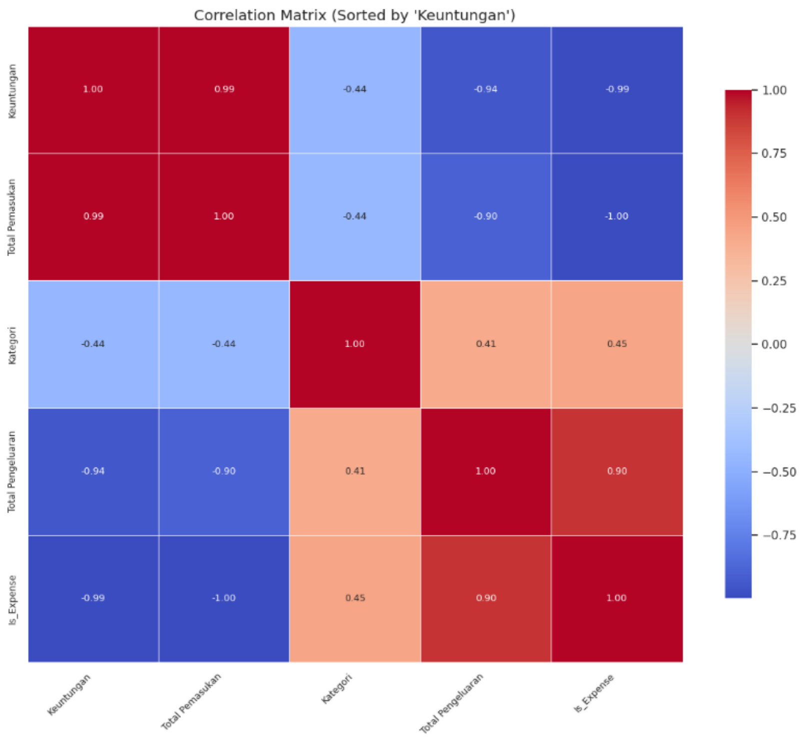


Figure 11. Feature Correlation Matrix

Key findings include:

1) Total Income → Profit: 0.99 (very strong positive correlation)
2) Total Expense → Profit: -0.94 (strong negative correlation)
3) *Is_Expense* → Profit: -0.99 (strong negative binary relationship)
4) Category → Profit: -0.44 (moderate negative correlation)

These results demonstrate that profit is highly influenced by linear relationships among financial variables, further justifying the use of a regression-based neural architecture.

D. Linear Regression Interpretation

The trained neural regression model produces the following prediction function:

$$y = 0{,}241741 + (0{,}226261).\ x1 + (-0.235270).\ x2 + (-0.533773).\ x3 + (0.000295).\ x4$$

Where:

A. Total Income *(x_1)*
B. Total Expense *(x_2)*
C. Is_Expense (x_3)
D. Category *(x_4)*

Positive coefficients (on income and category) indicate an increase in predicted profit, whereas negative coefficients (expense and expense-type indicators) decrease the predicted value. This aligns fully with financial logic and demonstrates that the model remains interpretable.

E. Comparison Between Actual and Predicted Values

Figure 12 presents a comparison of actual and predicted profit values across the training and testing samples.

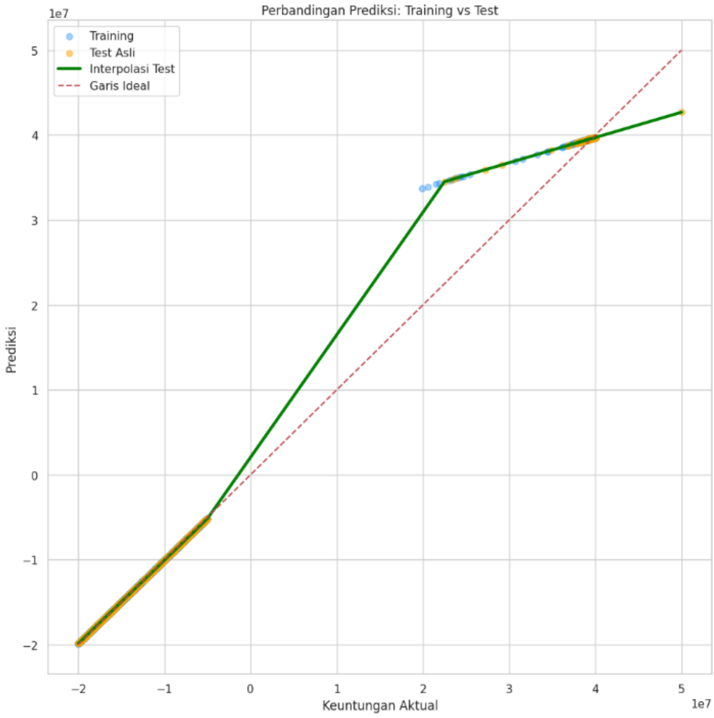


Figure 12. Actual vs Predicted Profit Comparison

The plotted points closely follow the ideal reference line, with the interpolation for the testing set showing a nearly parallel slope. This indicates that the model generalizes effectively and maintains strong predictive consistency across unseen data.

F. Linear Feature Relationships

Figure 13 visualizes linear trends between individual features and the target variable.

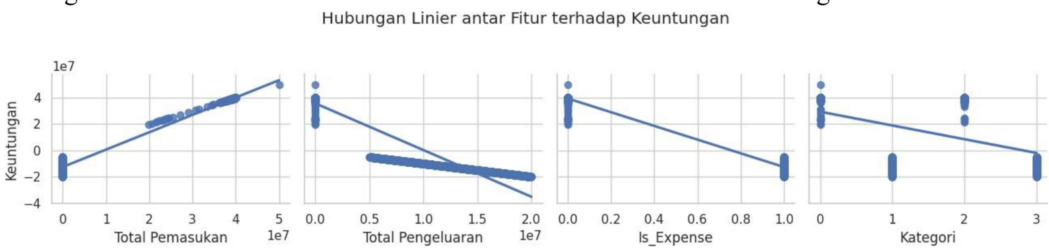


Figure 13. Linear Relationships Between Features and Profit Clear patterns emerge:

A. Total Income shows a strong positive linear relationship with profit.
B. Total Expense and Is_Expense exhibit strong negative slopes as expected.
C. Category presents a moderate effect, indicating differing profit tendencies across business types. These trends reinforce the interpretability of the model and validate the relevance of the selected features.

3.4 System-Level Interpretation

At the system level, the integrated ERP prediction module demonstrates consistent analytical behavior that aligns with observed historical patterns. Key observations include:

1) Revenue-generating categories exhibit upward predicted trends.
2) Expense-dominant categories naturally yield negative deviations, reflecting operational costs.
3) Quartile-based distribution indicators help managers determine whether predicted values fall within normal or unusual financial ranges.
4) Real-time computation enhances decision-making capacity by providing timely, data-driven financial insights.

This integration ensures that predictive analytics are directly actionable within the enterprise workflow.

3.5 Discussion

From a managerial point of view, the inclusion of this prediction model into ERP systems has a great impact on the company's financial planning going forward. Because management already holds accurate profit estimates, they no longer react when there is a problem, but can take anticipatory steps early. The budget can also be regulated more precisely, the operational cost ratio is more optimal, and cash flow is certainly more maintained. In short, data from this system can be a definite (data-driven) benchmark for formulating company targets, so there is no need for any more speculation in financial policy.

Although this Single Layer Perceptron (SLP)-based regression model shows a high level of accuracy on the current dataset, the linear modeling approach has inherent limitations that need to be acknowledged. This model assumes that the relationship between input variables and profit is always proportional and constant. As a result, these models have limitations in capturing extreme market anomalies, macroeconomic shocks, or seasonal patterns of consumer behavior that are non-linear in nature. If future business dynamics give rise to transaction complexity that can no longer be represented by linear correlation, this approach may experience a decrease in accuracy.

4. Conclusion

This research successfully develops and evaluates a profit prediction system that integrates Single-Layer Perceptron (SLP)-based Multiple Linear Regression models into web-based ERP applications. Using 3,127 financial history data points, the model achieved highly competitive performance, with $R^2 = 0.9988$, MAE = 258.080, RMSE = 893.381, and MAPE = 1.03%. By deliberately eliminating the hidden layer, the proposed architecture ensures computational efficiency and maintains the mathematical transparency that is strictly required in financial audits. In addition, rigorous hold-out validation confirms that the model effectively mitigates overfitting, successfully learning the basic patterns of transaction variance and not simply memorizing noise.

Crucially, the SLP model showed greater adaptability than the standard Ordinary Least Squares (OLS) baseline and was more robust to the residual effects of data outliers. The integration of these data-driven tools into the company’s workflow shifts management from reactive decision-making to proactive financial planning, optimising budget allocation and minimising speculation.

Despite having high accuracy on current datasets, linear modeling approaches have inherent limitations in capturing extreme market anomalies, macroeconomic shocks, or highly non-linear seasonal behaviors. Future research will need to explore the incorporation of non-linear algorithms, such as Random Forest or Gradient Boosting, to address increasingly complex transaction dynamics. In addition, future developments could extend this predictive framework into the broader ERP ecosystem, integrating modules for inventory forecasting and human resource management to provide deeper strategic insights.

References

[1] K. Puteri and A. Silvanie, “Machine learning untuk model prediksi harga sembako dengan metode regresi linier berganda,” *Jurnal Nasional Informatika*, vol. 1, no. 2, pp. 82–94, 2020, [Online]. Available: www.data.jakarta.go.id.

[2] O. Cecilia, R. Suri Meylinda, B. Alfi, and A. Richy Muhammad, “Analisis implementasi aplikasi konsep basis data relasional pada sistem pendapatan dan pengeluaran,” *Universitas Mercu Buana*, 2022.

[3] W. A. Amelia, “Implementasi jaringan syaraf tiruan sebagai sistem diagnosis stunting balita menggunakan metode backpropagation di Puskesmas Brebes,” *Jurnal Minfo Polgan*, vol. 12, no. 2, pp. 1874–1883, Oct. 2023, doi: 10.33395/jmp.v12i2.13044.

[4] O. J. Ababil, S. A. Wibowo, and H. Zulfia Zahro’, “Penerapan metode regresi linier dalam prediksi penjualan liquid vape di Toko Vapor Pandaan berbasis website,” *JATI (Jurnal Mahasiswa Teknik*

Informatika), vol. 6, no. 1, pp. 186–195, Mar. 2022, doi: 10.36040/jati.v6i1.4537.

[5] E. Hasibuan and A. Karim, "Implementasi machine learning untuk prediksi harga mobil bekas dengan algoritma regresi linear berbasis web," *Jurnal Ilmiah Komputasi*, vol. 21, no. 4, Dec. 2022, doi: 10.32409/jikstik.21.4.3327.

[6] P. Chang Hartono and A. Dwiyoga Widiantoro, "Analisis prediksi harga saham Unilever menggunakan regresi linier dengan RapidMiner," 2024. [Online]. Available: https://journal-computing.org/index.php/journal-cisa/index

[7] S. Prasetyo and T. Dewayanto, "Penerapan machine learning, deep learning, dan data mining dalam deteksi kecurangan laporan keuangan: A systematic literature review," *Diponegoro Journal of Accounting*, vol. 13, no. 3, pp. 1–12, 2024, [Online]. Available: http://ejournal-s1.undip.ac.id/index.php/accounting

[8] H. Todo, T. Chen, J. Ye, B. Li, Y. Todo, and Z. Tang, "Single-layer perceptron artificial visual system for orientation detection," *Front Neurosci*, vol. 17, Aug. 2023, doi: 10.3389/fnins.2023.1229275.

[9] N. P. Nur Fauzi, S. Khomsah, and A. D. Putra Wicaksono, "Penerapan feature engineering dan hyperparameter tuning untuk meningkatkan akurasi model random forest pada klasifikasi risiko kredit," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 12, no. 2, pp. 251–262, Apr. 2025, doi: 10.25126/jtiik.2025128472.

[10] Kotepuchai, P., & Limpiyakorn, Y. (2024). Machine Learning and SubstantivenAnalytical Procedure in Financial Audit. IJMRAP, 6(12), 169-173. http://ijmrap.com/wp-content/uploads/2024/06/IJMRAP-V6N12P114Y24.pdf

[11] A. D. Pujawan, R. A. Rendra, J. Arifin, and C. Agustin, "Design of information system vaccination report data logging web-based using waterfall method: Case study at Bandung Health Office," *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 9, no. 1, pp. 110–125, Mar. 2022, doi: 10.35957/jatisi.v9i1.1440.

.