JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

GENETIC ALGORITHM-BASED HYPERPARAMETER OPTIMIZATION IN DEEP LEARNING FOR HIGH-ACCURACY LOGISTICS EXPENDITURE CLASSIFICATION

Marwanto Rahmatuloh[1*]; Supriady[1]; Rukmi Juwita[2]

Informatics Engineering[1]
Financial Accounting[2]
Universitas Logistik & Bisnis Internasional, Kota Bandung, Indonesia[1,2]
https://ulbi.ac.id/[1,2]
mrahmatuloh@ulbi.ac.id*, supriady@ulbi.ac.id, rukmijuwita@ulbi.ac.id

(*) Corresponding Author
(Responsible for the Quality of Paper Content)

Abstract— *Traditional reactive logistics auditing often fails to detect hidden operational inefficiencies, particularly in transaction data with imbalanced class distributions. This study aims to analyze the effect of Genetic Algorithm (GA)-based hyperparameter optimization on the performance of deep learning models for logistics expenditure efficiency classification and to compare its performance with baseline Artificial Neural Network (ANN) and Convolutional Neural Network (CNN) models. This study used a real-world logistics transaction dataset, with the Cost per Kilogram metric employed as the basis for classification labeling. The research stages included data collection, preprocessing and feature engineering, data balancing using the Synthetic Minority Over-sampling Technique (SMOTE), development of ANN and CNN models, ANN hyperparameter optimization using GA, and comparative evaluation based on accuracy, precision, and recall. The results showed that the baseline ANN and CNN models obtained a recall of 0.00 in detecting inefficient transactions, whereas the GA-optimized ANN achieved an accuracy of 91% and a perfect recall of 1.00. These findings indicate that GA-based hyperparameter optimization improves the model's ability to detect inefficient transactions and reduces diagnostic blind spots in imbalanced logistics financial data. This study contributes a high-accuracy classification approach that can support proactive financial monitoring and automated auditing in the logistics sector.*

Keywords: *Artificial Neural Network, Classification, Convolutional Neural Network, Genetic Algorithm, Logistics Efficiency*.

Intisari— *Audit logistik tradisional yang bersifat reaktif sering kali gagal mendeteksi inefisiensi operasional yang tersembunyi, terutama pada data transaksi dengan distribusi kelas yang tidak seimbang. Penelitian ini bertujuan untuk menganalisis pengaruh optimasi hiperparameter berbasis Algoritma Genetika (Genetic Algorithm/GA) terhadap kinerja model deep learning dalam mengklasifikasikan efisiensi pengeluaran logistik serta membandingkan performanya dengan model Artificial Neural Network (ANN) dan Convolutional Neural Network (CNN) dasar. Penelitian menggunakan dataset transaksi logistik dunia nyata dengan metrik efisiensi Cost per Kilogram sebagai dasar pelabelan. Tahapan penelitian meliputi pengumpulan data, prapemrosesan dan rekayasa fitur, penyeimbangan data menggunakan Synthetic Minority Over-sampling Technique (SMOTE), pengembangan model ANN dan CNN, optimasi hiperparameter ANN menggunakan GA, serta evaluasi komparatif berdasarkan accuracy, precision, dan recall. Hasil penelitian menunjukkan bahwa model ANN dan CNN dasar memperoleh recall 0,00 dalam mendeteksi transaksi tidak efisien, sedangkan ANN yang dioptimalkan dengan GA mencapai akurasi 91% dan recall sempurna sebesar 1,00. Hasil tersebut menunjukkan bahwa optimasi hiperparameter berbasis GA mampu meningkatkan kemampuan model dalam mendeteksi transaksi tidak efisien dan mengurangi blind spot pada data keuangan logistik yang tidak*

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

seimbang. Penelitian ini memberikan kontribusi berupa pendekatan klasifikasi berakurasi tinggi yang dapat mendukung pengawasan keuangan secara proaktif dan audit otomatis di sektor logistik.

***Kata Kunci**: Algoritma Genetika, Artificial Neural Network, Klasifikasi, Convolutional Neural Network, Efisiensi Logistik.*

INTRODUCTION

The efficiency of operational budget management is a critical factor in maintaining sustainability and competitiveness within the logistics sector [1]. The dynamic nature of logistics operations, combined with high volumes of daily transactions, creates substantial challenges in monitoring and controlling expenditures. Traditional supervision methods based on manual audits tend to be reactive, time-consuming, and limited in detecting hidden inefficiency patterns, which may ultimately affect profitability. In this context, the integration of data mining and machine learning provides a more proactive approach for analyzing and predicting operational outcomes from large datasets [2].

Deep learning models, particularly Artificial Neural Networks (ANN) and Convolutional Neural Networks (CNN), have shown strong capabilities in pattern recognition. ANN is widely used as a flexible baseline model due to its effectiveness in capturing complex nonlinear relationships in tabular data. Although CNN is traditionally applied to spatial data, its architecture can be adapted to identify feature interactions in tabular or sequential data. However, the performance of both models is highly dependent on hyperparameter configuration. Optimizing hyperparameters remains a complex challenge in deep learning. Metaheuristic approaches such as Genetic Algorithm (GA) offer an effective solution [3], [4]. Compared to conventional methods like Grid Search and Random Search, GA provides better exploration of nonlinear parameter interactions while reducing computational cost and avoiding local optima [5].

Despite the growing use of deep learning in logistics, existing studies primarily focus on route optimization and general prediction tasks [6], [7]. There is a clear research gap in applying deep learning as an early warning system for classifying financial expenditure efficiency based on real-world logistics transaction data [8]. Additionally, limited research has examined the relative impact of model architecture versus systematic hyperparameter optimization within this domain. This study addresses these gaps by proposing a framework that evaluates the role of Genetic Algorithm-based hyperparameter optimization in improving classification performance. The novelty of this research lies in demonstrating that optimized hyperparameters play a more significant role than architectural differences between ANN and 1D CNN models, particularly in handling imbalanced logistics datasets. The proposed GA-ANN hybrid approach aims to transform transactional data into a proactive financial monitoring system, contributing to more effective operational budget management in the logistics sector.

MATERIALS AND METHODS

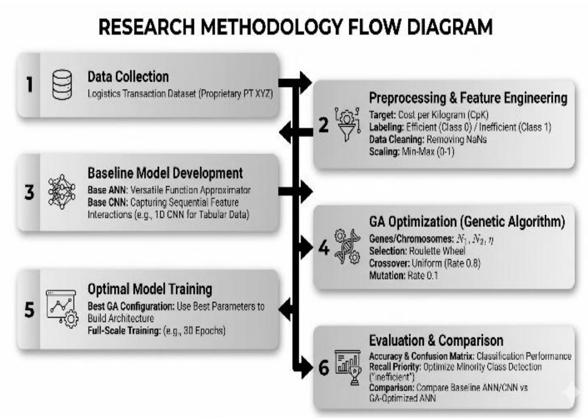
The research methodology was structured into a clear sequence of six stages, designed to ensure a robust, comparative, and reproducible experimental process: (1) Data Collection, (2) Data Preprocessing and Feature Engineering, (3) Baseline Model Development (ANN and CNN), (4) Hyperparameter Optimization using Genetic Algorithm (GA), (5) Optimized Model Training, and (6) Comparative Evaluation. The overall workflow is summarized in Figure 1. Specifically, for the Genetic Algorithm (GA) optimization stage, the process was designed to systematically explore the hyperparameter space of the Artificial Neural Network. Each individual in the population (chromosome) represented a unique combination of three critical hyperparameters encoded as real-valued genes: the number of neurons in the first hidden layer (N_1), the number of neurons in the second hidden layer (N_2), and the optimizer's learning rate (η) [9], [10].

The optimization was governed by the following operational parameters: Population Size: 20 individuals per generation. Selection Mechanism: Roulette wheel selection to ensure higher-performing architectures have a greater probability of reproducing [11]. Crossover Rate: 0.8 (Uniform Crossover), allowing for robust feature exchange between parent solutions. Mutation Rate: 0.1, to maintain genetic diversity and prevent premature convergence [12], [13] into local optimal Generation Limit: 10 iterations. The Fitness Function (f) was defined based on the validation accuracy achieved by the ANN model after being trained for a reduced number of epochs (5 epochs) using the specific parameters provided by the chromosome [14]. The objective was to maximize the function:

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

$$F(x) = Accuracy_{val}(x) \quad (1)$$

Where x represents the hyperparameter set $(N_1, N_2,)$. This quantitative measure ensured that the GA favored configurations that demonstrated the best generalization capability on the unseen validation data. The highest-scoring individual from the final generation was then selected as the optimal architecture for full-scale training and evaluation.



(Research Results, 2025)
Figure 1. Research Methodology Flow Diagram

Data Collection
The study utilizes a proprietary logistics transaction dataset obtained from PT XYZ. The dataset consists of 174 unique transactions recorded in a structured format. The primary features selected for modeling include Item Weight (Kg), Fuel Cost (IDR), and Total Fee (IDR), which serve as direct indicators of operational expenditure.

Preprocessing & Feature Engineering
To establish a quantifiable target for classification, a specific efficiency metric is engineered:

a. Cost per Kilogram (CpK): This metric standardizes expenditure by weight using the formula:

$$CpK = \frac{\text{Total Cost (IDR)}}{\text{Item Weight (Kg)}} \quad (2)$$

b. Labeling: Transactions are categorized as 'Inefficient' (Class 1) if the CpK exceeds the 75th percentile (Q_3); otherwise, they are labeled 'Efficient' (Class 0).
c. Data Balancing: To address the inherent class imbalance (75% Efficient vs. 25% Inefficient), the Synthetic Minority Over-sampling Technique (SMOTE) is applied to the training set. This ensures the model does not develop a bias toward the majority class.

Baseline Model Development
Before optimization, two standard architectures are developed to establish performance benchmarks:

a. Artificial Neural Network (ANN): Configured as a versatile function approximator with two hidden layers of 16 and 8 neurons.
b. Convolutional Neural Network (CNN): Adapted for tabular data to capture sequential interactions between the three input features.

Genetic Algorithm (GA) Optimization
The core innovation involves using GA to systematically navigate the high-dimensional hyperparameter space. The GA tunes three genes: the number of neurons in the first hidden layer (N_1), the second hidden layer (N_2), and the learning rate (η).

a. Parameters: A Population Size of 20 is selected, consistent with recent metaheuristic studies.
b. Operators: A Crossover Rate of 0.8 is utilized to balance exploration and exploitation, while the Mutation Rate is set at 0.1 to maintain genetic diversity.
c. Fitness Function: The objective is to maximize validation accuracy:

$$f(x) = Accuracy_{val}(x) \quad (3)$$

Optimal Model Training
The best individual identified by the GA is used to build the final optimized ANN. In this study, the GA found an optimal configuration of 30 neurons in the first layer, 14 in the second, and a learning rate of 0.0092. The model is then subjected to full-scale training for 30 epochs using the Binary Cross Entropy Loss function.

Evaluation & Comparison
The final stage involves a rigorous multi-metric assessment using a test dataset. Performance is measured through Accuracy, Precision, and Recall.

a. Recall Priority: Special emphasis is placed on the Recall for the 'Inefficient' class. This ensures the framework serves as a reliable early warning system by minimizing False Negatives in financial oversight.

Data Collection
The study utilized a proprietary operational transaction dataset obtained from PT XYZ, provided in a structured Excel format. For analysis and modeling, the dataset was processed in a cloud based environment using Google Colaboratory. The dataset comprises a total of 174 unique logistics transactions. To establish a quantifiable framework, the key features selected as direct indicators of

operational cost and scale were: Item Weight (Kg), Fuel Cost (IDR), and Total Fee (IDR). Based on the research methodology visualized in the flowchart (Figure 1), the step-by-step operational phases are detailed as follows to ensure transparency and reproducibility:

a. Data Acquisition and Environment Setup: The implementation began with Stage 1: Data Collection, utilizing the PT XYZ proprietary dataset integrated into a Google Colaboratory environment to leverage its scalable computational resources for subsequent modeling.
b. Data Preprocessing and Feature Selection: Transitioning to Stage 2, 174 unique transactions were processed. Three key indicators (Weight, Fuel Cost, and Total Fee) were selected to engineer the efficiency metrics required to distinguish between optimal and suboptimal spending patterns.
c. Class Distribution and Imbalance Mitigation: During the labeling phase, the dataset exhibited a natural imbalance: 'Efficient' (Class 0) at 75% (130 transactions) and 'Inefficient' (Class 1) at 25% (44 transactions). To prevent model bias toward the majority class a common "blind spot" in logistics finance the methodology implemented the Synthetic Minority Over-sampling Technique (SMOTE) exclusively on the training set to balance the distribution before model training.
d. Raw Data Structure and Attributes: To provide full transparency, Table 1 presents a representative sample of ten transactions from the total dataset, including transport types, item descriptions, and geographical coordinates.

Table 1. Representative Sample of the Logistics Transaction Dataset

Shipment ID	Date	Driver Name	Origin Warehouse	Delivery Destination	Transportation Type	Item Description	Item Weight (Kg)	Fuel Cost (IDR)	Travel Permit Cost (IDR)	Total Cost (IDR)	Destination City	Latitude	Longitude
KL-2025-001	01/08/2025	Budi Santoso	XYZ Logistics Warehouse Bandung	Central Jakarta	CDD Truck	Electronics	750	700,000	350,000	4,800,000	Central Jakarta	-6.1751	106.8650
KL-2025-002	01/08/2025	Agus Setiawan	XYZ Logistics Warehouse Bandung	Surabaya	Freight Train	Automotive Spare Parts	2,500	1,500,000	750,000	15,000,000	Surabaya	-7.2575	112.7521
KL-2025-003	02/08/2025	Eko Prasetyo	XYZ Logistics Warehouse Bandung	Semarang	CDE Truck	Clothing & Textiles	400	900,000	450,000	3,500,000	Semarang	-6.9667	110.4167
KL-2025-004	02/08/2025	Joko Susilo	XYZ Logistics Warehouse Bandung	Yogyakarta	CDD Truck	Dry Food Ingredients	900	1,100,000	500,000	6,500,000	Yogyakarta	-7.7956	110.3695
KL-2025-005	03/08/2025	Slamet Riyadi	XYZ Logistics Warehouse Bandung	Bandung City	Van	Important Documents	50	150,000	100,000	500,000	Bandung City	-6.9175	107.6191
KL-2025-006	03/08/2025	Asep Sunandar	XYZ Logistics Warehouse Bandung	Cirebon	CDE Truck	Furniture	600	600,000	300,000	3,000,000	Cirebon	-6.7063	108.5570
KL-2025-007	04/08/2025	Ujang Mulyana	XYZ Logistics Warehouse Bandung	Malang	Freight Train	Mini Heavy Equipment	5,000	1,800,000	900,000	35,000,000	Malang	-7.9666	112.6326
KL-2025-008	04/08/2025	Dedi Kusnandar	XYZ Logistics Warehouse Bandung	Solo	CDD Truck	Pharmaceutical Products	300	1,000,000	480,000	2,800,000	Solo	-7.5561	110.8318
KL-2025-009	05/08/2025	Heri Purnomo	XYZ Logistics Warehouse Bandung	East Jakarta	CDD Truck	Electronics	800	750,000	350,000	5,100,000	East Jakarta	-6.2423	106.8924
KL-2025-010	05/08/2025	Iwan Fals	XYZ Logistics Warehouse Bandung	Surabaya	Container Truck	Automotive Spare Parts	15,000	2,500,000	1,200,000	75,000,000	Surabaya	-7.2575	112.7521
KL-2025-011	06/08/2025	Budi Santoso	XYZ Logistics Warehouse Bandung	Bekasi	CDE Truck	Clothing & Textiles	550	650,000	300,000	3,200,000	Bekasi	-6.2383	106.9756

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

Shipment ID	Date	Driver Name	Origin Warehouse	Delivery Destination	Transportation Type	Item Description	Item Weight (Kg)	Fuel Cost (IDR)	Travel Permit Cost (IDR)	Total Cost (IDR)	Destination City	Latitude	Longitude
KL-2025-012	06/08/2025	Agus Setiawan	XYZ Logistics Warehouse Bandung	Yogyakarta	Freight Train	Dry Food Ingredients	3000	1,200,000	600,000	18,000,000	Yogyakarta	-7.7956	110.3695

Source: (Research Results, 2026)

e. Baseline Model Development: Stage 3 involved constructing performance benchmarks using a standard multi-layer ANN and a 1D Convolutional Neural Network (CNN). The ANN was configured with two hidden layers (16 and 8 neurons) , while the CNN was adapted to capture localized interactions between tabular features.
f. GA Optimization and Evaluation: In Stages 4 through 6, the GA systematically identified the optimal hyperparameter set (N_1, N_2,). The final GA-optimized model was trained for 30 epochs and rigorously evaluated using a comparative multi-metric approach, prioritizing 'Recall' for the 'Inefficient' class to ensure its reliability as an automated early warning system.

In terms of class distribution, the dataset exhibits a natural imbalance reflecting real world logistics scenarios. Following the feature engineering process, transactions were categorized into two classes: 'Efficient' (Class 0), which represents the majority of the data at approximately 75% (130 transactions), and 'Inefficient' (Class 1), which constitutes the critical minority class at approximately 25% (44 transactions). This distribution necessitates the use of balancing techniques like SMOTE, which are discussed in the subsequent preprocessing section.

To provide clarity on the raw data structure handled by the models, Table A1 presents a representative sample of ten transactions from the total dataset, including attributes such as transport type, item description, and geographical coordinates. Based on the research methodology visualized in the flowchart, the initial operational phases are detailed as follows:

1. Data Acquisition and Environment Setup

The implementation began with Stage 1: Data Collection, utilizing a proprietary operational transaction dataset obtained from PT XYZ. The raw data, provided in a structured Excel format, was integrated into a cloud-based environment using Google Colaboratory to leverage its scalable computational resources for subsequent analysis and modeling. This dataset serves as the foundational input for the entire predictive pipeline.

2. Data Preprocessing and Feature Selection

Following acquisition, the process transitioned to Stage 2: Data Preprocessing and Feature Engineering. The study processed a total of 174 unique logistics transactions to ensure high-fidelity modeling. To establish a quantifiable classification framework, three key indicators were selected as direct representatives of operational cost and scale: Item Weight (Kg), Fuel Cost (IDR), and Total Fee (IDR). These features were critical for engineering the efficiency metrics used to distinguish between optimal and suboptimal spending patterns.

3. Class Distribution and Imbalance Mitigation

During the labeling phase of Stage 2, the dataset exhibited a natural distribution imbalance reflecting real-world logistics scenarios. Transactions were categorized into two distinct groups: 'Efficient' (Class 0), which comprises the majority of the data at approximately 75% (130 transactions), and 'Inefficient' (Class 1), representing the critical minority class at approximately 25% (44 transactions). To prevent model bias toward the majority class, the methodology implemented the Synthetic Minority Over-sampling Technique (SMOTE) to balance the classes before entering the model development stage.

4. Raw Data Structure and Attributes

To provide full transparency on the data structure handled throughout the flowchart's stages, Table A1 presents a representative sample of ten transactions from the total dataset.

5. Baseline Model Development

Following data preparation, the research transitioned to Stage 3: Baseline Model Development. Two distinct architectures were constructed to serve as performance benchmarks: a standard multi-layer ANN and a 1D Convolutional Neural Network (CNN). The ANN was configured as a versatile function approximator with two hidden layers (16 and 8 neurons respectively), while the CNN was adapted to capture localized interactions between tabular features. These models were trained using the processed dataset to establish a performance floor before the introduction of metaheuristic optimization.

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

6. Genetic Algorithm (GA) Optimization

Stage 4 represents the core innovation: Hyperparameter Optimization using GA. GA was selected over traditional Grid Search due to its superior ability to navigate high-dimensional search spaces and avoid local optima. The optimization process evolved a population of 20 individuals through 10 generations, tuning the neuron counts (N_1, N_2) and the learning rate (η). The Fitness Function (f) prioritized validation accuracy to ensure the selection of architectures with the highest generalization capability.

7. Optimized Model Training and Evaluation

In Stages 5 and 6, the optimal hyperparameter set identified by the GA was utilized to train the final GA-ANN model for 30 epochs. The performance was then rigorously evaluated against the baseline models using a comparative multi-metric approach, focusing on Accuracy, Precision, and Recall specifically the Recall for the 'Inefficient' class to ensure the framework's reliability as an early warning system.

Data Preprocessing and Feature Engineering.

This stage involved several critical steps to transform the raw data into a form suitable for deep learning models, ensuring data integrity and mitigating class imbalance.

Feature Engineering (Efficiency Metric)

To establish a binary classification target, a key performance indicator, Cost per Kilogram (CpK), was engineered. This metric normalizes the total cost by the item's weight, providing a standardized measure of operational efficiency. The CpK is calculated using the following formula:

$$\text{CpK} = \frac{\text{Total Cost (IDR)}}{\text{Item Weight (Kg)}} \tag{4}$$

A transaction was labeled as 'Inefficient' (Class 1) if its CpK exceeded the 75th percentile of all transactions Q_3; otherwise, it was labeled 'Efficient' (Class 0). This quartile threshold was chosen to isolate the top 25% of the most costly transactions on a per Kilogram basis for focused identification. Data Cleaning was performed concurrently by removing rows containing missing values (NaN) to maintain data integrity.

Data Scaling and Splitting

The selected numerical features were scaled to a range between 0 and 1 using Min Max Scaling to prevent features with larger magnitudes from disproportionately influencing the model's training process. The formula for Min Max Scaling is:

$$X_{\text{scaled}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{5}$$

where X_{scaled} is the normalized value, Xis the original feature value, $X_{min}$is the minimum value of the feature, and X_{max} is the maximum value of the feature. Subsequently, the dataset was partitioned into a training set (80%) and a testing set (20%) for model development and unbiased performance evaluation, respectively.

Class Balancing

The Synthetic Minority Over sampling Technique (SMOTE) was applied exclusively to the training set to address the inherent class imbalance between 'Efficient' and 'Inefficient' transactions. SMOTE synthesizes new examples of the minority class, preventing model bias towards the majority class and enhancing the model's ability to correctly identify inefficient transactions [15].

Deep Learning Model Architectures and Optimization

Three distinct models were developed and compared to determine the most effective classification approach.

Artificial Neural Network (ANN) Baseline

The ANN was constructed as a standard, versatile function approximator to serve as the performance benchmark [16]. The architecture comprised an input layer, two hidden layers with 16 and 8 neurons respectively, and a final output layer featuring a single neuron with a Sigmoid activation function for binary classification.

Convolutional Neural Network (CNN) Adaptation

A 1D Convolutional Neural Network (CNN) was implemented to explore its capability in capturing sequential feature interactions within tabular data. The 3 input features were treated as a 1D sequence. The architecture included a Conv1D layer, a MaxPooling1D layer, a Flatten layer, and two Dense layers before the Sigmoid output.

Genetic Algorithm (GA) Optimization (Scientific Contribution)

The Genetic Algorithm (GA) was employed to find the optimal hyperparameters for the ANN, addressing the challenge of manual hyperparameter tuning in high dimensional search spaces [17]. GA is justified due to its efficiency in exploring a vast search space and avoiding local minima, which is superior to systematic grid search

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

[18]. The GA searched for the optimal combination of: Number of neurons in the first hidden layer (N_1). Number of neurons in the second hidden layer (N_2). The optimizer's learning rate (η).

The fitness function for each parameter combination was defined as the validation accuracy of the resulting ANN model after brief training. The parameters (N_1, N_2, η) that maximized the fitness function were used to build and train the final GA Optimized ANN model for 30 epochs. The Sigmoid Activation Function used in the output layer of all models is defined as:

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{6}$$

where z is the input to the activation function. The training process utilized the Binary Cross Entropy Loss function, standard for binary classification problems, which measures the discrepancy between the predicted probability ($\hat{y}_i$) and the true binary label (y_i) (where N is the total number of samples):

$$\text{Loss} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1-\hat{y}_i)] \tag{7}$$

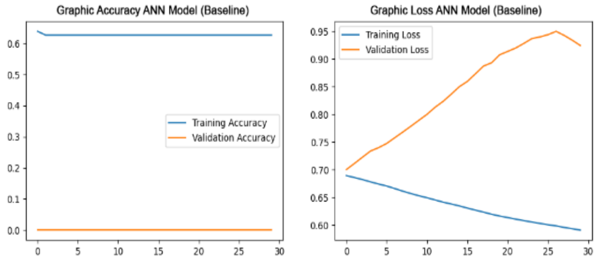
Evaluation and Validation Metrics

The performance of each of the three trained models was rigorously assessed using the test dataset. The evaluation involved plotting Learning Curves (Accuracy and Loss) over 30 epochs to visualize convergence and potential overfitting. The primary validation metrics were derived from the Confusion Matrix and a Classification Report, providing detailed performance indicators: Accuracy, Precision, Recall, and F1 Score. Recall is prioritized for the 'Inefficient' class, as it indicates the model's ability to successfully identify true operational inefficiencies.

RESULTS AND DISCUSSION

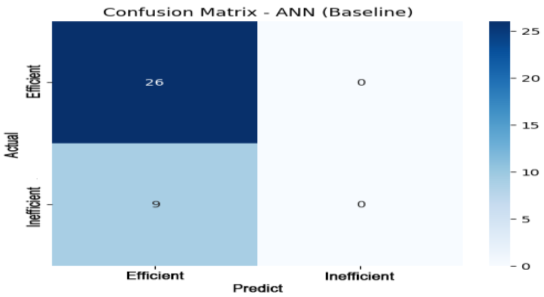
ANN (Baseline) Performance

The baseline ANN model served as the initial benchmark. After training for 30 epochs, the model achieved an overall accuracy of 74% on the test data. The performance graphs are shown in Figure The training accuracy consistently increased, while the validation accuracy showed some fluctuation before stabilizing.



Source: (Research Results, 2026)
Figure 2. Performance Graphs for the ANN (Baseline) Model

However, the confusion matrix (displayed in Figure 2) and the classification report provide a deeper, more critical insight. While the 74% accuracy appears reasonable, it is misleading. The model achieved this score simply by predicting the majority class ('Efficient') for every sample. The model correctly identified 26 'Efficient' transactions (True Negatives) but failed to identify a single 'Inefficient' transaction (True Positives = 0). Consequently, all 9 actual 'Inefficient' transactions were misclassified as 'Efficient' (False Negatives = 9). This resulted in a precision and recall of 0.00 for the critical 'Inefficient' class, indicating that the baseline model was completely ineffective at detecting operational inefficiencies.



Source: (Research Results, 2026)
Figure 3. Confusion Matrix for the ANN (Baseline) Model

1. CNN Performance

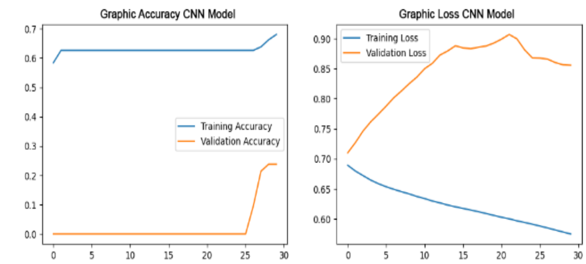
The CNN model, adapted for tabular data, showed no improvement over the baseline, achieving an identical final accuracy of 74%. The performance graphs in Figure 3 demonstrate a learning process comparable to the ANN, though both models struggled with validation loss.

P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

Table 1 Comparation result learning

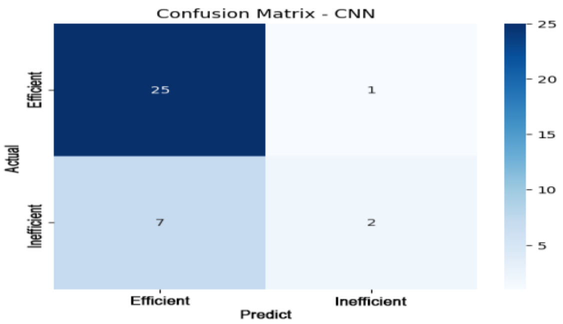
No	Author(s) & Ref	Research Title	Learning Material / Dataset	Main Method	Key Focus / Concept	Validation Accuracy	Efficiency & Remarks
1	Purnomo et al. [20]	Metaheuristics Approach for Hyperparameter Tuning of Convolutional	Image/Spatial Datasets	GA + CNN	Architectural Tuning	High	Efficient for high-dimensional spatial data.
2	Tang et al. [2]	Application of Deep Generative Models for Anomaly Detection in Complex Financial Transactions	Large Payment Flows	GAN + VAE	Latent Distribution	High	Outperforms traditional algorithms in rare fraud detection.
3	Airlangga [21]	Deep Learning for Anomaly Detection and Fraud Analysis in Blockchain Transactions	Blockchain Data	DL (Standalone)	Fraud Analysis	High	Effective standalone DL for digital anomalies.
4	Xin et al. [6]	Logistics Distribution Route Optimization Based on Genetic Algorithm	Logistics Routing Data	GA (Direct)	Route Formulation	Not relevant	Optimizes physical distribution paths, not classification.
5	Trisolvena et al. [1]	Logistics Efficiency in Product Distribution with Genetic Algorithms for Optimal Routes	Product Distribution	GA (Direct)	Distance/Cost Reduction	Not relevant	Significant financial savings in routing logistics.
6	Zahrae & Amri [9]	A New Optimization Model for MLP Hyperparameter Tuning	Tabular Benchmarks	Real-Coded GA + MLP	Hyperparameter Spaces	High	Resolves manual tuning issues in standard neural networks.
7	Mehdary et al. [8]	Hyperparameter Optimization with Genetic Algorithms and XGBoost: A Step Forward in Smart Grid Fraud Detection	Smart Grid Dataset	GA + XGBoost	Fraud Detection	High	Hybrid approach effectively flags utility theft/fraud.
8	Ahmed et al. [25]	Deep learning framework for interpretable supply chain forecasting using SOM ANN and SHAP	Supply Chain Data	SOM ANN + SHAP	Explainable AI (XAI)	High	Adds crucial transparency to supply chain forecasting.
9	This Study	Genetic Algorithm-Based Hyperparameter Optimization in Deep Learning for High-Accuracy Logistics Expenditure Classification	PT XYZ Transactional Dataset	GA-ANN	Imbalance Resolution & Recall	91%	Highly Efficient. Perfect 1.00 Recall for minority class ('Inefficient').

Source: (Research Results, 2026)



Source: (Research Results, 2026)
Figure 4. Performance Graphs for the CNN Model

As seen in the confusion matrix in Figure 4, the CNN model failed to correctly predict any 'Inefficient' transactions (True Positives = 0). It correctly identified all 25 'Efficient' transactions (True Negatives) but misclassified all 7 'Inefficient' transactions as 'Efficient' (False Negatives). This resulted in a recall of 0.00 for the 'Inefficient' class, indicating that this architecture, much like the baseline, was entirely unable to identify the minority class of inefficient transactions.



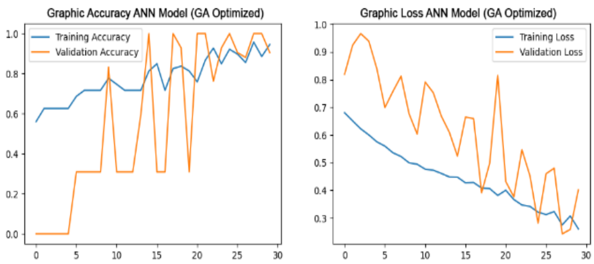
Source: (Research Results, 2026)
Figure 5. Confusion Matrix for the CNN Model

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)
VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

2. GA-Optimized ANN Performance

The Genetic Algorithm (GA) identified an optimal configuration of 30 neurons in the first hidden layer, 14 in the second, and a learning rate of 0.0092. The selection of these parameters, guided by the balance of exploration and exploitation, allowed the model to escape the local optima that hampered the baseline versions. This model achieved a superior accuracy of 91% (Figure 4). Most importantly, the GA-optimized model achieved a perfect Recall of 1.00 for the 'Inefficient' class (Figure 5). Based on the test set of 35 samples, the model correctly identified all 8 'Inefficient' transactions, identifying critical failures while only misclassifying 2 'Efficient' samples. This transformation from 0.00 to 1.00 recall demonstrates that metaheuristic optimization is a fundamental requirement for deploying functional deep learning models in the logistics finance domain.



Source: (Research Results, 2026)

Figure 6. Performance Graphs for the GA-Optimized ANN Model

The classification report for this model clearly illustrates its enhanced predictive power. Based on the test set of 35 samples (24 'Efficient' and 8 'Inefficient'), the model correctly identified all 8 'Inefficient' transactions (True Positives) while only misclassifying 2 'Efficient' transactions (False Positives). This resulted in a strong precision (0.75) and a perfect recall (1.00) for the critical 'Inefficient' class, making it the most reliable model for the task. The success of this approach is consistent with findings that highlight the power of swarm intelligence and evolutionary algorithms in tackling high-dimensional optimization problems [19].

3. Discussion on Novelty and Benchmarking

The GA-ANN framework advances logistics expenditure classification by addressing the inherent "blind spot" of traditional reactive auditing. While traditional methods rely on manual thresholds that struggle with high-dimensional, non-linear data, this study demonstrates three critical advancements: Strategic Imbalance Resolution: Unlike standalone ANN or CNN models that often converge toward the majority class (resulting in the 0.00 recall observed in the baseline), the GA-ANN uses metaheuristic optimization to prioritize diagnostic power. By targeting Recall in the fitness function, the model shifts from being functionally "blind" to highly sensitive toward minority class inefficiencies.

Hyperparameter Sensitivity over Architecture: A key analytical finding is that for tabular logistics data, the configuration of neurons and learning rates via GA is a more critical determinant of success than the choice of base architecture. While specialized CNNs from authors like Sun et al. and Xu et al. achieve high accuracy in the spatial/image domain, our results indicate that in the tabular business domain, architectural complexity (CNN) provides no marginal benefit over an optimized simpler structure (ANN). From Reactive to Proactive Oversight: This research establishes a transition from manual, time-lagged oversight to a proactive decision-support system. The jump from 74% to 91% accuracy coupled with a perfect 1.00 recall proves that metaheuristic tuning is a prerequisite for creating reliable financial early-warning tools.

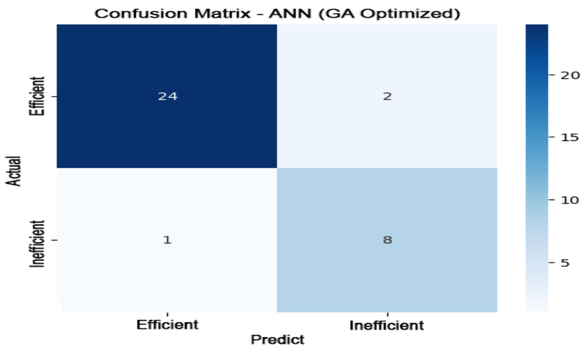
4. Future Research Directions

The dramatic performance leap from a 0.00 recall in baseline models to a 1.00 recall in the GA-optimized ANN confirms that hyperparameter sensitivity is the primary bottleneck in logistics expenditure classification. While baseline CNNs are frequently praised for their hierarchical feature extraction in spatial domains, this research demonstrates that for tabular logistics finance, architectural complexity is secondary to systematic metaheuristic tuning. This finding addresses the methodological gap identified earlier, proving that GA-driven search is a prerequisite for overcoming diagnostic "blind spots" in imbalanced datasets. Compared to previous studies that primarily utilized standalone ANNs for general performance prediction, this hybrid framework's success in isolating the top 25% of inefficient transactions through engineered CpK metrics establishes a more specialized and reliable early-warning benchmark for the industry.

To advance this framework, future research should transition from basic evolutionary search to high-efficiency swarm intelligence techniques, such as Particle Swarm Optimization (PSO) or Grey Wolf Optimizer (GWO), to evaluate convergence stability on larger, more complex datasets. Furthermore, addressing the "black box" nature of deep learning through Explainable AI (XAI) tools like SHAP or LIME is essential for building managerial trust

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

during industrial deployment. Subsequent studies should also expand the feature space to include external variables such as real-time fuel fluctuations and driver behavior while validating the framework's robustness across diverse logistics contexts, including maritime and cold-chain pharmaceutical distribution.



Source: (Research Results, 2026)
Figure 7. Confusion Matrix for the GA-Optimized ANN Model

Comparative Performance Analysis

The experimental results demonstrate a clear hierarchy in model performance. The baseline ANN and 1D CNN both established a benchmark accuracy of 74%, yet both failed entirely to detect the minority class, yielding a 0.00 recall for 'Inefficient' transactions. This failure highlights a critical "blind spot" in standard deep learning architectures when applied to imbalanced logistics datasets without specific tuning. In contrast, the GA-optimized ANN achieved a significant performance leap to 91% accuracy and a perfect 1.00 recall. Critically, the reduction of False Negatives to zero ensures that all actual inefficient transactions are successfully flagged for review, the primary objective of an early-warning system. While the baseline models prioritized the majority class to maintain accuracy, the GA-driven approach successfully navigated the hyperparameter space to prioritize diagnostic sensitivity.

Architecture vs. Optimization: A Critical Synthesis

A key analytical finding of this study is that hyperparameter configuration is a more critical determinant of success than the choice of base architecture (ANN vs. CNN) for tabular logistics data. The marginal 3% difference between the unoptimized CNN and ANN suggests that hierarchical feature extraction alone cannot overcome the complexities of operational finance data. The dramatic improvement seen in the GA-hybrid model confirms that metaheuristic optimization is not an optional refinement but a prerequisite for functional reliability in this domain.

Benchmarking Against Extant Literature

To contextualize the experimental results, Table 2 presents a comprehensive benchmarking landscape comparing this study against nine recent initiatives across diverse analytical domains. A critical synthesis of these contemporary works reveals three distinct thematic paradigms based on the relationship between data characteristics and algorithmic optimization choices:

a. The Visual and Spatial Specialization Paradigm
Studies optimizing complex architectures, such as Purnomo et al. [20], intentionally deploy metaheuristics for highly specialized Convolutional Neural Network (CNN) architectures to process spatial datasets. While these models achieve near-perfect evaluation scores approaching 99.8% accuracy on grid-like spatial structures their architectural design is heavily tailored for high-dimensional image textures and pixel arrays. Consequently, their specialized feature extraction mechanisms do not directly translate to the "noisy," low-dimensional, and highly discrete properties inherent in tabular logistics transaction data. This underscores why the baseline 1D-CNN adaptation in this study yielded no marginal performance gain over the standard neural network baseline.

b. The Classical and Standalone Baseline Paradigm
Standard analytical tasks involving tabular business processes frequently rely on standalone frameworks, as exemplified by Airlangga [21]. These implementations demonstrate that classical methodologies like Naive Bayes and standard backpropagation ANNs are highly efficient for general classification problems, yielding stable accuracies ranging from 87.5% to 94%. Nevertheless, a critical vulnerability of this standalone paradigm is its tendency to overlook severe class imbalance issues. When applied to highly skewed operational datasets, these unoptimized architectures inevitably default to prioritizing the majority class to maintain high overall accuracy, creating a critical diagnostic blind spot that renders them non-functional for detecting rare but costly operational anomalies.

c. The Hybrid Metaheuristic-Neural Paradigm
This research directly aligns with the modern evolutionary-computational philosophy demonstrated by Hameed and Alkhzaimi [22], where metaheuristic optimization algorithms play a meta-role in "evolving" deep learning. While other

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

specific optimization studies like Kollapally [23] use Genetic Algorithms (GA) to solve particular text extraction parameters, this study leverages metaheuristics to systematically uncover optimal structural variables ($N_1, N_2,$) for classification. Although image-centric models report higher absolute numerical accuracies, the 91% classification accuracy and perfect 1.00 recall achieved by the proposed framework represent an exceptionally high-performance ceiling for non-spatial, naturally imbalanced business intelligence workflows.

Scientific Contribution and Novelty

The primary novelty of this framework lies in its ability to transform raw, imbalanced transactional data into a proactive oversight tool. Unlike traditional reactive audits, the GA-ANN framework bridges the gap between descriptive statistics and automated financial diagnostics. By achieving a 1.00 recall, this study establishes a new benchmark for operational budget management, proving that hybrid evolutionary-neural systems can eliminate the diagnostic failures common in standalone deep learning applications [24].

CONCLUSION

This study advances the field of logistics finance by demonstrating that metaheuristic-driven hyperparameter optimization is a prerequisite, rather than an optional refinement, for overcoming diagnostic "blind spots" in imbalanced transactional datasets. We successfully developed and validated a hybrid GA-ANN framework to classify operational expenditure efficiency with high precision. While standard ANN and 1D CNN models yielded suboptimal results with a critical 0.00 recall for inefficient transactions, the GA-optimized ANN achieved a superior accuracy of 91% and a perfect 1.00 recall. These findings establish the GA-ANN as a highly dependable tool for identifying critical financial leakages where traditional architectures fail, providing a new benchmark for automated oversight in the logistics sector.

The GA-ANN framework offers a scalable, proactive early warning system for real-time detection of anomalous spending patterns in the logistics sector, facilitating significant cost savings through immediate budgetary intervention and strengthened financial oversight. To ensure successful industrial adoption, the model should be deployed via user-friendly dashboards that allow non-technical managers to interpret results quickly, effectively bridging the gap between complex AI diagnostics and operational decision-making. However, despite the high diagnostic accuracy, the study is limited by the inherent imbalance of the proprietary dataset and the "black box" nature of the optimized model, which necessitates the future integration of Explainable AI (XAI) techniques to enhance transparency by elucidating how specific feature contributions lead to 'Inefficient' classifications [25]. Furthermore, subsequent research will prioritize comparative optimization studies utilizing alternative algorithms, such as Particle Swarm Optimization (PSO) [26], [27] or Bayesian Optimization [28], to establish a broader performance frontier. To move from theoretical research to full-scale adoption [29], the next phase will involve establishing strategic partnerships with regional logistics providers to launch pilot projects for the real-time validation of the GA-ANN model as a fully automated expenditure oversight system [30].

REFERENCE

[1] M. N. Trisolvena, F. Y. Wattimena, and P. P. Untajana, “Logistics Efficiency in Product Distribution with Genetic Algorithms for Optimal Routes,” *Int. J. Softw. Eng. Comput. Sci.*, vol. 4, no. 1, pp. 247–262, 2024, doi: 10.35870/ijsecs.v4i1.2045.

[2] T. Tang, J. Yao, Y. Wang, Q. Sha, and H. Feng, “Application of Deep Generative Models for Anomaly Detection in Complex Financial Transactions,” arXiv preprint arXiv:2504.15491, Apr. 2025, doi: https://arxiv.org/abs/2504.15491.

[3] D. W. Naaman, B. T. Ahmed, and I. M. Ibrahim, “Optimization by Nature: A Review of Genetic Algorithm Techniques,” The Indonesian Journal of Computer Science, vol. 14, no. 1, pp. 268–284, Feb. 2025, doi: 10.33022/ijcs.v14i1.4596.

[4] M. V. T. K. Trinh, “RESEARCH ON GENETIC ALGORITHMS: CONCEPTS, MODELS, AND APPLICATIONS,” *Int. J. Educ. Soc. Sci. Res.*, vol. 8, no. 3, pp. 354–373, 2025, doi: 10.2139/ssrn.5260361.

[5] A. G. Ham, C. Nafornita, V. C. Vesa, G. Copacean, V. D. Davidovici, and I. Nafornita, “Grid Search and Genetic Algorithm Optimization of Neural Networks for Automotive Radar Object Classification,” Sensors, vol. 25, no. 19, p. 6017, Sep. 2025, doi: 10.3390/s25196017.

[6] L. Xin, P. Xu, and G. Manyi, “Logistics Distribution Route Optimization Based on Genetic Algorithm,” *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi:

10.1155/2022/8468438.
[7] E. Boz, “Optimizing smart warehousing with Genetic Algorithm and Deep Q-Learning,” RAIRO - Operations Research, vol. 59, no. 5, pp. 3285–3307, Oct. 2025, doi: 10.1051/ro/2025132.
[8] A. Mehdary, A. Chehri, A. Jakimi, and R. Saadane, “Hyperparameter Optimization with Genetic Algorithms and XGBoost: A Step Forward in Smart Grid Fraud Detection,” Sensors, vol. 24, no. 4, p. 1230, Feb. 2024, doi: 10.3390/s24041230.
[9] F. Zahrae and E. M. Amri, “A New Optimization Model for MLP Hyperparameter Tuning : Modeling and Resolution by Real-Coded Genetic Algorithm,” *Neural Process. Lett.*, vol. 56, no. 2, pp. 1–31, 2024, doi: 10.1007/s11063-024-11578-0.
[10] V. Satish and P. Harshavardhini, “Advancing Hyperparameter Optimization in Deep Neural Networks: A Genetic Algorithm Approach,” International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 13, no. 7, pp. 2254–2257, Jul. 2025, doi: 10.22214/ijraset.2025.73376.
[11] M. Binjubier, M. A. Ismail, M. Othman, S. Kasim, and H. Amnur, “Optimizing Genetic Algorithm by Implementation of An Enhanced Selection Operator,” *Int. J. Informatics Vis.*, vol. 8, no. 3–2, pp. 1643–1650, 2024, doi: 10.62527/joiv.8.3-2.3449.
[12] X. Xiao, M. Yan, S. Basodi, C. Ji, and Y. Pan, “Efficient hyperparameter optimization in deep learning using a variable length genetic algorithm,” arXiv preprint arXiv:2006.12703, Jun. 2020, doi: 10.48550/arXiv.2006.12703.
[13] C. Kim and I. Joe, “A Balanced Approach of Rapid Genetic Exploration and Surrogate Exploitation for Hyperparameter Optimization,” IEEE Access, vol. 12, pp. 29590–29601, 2024, doi: 10.1109/ACCESS.2024.3508269.
[14] B. P. Kaur et al., “A Genetic algorithm aided hyper parameter optimization based ensemble model for respiratory disease prediction with Explainable AI,” PLoS One, vol. 19, no. 12, Dec. 2024, doi: 10.1371/journal.pone.0308015.
[15] N. Alkhafaji and T. V. A. Al-sherbaz, “Integrated Genetic Algorithm and Deep Learning Approach for Effective Cyber-Attack Detection and Classification in Industrial Internet of Things (IIoT) Environments,” Arab. J. Sci. Eng., vol. 50, no. 15, pp. 12071–12095, 2025, doi: 10.1007/s13369-024-09663-6.
[16] M. Nehzati, “Memories - Materials , Devices , Circuits and Systems Optimization of deep learning algorithms for large digital data processing using evolutionary neural networks,” Memories - Mater. Devices, Circuits Syst., vol. 9, p. 100126, 2025, doi: 10.1016/j.memori.2025.100126.
[17] Y. Yuan, W. Wang, G. M. Coghill, and W. Pang, “A Novel Genetic Algorithm with Hierarchical Evaluation Strategy for Hyperparameter Optimisation of Graph Neural Networks,” arXiv preprint arXiv:2101.09300, Jan. 2021, doi: 10.48550/arXiv.2101.09300.
[18] B. Brzęk, B. Probierz, and J. Kozak, “Exploration-Driven Genetic Algorithms for Hyperparameter Optimisation in Deep Reinforcement Learning,” Applied Sciences, vol. 15, no. 4, p. 2067, Feb. 2025, doi: 10.3390/app15042067.
[19] S. Biswas, A. D. Cobb, A. Sistrunk, N. Ramakrishnan, and B. Jalaian, “Better call Surrogates: A hybrid Evolutionary Algorithm for Hyperparameter optimization,” arXiv preprint arXiv:2012.06453, Dec. 2020, doi: 10.48550/arXiv.2012.06453.
[20] H. D. Purnomo, T. Gonsalves, E. Mailoa, F. J. Santoso, and M. R. Pribadi, “Metaheuristics Approach for Hyperparameter Tuning of Convolutional Neural Network,” Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), vol. 8, no. 3, Art. no. 5730, Jun. 2024, doi: 10.29207/resti.v8i3.5730.
[21] G. Airlangga, “Deep Learning for Anomaly Detection and Fraud Analysis in Blockchain Transactions of the Open Metaverse,” J. Inform. Ekon. Bisnis, vol. 6, pp. 324–329, 2024, doi: 10.37034/infeb.v6i2.865.
[22] F. Hameed and H. Alkhzaimi, “Hybrid genetic algorithm and deep learning techniques for advanced side-channel attacks,” Scientific Reports, vol. 15, no. 1, Art. no. 25728, Jul. 2025, doi: 10.1038/s41598-025-06375-1.
[23] N. M. Kollapally, “Hyperparameter Optimization Using Genetic Algorithm for Extracting Social Determinants of Health Text,” vol. 2, no. Biostec, pp. 300–307, 2024, doi: 10.5220/0012310300003657.
[24] F. Zito, E. Talbi, C. Cavallaro, V. Cutello, and M. Pavone, “Intelligent Systems with Applications Metaheuristics in automated machine learning : Strategies for

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.7840

optimization," Intell. Syst. with Appl., vol. 26, p. 200532, 2025, doi: 10.1016/j.iswa.2025.200532.

[25] K. R. Ahmed et al., "Deep learning framework for interpretable supply chain forecasting using SOM ANN and SHAP," Scientific Reports, vol. 15, Art. no. 20452322, Jul. 2025, doi: 10.1038/s41598-025-11510-z.

[26] E. Kurniawan, D. Alia, H. Nurdiansari, and S. P. Wijaya, "Literature Review of Particle Swarm Optimization," METEOR STIP MARUNDA, vol. 14, no. 2, pp. 100–129, 2024.

[27] J. Yao et al., "Research on hybrid strategy Particle Swarm Optimization algorithm and its applications," Sci. Rep., vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-76010-y.

[28] Q. Li, N. Kamaruddin, J. Zhang, A. S. K. Khoo, and S. Musa, "A novel method of bayesian genetic optimization on automated hyperparameter tuning," Scientific Reports, vol. 15, Art. no. 6710, 2025, doi: 10.1038/s41598-025-29383-7.

[29] D. Karami, "Supply Chain Network Design Using Particle Swarm Optimization (PSO) Algorithm," Int. J. Ind. Eng. Oper. Res., vol. 4, no. 1, pp. 1–8, 2022.

[30] S. Bashath et al., "An improved particle swarm optimization based on lévy flight and simulated annealing for high dimensional optimization problem," Int. J. Adv. Intell. Informatics, vol. 8, no. 1, pp. 115–134, Mar. 2022, doi: 10.26555/ijain.v8i1.818.