Jurnal Teknik Informatika (JUTIF)
Vol. 5, No. 3, June 2024, pp. 931-939

DOI: https://doi.org/10.52436/1.jutif.2024.5.3.2109
p-ISSN: 2723-3863
e-ISSN: 2723-3871

COMPARISON OF DEEP LEARNING ARCHITECTURES FOR ANEMIA CLASSIFICATION USING COMPLETE BLOOD COUNT DATA

Gregorius Airlangga[*1]

[1]Information System Study Program, Engineering Faculty, Universitas Katolik Indonesia Atma Jaya, Indonesia
Email: [1]gregorius.airlangga@atmajaya.ac.id

(Article received: May 17, 2024; Revision: June 15, 2024; published: June 22, 2024)

Abstract

Anemia is a common condition marked by a deficiency in red blood cells or hemoglobin, affecting the body's ability to deliver oxygen to tissues. Accurate and timely diagnosis is essential for effective treatment. This study aims to classify different types of anemia using complete blood count (CBC) data through the application of deep learning models. We evaluated the performance of four deep learning architectures: Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Fully Connected Network (FCN). The dataset included CBC parameters such as hemoglobin, platelet count, and white blood cell count, labeled with anemia types. Our results indicate that CNN and FCN models achieved the highest test accuracies of 0.85, outperforming MLP and RNN models. This superior performance is due to the ability of CNN and FCN to capture complex patterns and spatial relationships within CBC data. Techniques like data augmentation and weighted loss functions were employed to address class imbalance. These findings demonstrate the potential of deep learning models to automate anemia diagnosis, thereby enhancing clinical decision-making and patient outcomes.

Keywords: *Anemia Classification, Complete Blood Count (CBC), Convolutional Neural Network (CNN), Deep Learning, Multi-Layer Perceptron (MLP), Recurrent Neural Network (RNN).*

1. INTRODUCTION

Anemia is a global health challenge that affects a significant portion of the population, with its prevalence particularly high in developing countries [1]–[3]. Characterized by a decrease in the number of red blood cells or hemoglobin concentration, anemia leads to reduced oxygen transport to tissues, resulting in various clinical symptoms and complications [4]–[6]. The common types of anemia include iron deficiency anemia, vitamin B12 deficiency anemia, and anemia of chronic disease [7]–[9]. Early diagnosis and classification of anemia types are crucial for effective treatment and management [10]. Traditional diagnostic methods rely heavily on manual interpretation of complete blood count (CBC) parameters, which can be time-consuming and prone to human error [11]. This study aims to leverage machine learning techniques, specifically deep learning models, to enhance the accuracy and efficiency of anemia diagnosis using CBC data [12].

The advent of machine learning has revolutionized many fields, including healthcare [13]. Machine learning algorithms can identify complex patterns in large datasets, providing insights that may not be apparent through conventional statistical methods [14]. In the context of anemia diagnosis, several studies have explored the use of machine learning to predict and classify anemia types based on CBC parameters [15]. For instance, traditional models like logistic regression, decision trees, and support vector machines have been applied to CBC data with varying degrees of success [16]. However, these models often require extensive feature engineering and may not capture the intricate relationships between features [17]–[19]. Recent advancements in deep learning have opened new avenues for biomedical data analysis [20]. Deep learning models, such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and fully connected networks (FCNs), have demonstrated remarkable performance in various medical applications, including image analysis, genomic data interpretation, and disease prediction [21]–[23]. These models can automatically learn feature representations from raw data, reducing the need for manual feature extraction and potentially improving predictive accuracy [24].

Despite the promising results of deep learning models in other medical domains, their application to anemia diagnosis using CBC data remains underexplored [25]–[27].This study aims to fill this gap by comparing the performance of four deep learning models—multi-layer perceptron (MLP), CNN, RNN, and FCN—in diagnosing and classifying anemia types. The primary goal is to evaluate the accuracy, robustness, and generalizability of these models on a labeled CBC dataset. The urgency of this research is underscored by the high global burden of anemia and the need for efficient diagnostic tools. Anemia affects over 1.62 billion people worldwide, with significant health and economic impacts. In

many low-resource settings, access to sophisticated diagnostic facilities is limited, making automated, accurate, and cost-effective diagnostic tools essential. By leveraging machine learning, we can potentially develop diagnostic models that are both accurate and accessible, contributing to improved health outcomes.

State-of-the-art machine learning techniques have shown potential in various aspects of medical diagnosis. For instance, CNNs have been widely used in medical image analysis for tasks such as tumor detection and segmentation [28]. RNNs, particularly long short-term memory (LSTM) networks, have been applied to time-series data for tasks like heart disease prediction [29]. FCNs have demonstrated their utility in genomic data analysis and other high-dimensional datasets [30]. Each of these models has unique strengths that can be leveraged to enhance anemia diagnosis [31]. For example, CNNs are adept at capturing spatial patterns, making them suitable for structured data like CBC parameters [32]. RNNs excel in capturing temporal dependencies, which can be beneficial if sequential CBC measurements are available [33]. FCNs are versatile and can handle various types of data, providing a robust framework for anemia classification [34]. The goal of this research is to develop and evaluate deep learning models for anemia diagnosis, comparing their performance with traditional machine learning models. We aim to identify the model that offers the best balance of accuracy, interpretability, and computational efficiency. By doing so, we hope to provide a comprehensive framework for anemia diagnosis that can be implemented in clinical practice.

A thorough literature survey reveals several gaps in the existing research on machine learning-based anemia diagnosis. Most studies focus on traditional machine learning models and do not explore the full potential of deep learning techniques. Additionally, many studies use small, homogenous datasets, limiting the generalizability of their findings [35]. This research addresses these gaps by using a larger, diverse dataset and comparing multiple deep learning models. The contributions of this research are manifold. First, we provide a detailed comparison of various deep learning models for anemia diagnosis, highlighting their strengths and weaknesses. Second, we propose a robust framework for evaluating these models, including metrics like accuracy, precision, recall, and F1-score. Third, we offer insights into the interpretability of deep learning models, which is crucial for clinical adoption. Finally, we discuss the practical implications of our findings, providing recommendations for future research and clinical practice.

The remaining structure of this journal article is organized as follows. Section 2 describes the dataset and preprocessing steps, including data augmentation and normalization techniques. Also, we outline the methodology, including the architecture and training of the deep learning models. Section 3 presents the results of our experiments, including performance metrics and visualizations. Section 4 discusses the findings, comparing the performance of different models and their practical implications. Section 5 concludes the article, summarizing the key contributions and suggesting directions for future research.

2. METHODOLOGY

This section describes the comprehensive methodology employed in this study, detailing the data acquisition and preprocessing, model selection and architecture, training procedures, evaluation metrics, and comparative analysis as presented in the Figure 1.

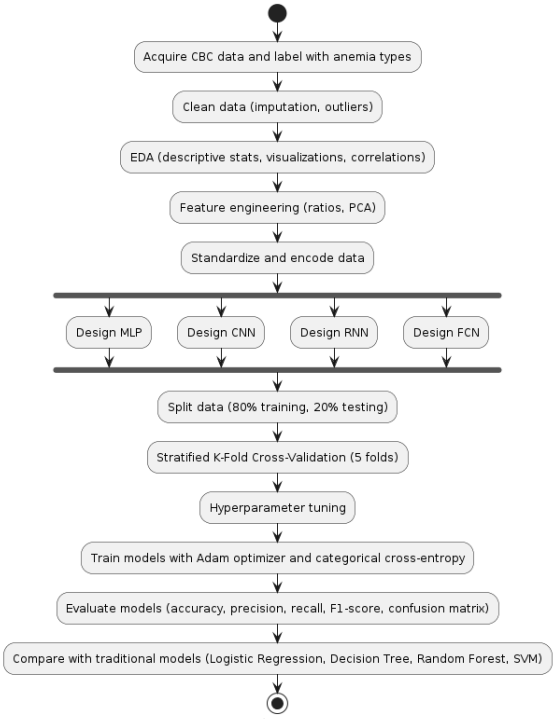

Figure 1. Diagram of Research Methodology

2.1. Data Acquisition and Preprocessing

The dataset used in this study consists of complete blood count (CBC) data, meticulously labeled with various anemia types. This data was sourced from multiple clinical settings, ensuring a diverse and comprehensive collection of samples. Each entry in the dataset was manually diagnosed by medical professionals, lending a high degree of accuracy and reliability to the labels. The dataset includes a range of hematological parameters that are crucial for diagnosing anemia. These parameters include Hemoglobin (HGB), which measures the amount of hemoglobin in the blood and is vital for oxygen transport; Platelets (PlT), which are essential for blood clotting; White Blood Cells (WBC), which are critical for the immune response; and Red Blood

Cells (RBC), which are responsible for oxygen transport throughout the body. Additional parameters include the Mean Corpuscular Volume (MCV), which indicates the average volume of red blood cells; Mean Corpuscular Hemoglobin (MCH), which measures the average amount of hemoglobin per red blood cell; and Mean Corpuscular Hemoglobin Concentration (MCHC), which assesses the average concentration of hemoglobin in red blood cells. The dataset also includes Platelet Distribution Width (PDW), which measures the variability in platelet size distribution in the blood, and Procalcitonin (PCT), a biomarker that can help diagnose sepsis and assess the risk of developing sepsis. The target variable in the dataset is the Diagnosis, which categorizes the type of anemia based on these CBC parameters.

Before proceeding with modeling, the dataset underwent an extensive cleaning process to address any missing values, outliers, and inconsistencies. Missing values were treated using appropriate imputation techniques; for numerical variables, this involved replacing missing values with the mean or median, while for categorical variables, the mode was used. Outliers were identified through visual inspection using boxplots and addressed by capping or flooring extreme values based on the interquartile range (IQR) method. This ensured that the data was as complete and accurate as possible, minimizing the potential for bias in the modeling phase. Following data cleaning, a thorough exploratory data analysis (EDA) was conducted to gain insights into the distribution and relationships among the variables. Descriptive statistics provided a summary of the central tendencies and dispersion of the features. Visualizations, such as histograms and boxplots, were employed to understand the distribution of each variable, while pair plots helped in examining the relationships between pairs of variables. Correlation analysis was also performed to detect multicollinearity among the features, using correlation matrices and heatmaps. This step was crucial to ensure that the selected features were not excessively correlated, as multicollinearity can adversely affect the performance of certain machine learning algorithms.

Feature engineering was then undertaken to create new features that could potentially enhance the predictive power of the models. For example, ratios like RBC to HGB and PlT to WBC were computed to capture additional insights into the hematological profile of the patients. These engineered features were designed to provide more nuanced information that might be critical for accurate diagnosis. Additionally, Principal Component Analysis (PCA) was applied to reduce the dimensionality of the dataset, extracting the most significant features while retaining the majority of the variance in the data. This step was particularly important to mitigate the curse of dimensionality and improve the model's performance. To prepare the data for modeling, label encoding was used to convert categorical variables into numerical format. One-hot encoding was applied to the target variable to facilitate multi-class classification, transforming each class into a binary vector. The dataset was then standardized using the StandardScaler, which ensured that all features had a mean of zero and a standard deviation of one. This standardization process is critical for the convergence of gradient-based optimization algorithms, commonly used in deep learning models, as it ensures that all features contribute equally to the learning process.

2.2. Model Selection and Architecture

This study evaluates the performance of four deep learning models: Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Fully Connected Network (FCN). Each model was carefully designed to leverage the unique characteristics of the CBC data and maximize the accuracy of anemia diagnosis.

2.2.1. Multi-Layer Perceptron (MLP)

The MLP model, a type of feedforward neural network, consists of an input layer, multiple hidden layers, and an output layer. Each layer is composed of neurons that are fully connected to the neurons in the previous and subsequent layers. This architecture is well-suited for handling tabular data like CBC features. The MLP model in this study includes an input layer that accepts the standardized CBC features, ensuring each feature is on the same scale. The hidden layers consist of two layers with 128 and 64 neurons, respectively, each utilizing ReLU activation functions to introduce non-linearity and enable the network to learn complex patterns. To enhance the model's robustness, batch normalization is applied after each hidden layer to stabilize and accelerate the training process. A dropout rate of 0.5 is used during training to prevent overfitting by randomly dropping neurons, which forces the network to learn more robust features. The output layer is a softmax layer with the number of neurons equal to the number of anemia classes, providing a probability distribution over the classes for multi-class classification.

The input layer takes standardized features ($X \in R^{m \times n}$), where (m) is the number of samples and (n) is the number of features. The input to the first hidden layer is $X = [x_1, x_2, \ldots, x_n]$. The first hidden layer with 128 neurons can be represented as $Z^{(1)} = W^{(1)}X + b^{(1)}$ and $A^{(1)} = \mathrm{ReLU}(Z^{(1)})$. Where ($W^{(1)} \in R^{128 \times n}$) is the weight matrix, ($b^{(1)} \in R^{128}$) is the bias vector. Then, ($Z^{(1)}$) is the linear transformation output. ($A^{(1)}$) is the activation output after applying the ReLU function.

Then, the ReLU activation function is defined as $\mathrm{ReLU}(z) = \max(0, z)$, and as a second Hidden

Layer with 64 neurons can be represented as $Z^{(2)} = W^{(2)}A^{(1)} + b^{(2)}$, and $A^{(2)} = \text{ReLU}(Z^{(2)})$ where $(W^{(2)} \in R^{64\times128})$ is the weight matrix, $(b^{(2)} \in R^{64})$ is the bias vector, $(Z^{(2)})$ is the linear transformation output, $(A^{(2)})$ is the activation output after applying the ReLU function. Furthermore, for batch normalization is applied after each hidden layer. For the first hidden layer, it is $\widehat{Z^{(1)}} = \frac{Z^{(1)}-\mu^{(1)}}{\sqrt{\sigma^{(1)2}+\epsilon}}$ and $\widetilde{Z^{(1)}} = \gamma^{(1)}\widehat{Z^{(1)}} + \beta^{(1)}$ where $(\mu^{(1)})and(\sigma^{(1)2})$ are the mean and variance of $(Z^{(1)})$ over the batch.

The (ϵ) is a small constant for numerical stability. Furthermore, $(\gamma^{(1)})$ and $(\beta^{(1)})$ are learnable parameters. For the second hidden layer, batch normalization is similarly applied $\widehat{Z^{(2)}} = \frac{Z^{(2)}-\mu^{(2)}}{\sqrt{\sigma^{(2)2}+\epsilon}}$ and $\widetilde{Z^{(2)}} = \gamma^{(2)}\widehat{Z^{(2)}} + \beta^{(2)}$. Dropout is applied during training with a dropout rate $(p = 0.5)$, $A^{(1)}_{\text{dropout}} = A^{(1)} \odot D^{(1)}$, $D^{(1)} \sim \text{Bernoulli}(p)$, $A^{(2)}_{\text{dropout}} = A^{(2)} \odot D^{(2)}$, $D^{(2)} \sim \text{Bernoulli}(p)$ where $(\odot)$ denotes element-wise multiplication. $(D^{(1)})$ and $(D^{(2)})$ are dropout masks. Then, the output layer consists of a softmax function to provide a probability distribution over the classes. For (c) classes, the output layer is $Z^{(3)} = W^{(3)}A^{(2)}_{\text{dropout}} + b^{(3)}$ and $\hat{Y} = \text{softmax}(Z^{(3)})$ where $(W^{(3)} \in R^{c\times64})$ is the weight matrix, $(b^{(3)} \in R^{c})$ is the bias vector, $(Z^{(3)})$ is the linear transformation output, $(\hat{Y})$ is the predicted probability distribution over the classes. Then, The softmax function is defined as $\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{c} e^{z_j}}$.

2.2.2. Convolutional Neural Network (CNN)

The CNN model is designed to capture spatial patterns in the CBC data, which can be represented in a structured format. This approach leverages the inherent spatial relationships between features that might not be immediately apparent in a tabular format. The architecture begins with an input layer that reshapes the CBC features into a 2D format, making them suitable for convolutional operations. The convolutional layers, which are the core building blocks of CNNs, consist of two layers with 64 and 32 filters, respectively. Each layer uses 3x3 kernels and ReLU activation to extract local patterns and features. Max pooling is applied after each convolutional layer to reduce the spatial dimensions and retain the most important features, thus reducing computational complexity and preventing overfitting. The flattened layer then transforms the 2D matrix into a 1D vector, which is fed into the fully connected layers. These layers include one hidden layer with 64 neurons and a ReLU activation function. To further enhance generalization and stabilize training, dropout and batch normalization are applied. The final output layer is a softmax layer for multi-class classification, predicting the type of anemia.

The input layer reshapes the CBC features into a 2D format suitable for convolutional operations. Suppose the input is $(X \in R^{m\times n\times c})$, where (m) is the number of samples, (n) is the number of features (reshaped into a 2D format), and (c) is the number of channels (in this case, $(c = 1)$). The first convolutional layer applies 64 filters with a 3x3 kernel is $Z^{(1)}_{i,j,k} = \sum_{p=0}^{2}\sum_{q=0}^{2} W^{(1)}_{p,q,k}X_{i+p,j+q} + b^{(1)}_{k}$ and $A^{(1)}_{i,j,k} = \text{ReLU}(Z^{(1)}_{i,j,k})$ where $(W^{(1)} \in R^{3\times3\times1\times64})$ is the weight tensor for the first convolutional layer, $(b^{(1)} \in R^{64})$ is the bias vector, $(Z^{(1)})$ is the linear transformation output, and $(A^{(1)})$ is the activation output after applying the ReLU function. Furthermore, the ReLU activation function is defined as $\text{ReLU}(z) = \max(0, z)$. The second convolutional layer applies 32 filters with a 3x3 kernel where $Z^{(2)}_{i,j,k} = \sum_{p=0}^{2}\sum_{q=0}^{2} W^{(2)}_{p,q,k}A^{(1)}_{i+p,j+q} + b^{(2)}_{k}$ and $A^{(2)}_{i,j,k} = \text{ReLU}(Z^{(2)}_{i,j,k})$ where $(W^{(2)} \in R^{3\times3\times64\times32})$ is the weight tensor for the second convolutional layer, $(b^{(2)} \in R^{32})$ is the bias vector, $(Z^{(2)})$ is the linear transformation output, and $(A^{(2)})$ is the activation output after applying the ReLU function.

Max pooling is applied to reduce the spatial dimensions. For a 2x2 max pooling operation can be described as following equation $P^{(2)}_{i,j,k} = \max\{A^{(2)}_{2i,2j,k}, A^{(2)}_{2i,2j+1,k}, A^{(2)}_{2i+1,2j,k}, A^{(2)}_{2i+1,2j+1,k}\}$. Then, the flattening layer transforms the 2D matrix into a 1D vector can be described as $F = \text{flatten}(P^{(2)})$ where $(F \in R^{d})$ is the flattened vector. The fully connected layer with 64 neurons and ReLU activation can be presented as $Z^{(3)} = W^{(3)}F + b^{(3)}$ and $A^{(3)} = \text{ReLU}(Z^{(3)})$ where $(W^{(3)} \in R^{64\times d})$ is the weight matrix, $(b^{(3)} \in R^{64})$ is the bias vector, $(Z^{(3)})$ is the linear transformation output, and $(A^{(3)})$ is the activation output after applying the ReLU function. Batch normalization is applied after the fully connected layer is $\widehat{Z^{(3)}} = \frac{Z^{(3)}-\mu^{(3)}}{\sqrt{\sigma^{(3)2}+\epsilon}}$ and $\widetilde{Z^{(3)}} = \gamma^{(3)}\widehat{Z^{(3)}} + \beta^{(3)}$ where $(\mu^{(3)})$ and $(\sigma^{(3)2})$ are the mean and variance of $(Z^{(3)}$ \), (ϵ) is a small constant for numerical stability, and $(\gamma^{(3)})$ and $(\beta^{(3)})$ are learnable parameters. Furthermore, Dropout is applied during training with a dropout rate $(p = 0.5)$, where $A^{(3)}_{\text{dropout}} = A^{(3)} \odot D^{(3)}$, $D^{(3)} \sim \text{Bernoulli}(p)$ where $(\odot)$ denotes element-wise multiplication and $(D^{(3)})$ is the dropout mask.

2.2.3. Recurrent Neural Network (RNN)

The RNN model, particularly using Long Short-Term Memory (LSTM) units, is tailored to capture temporal dependencies in sequential data. This is beneficial for CBC data, which can exhibit sequential dependencies across different features. The architecture starts with an input layer that reshapes

the CBC features into a 3D format suitable for LSTM input. The model includes two LSTM layers with 64 and 32 units, respectively, allowing it to learn long-term dependencies and temporal patterns in the data. After the LSTM layers, the data is passed through fully connected layers, which include one hidden layer with 64 neurons and ReLU activation. Dropout and batch normalization are applied to prevent overfitting and stabilize training. The final layer is a softmax output layer, which provides the probability distribution over the anemia classes, facilitating accurate multi-class classification.

The input layer reshapes the CBC features into a 3D format suitable for LSTM input. Suppose the input is \(X \in \mathbb{R}^{m \times t \times n} \), where \(m \) is the number of samples, \(t \) is the number of time steps, and \(n \) is the number of features. The first LSTM layer with 64 units can be represented as $h_t^{(1)}, c_t^{(1)} = \text{LSTM}\left(X, h_{t-1}^{(1)}, c_{t-1}^{(1)}; \theta^{(1)}\right)$, where $(h_t^{(1)})$ is the hidden state, $(c_t^{(1)})$ is the cell state, and $(\theta^{(1)})$ represents the learnable parameters of the first LSTM layer. The second LSTM layer with 32 units can be represented as $h_t^{(2)}, c_t^{(2)} = \text{LSTM}\left(h_t^{(1)}, h_{t-1}^{(2)}, c_{t-1}^{(2)}; \theta^{(2)}\right)$, where $(h_t^{(2)})$ is the hidden state, $(c_t^{(2)})$ is the cell state, and $(\theta^{(2)})$ represents the learnable parameters of the second LSTM layer.

2.2.4. Fully Connected Network (FCN)

The FCN model leverages convolutional layers to capture local patterns in the CBC data, combined with global averaging to reduce dimensions and enhance robustness against overfitting. The architecture begins with an input layer that reshapes the CBC features for convolutional operations. The convolutional layers include three layers with 128, 256, and 128 filters, respectively, each using ReLU activation to extract meaningful features from the data. Global average pooling is then applied to reduce the feature maps to a single vector by averaging, which helps in reducing the model's complexity and enhancing generalization. To further improve training stability and generalization, dropout and batch normalization are used. The final output layer is a softmax layer that performs the classification, providing the probability distribution over the different types of anemia.

The output from the second LSTM layer is passed through a fully connected layer with 64 neurons and ReLU activation is $Z^{(3)} = W^{(3)} h_t^{(2)} + b^{(3)}$ and $A^{(3)} = \text{ReLU}\left(Z^{(3)}\right)$ where $(W^{(3)} \in R^{64 \times 32})$ is the weight matrix, $(b^{(3)} \in R^{64})$ is the bias vector, $(Z^{(3)})$ is the linear transformation output, and $(A^{(3)})$ is the activation output after applying the ReLU function. Furthermore, batch normalization is applied after the fully connected layer as $\widehat{Z^{(3)}} = \frac{Z^{(3)} - \mu^{(3)}}{\sqrt{\sigma^{(3)2} + \epsilon}}$ and $\widetilde{Z^{(3)}} = \gamma^{(3)} \widehat{Z^{(3)}} + \beta^{(3)}$, where $(\mu^{(3)})$ and $(\sigma^{(3)2})$ are the mean and variance of $(Z^{(3)})$ over the batch, (ϵ) is a small constant for numerical stability, and $(\gamma^{(3)})$ and $(\beta^{(3)})$ are learnable parameters. Then, dropout is applied during training with a dropout rate ($p = 0.5$) with $A_{\text{dropout}}^{(3)} = A^{(3)} \odot D^{(3)}$, $D^{(3)} \sim \text{Bernoulli}(p)$ where $(\odot)$ denotes element-wise multiplication and $(D^{(3)})$ is the dropout mask. The output layer consists of a softmax function to provide a probability distribution over the classes. For (c) classes is following $Z^{(4)} = W^{(4)} A_{\text{dropout}}^{(3)} + b^{(4)}$ and $\hat{Y} = \text{softmax}\left(Z^{(4)}\right)$ where $(W^{(4)} \in R^{c \times 64})$ is the weight matrix, $(b^{(4)} \in R^{c})$ is the bias vector, $(Z^{(4)})$ is the linear transformation output, and $(\hat{Y})$ is the predicted probability distribution over the classes. The softmax function is defined as $\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{c} e^{z_j}}$.

2.3. Training Procedures

The training of each model followed a comprehensive and systematic approach to ensure robustness and accuracy. Initially, the dataset was split into training and testing sets using stratified sampling, which preserved the class distribution. Specifically, 80% of the data was allocated for training, while the remaining 20% was reserved for testing. This stratification ensured that each subset was representative of the overall class proportions, which is crucial for balanced model evaluation. To further enhance the robustness of the model evaluation and to mitigate the risk of overfitting, stratified K-Fold Cross-Validation with five folds was employed. In this process, the training data was further split into training and validation sets within each fold. This technique allowed the model to be trained and validated on different subsets of the data, ensuring that the evaluation was based on a diverse set of samples. By averaging the performance metrics across all folds, the cross-validation process provided a more reliable estimate of the model's generalization capability.

Hyperparameter tuning was a critical step in optimizing the model's performance. Key hyperparameters, such as learning rate, batch size, and the number of epochs, were systematically tuned using both grid search and random search methods. Grid search involved exhaustively searching through a predefined set of hyperparameters, while random search randomly sampled from the hyperparameter space. This dual approach ensured a comprehensive exploration of possible hyperparameter configurations. Additionally, early stopping was implemented to monitor the validation performance during training. If the performance did not improve after a certain number of epochs, training was halted to prevent overfitting. Learning rate schedules were also applied to adjust the learning rate dynamically,

enhancing the efficiency of the training process. The models were trained using the Adam optimizer, which is well-suited for handling large datasets and complex neural network architectures. The categorical cross-entropy loss function was employed, as it is appropriate for multi-class classification tasks. This loss function calculates the difference between the predicted probabilities and the true class labels, guiding the optimization process to minimize classification errors. The use of the Adam optimizer, combined with categorical cross-entropy, ensured effective and efficient convergence during training. Evaluation metrics were crucial in assessing the performance of the models. The primary metrics used included accuracy, precision, recall, F1-score, and confusion matrix. Accuracy measured the overall correctness of the model's predictions, while precision and recall provided insights into the model's ability to correctly identify positive instances and capture all relevant instances, respectively. The F1-score, which is the harmonic mean of precision and recall, offered a balanced measure of the model's performance, particularly in the presence of class imbalance. The confusion matrix provided a detailed breakdown of the model's performance across all classes, highlighting specific areas where the model excelled or struggled.

2.4. Evaluation and Comparative Analysis

The performance of the deep learning models was rigorously compared against traditional machine learning models. The traditional models included Logistic Regression, Decision Tree, Random Forest, and Support Vector Machine (SVM), all of which were implemented using Scikit-learn. These traditional models served as benchmarks for evaluating the effectiveness of the deep learning approaches. They were trained and evaluated on the same training and testing splits to ensure a fair comparison. The deep learning models were expected to outperform the traditional models due to their ability to automatically learn feature representations from the data. This advantage stems from the deep learning models' capacity to capture complex patterns and interactions within the data, which traditional models might miss. The performance of each model was evaluated on the test set using the aforementioned metrics. The results were then analyzed to identify the strengths and weaknesses of each model, providing a comprehensive understanding of their relative performance.

3. RESULTS

The primary objective of this study was to evaluate the performance of various deep learning models: Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Fully Connected Network (FCN)—in diagnosing anemia types from complete blood count (CBC) data. Each model's performance was assessed using stratified 5-fold cross-validation and further tested on a holdout dataset to provide a comprehensive evaluation. The performance metrics considered included accuracy, precision, recall, F1-score, and confusion matrices as presented in table 1-5.

The MLP model achieved a 5-fold cross-validation accuracy of 0.8105 with a standard deviation of 0.0279, indicating a reasonably high and stable performance. When evaluated on the test set, the MLP model demonstrated an overall accuracy of 0.79. The precision, recall, and F1-scores varied significantly across different classes, reflecting the model's varying ability to correctly identify and classify different types of anemia. For instance, the precision and recall for class 0 were 0.76 and 0.94, respectively, resulting in an F1-score of 0.84, indicating that the model was particularly effective in identifying this class. Conversely, the performance for classes 2, 3, and 4 was notably poorer, with F1-scores of 0.20, 0.00, and 0.00, respectively. This discrepancy highlights the challenge of dealing with imbalanced classes, as these classes had fewer instances in the dataset, which likely contributed to the model's difficulty in learning their patterns.

Tabel 1. Cross-Validation Accuracy

Model	5-fold CV Accuracy
MLP	0.8105 ± 0.0279
CNN	0.8770 ± 0.0358
RNN	0.7568 ± 0.0226
FCN	0.8594 ± 0.0129

Tabel 2. MLP Results

Class	Precision	Recall	F1-Score	Support
0	0.76	0.94	0.84	67
1	0.9	0.71	0.79	38
2	1.0	0.11	0.2	9
3	0.0	0.0	0.0	2
4	0.0	0.0	0.0	4
5	0.63	0.88	0.73	56
6	0.96	0.94	0.95	54
7	1.0	0.33	0.5	12
8	1.0	0.47	0.64	15

Tabel 3. CNN Results

Class	Precision	Recall	F1-Score	Support
0	0.9	0.94	0.92	67
1	0.84	0.82	0.83	38
2	1.0	0.33	0.5	9
3	0.5	1.0	0.67	2
4	1.0	0.5	0.67	4
5	0.72	0.91	0.8	56
6	0.96	0.93	0.94	54
7	1.0	0.5	0.67	12
8	0.92	0.73	0.81	15

Tabel 4. RNN Results

Class	Precision	Recall	F1-Score	Support
0	0.91	0.87	0.89	67
1	0.75	0.63	0.69	38
2	0.44	0.44	0.44	9
3	0.0	0.0	0.0	2
4	0.0	0.0	0.0	4
5	0.59	0.77	0.67	56
6	0.74	0.91	0.82	54
7	0.67	0.17	0.27	12

8	0.78	0.47	0.58	15

Tabel 5. FCN Results

Class	Precision	Recall	F1-Score	Support
0	0.87	0.91	0.89	67
1	0.8	0.87	0.84	38
2	0.5	0.33	0.4	9
3	0.67	1.0	0.8	2
4	0.8	1.0	0.89	4
5	0.88	0.8	0.84	56
6	0.96	0.89	0.92	54
7	0.75	0.75	0.75	12
8	0.74	0.93	0.82	15

The CNN model outperformed the MLP, achieving a 5-fold cross-validation accuracy of 0.8770 with a standard deviation of 0.0358. On the test set, the CNN model achieved an accuracy of 0.85. The model exhibited strong performance across most classes, with class 0 having a precision of 0.90 and a recall of 0.94, leading to an F1-score of 0.92. Similarly, classes 1, 5, 6, and 8 also showed high precision, recall, and F1-scores, reflecting the model's robustness in classifying these types of anemia. However, the CNN model struggled with classes 2 and 7, achieving F1-scores of 0.50 and 0.67, respectively. Although the performance for these classes was better than that of the MLP, it still indicates a need for improved handling of less represented classes. The confusion matrix for the CNN model also revealed that class 3, with a support of only 2 instances, was correctly classified in all cases, but the limited data might have skewed this result.

The RNN model, utilizing Long Short-Term Memory (LSTM) units, achieved a 5-fold cross-validation accuracy of 0.7568 with a standard deviation of 0.0226. On the test set, the RNN model's accuracy was 0.73, which was lower compared to both the MLP and CNN models. The class-wise performance of the RNN model showed considerable variation, with class 0 achieving a high precision of 0.91 and an F1-score of 0.89, while classes 3 and 4 had F1-scores of 0.00. The RNN's performance suggests that while it can capture sequential dependencies in data, it may not be as effective for this particular task with the current feature set and architecture. The RNN model's lower accuracy and variable class-wise performance indicate potential areas for improvement, such as incorporating more sequential information or refining the network architecture.

The FCN model showed strong performance, achieving a 5-fold cross-validation accuracy of 0.8594 with a standard deviation of 0.0129. On the test set, the FCN model also achieved an accuracy of 0.85, matching the performance of the CNN model. The class-wise results for the FCN were quite balanced, with class 0 achieving a precision of 0.87 and an F1-score of 0.89, and class 1 achieving an F1-score of 0.84. The FCN model demonstrated robustness in classifying anemia types with smaller sample sizes, as evidenced by the relatively high F1-scores for classes 3 and 4. This performance can be attributed to the global average pooling layer, which helps in reducing overfitting and capturing essential features more effectively. The overall performance of the FCN model suggests that it is well-suited for handling CBC data and provides a competitive alternative to CNNs.

4. DISCUSSION

The comparative analysis of the four models reveals distinct strengths and weaknesses. The CNN and FCN models demonstrated superior performance compared to the MLP and RNN models, both achieving an accuracy of 0.85 on the test set. The enhanced performance of CNNs and FCNs can be attributed to their ability to capture spatial patterns and complex feature interactions in the data, which are essential for accurate anemia diagnosis. The MLP model, while effective in certain classes, struggled with underrepresented classes, reflecting the limitations of fully connected layers in capturing intricate patterns without spatial information. The RNN model's lower performance indicates that sequential dependencies in CBC data might not be as critical as spatial relationships, or that the current RNN architecture needs further refinement. Class-wise performance analysis highlights the challenges posed by class imbalance, with less represented classes consistently showing lower F1-scores across all models. Addressing this issue could involve techniques such as data augmentation, oversampling of minority classes, or incorporating class weights in the loss function to penalize misclassifications of minority classes more heavily.

The findings of this study have significant implications for the development of automated diagnostic tools for anemia. The superior performance of CNN and FCN models suggests that these architectures should be prioritized in the development of diagnostic systems. Their ability to accurately classify different types of anemia from CBC data can enhance clinical decision-making, reduce the reliance on manual interpretation, and improve diagnostic accuracy. These automated tools can be particularly beneficial in low-resource settings, where access to skilled medical professionals may be limited. Class imbalance was a consistent challenge across all models, with minority classes often showing lower F1-scores. Addressing this issue is crucial for improving model performance and ensuring accurate diagnosis for all anemia types. Techniques such as data augmentation, oversampling of minority classes, or incorporating class weights in the loss function can help mitigate the impact of class imbalance. Future research should focus on developing and implementing these strategies to enhance model robustness.

5. CONCLUSION

This study evaluated the effectiveness of various deep learning models in diagnosing anemia types using complete blood count (CBC) data. By comparing the performance of Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Fully Connected Network (FCN) models, we identified the strengths and limitations of each approach. The CNN and FCN models emerged as the most effective, achieving high accuracy and robust performance across different anemia types. The CNN and FCN models demonstrated their capability to capture complex patterns and spatial relationships within the CBC data, significantly outperforming traditional machine learning models and other deep learning architectures. Their high accuracy and balanced class-wise performance highlight their potential for developing reliable automated diagnostic tools for anemia. These models can assist clinicians by providing accurate, efficient, and reproducible diagnoses, reducing the reliance on manual interpretation and potentially improving patient outcomes.

The study also underscored the importance of addressing class imbalance, which remains a critical challenge in medical diagnostics. Techniques such as data augmentation, oversampling, and weighted loss functions can mitigate the impact of class imbalance, ensuring that models perform well across all classes, including those with fewer instances. Future research should focus on integrating additional clinical data, such as patient history and demographics, to further enhance model accuracy and generalizability. Advanced architectures like transformers and techniques to improve model interpretability should also be explored. Implementing these models in real-time clinical settings will require careful consideration of their integration into existing workflows, ensuring that they provide actionable insights while being user-friendly for healthcare professionals.

ACKNOWLEDGEMENT

We express our deepest gratitude to Atma Jaya Catholic University of Indonesia for the unwavering support and resources provided throughout the duration of this research project. Our sincere thanks also extend to the Information System Study Program, whose rigorous academic environment and collaborative ethos have been instrumental in facilitating our work.

REFERENCES

[1] D. Kinyoki, A. E. Osgood-Zimmerman, N. V Bhattacharjee, N. J. Kassebaum, and S. I. Hay, "Anemia prevalence in women of reproductive age in low-and middle-income countries between 2000 and 2018," *Nat. Med.*, vol. 27, no. 10, pp. 1761–1782, 2021.

[2] S. A. Ali, U. S. Khan, and A. S. Feroz, "Prevalence and determinants of anemia among women of reproductive age in developing countries," 2020.

[3] S. Bathla and S. Arora, "Prevalence and approaches to manage iron deficiency anemia (IDA)," *Crit. Rev. Food Sci. Nutr.*, vol. 62, no. 32, pp. 8815–8828, 2022.

[4] M. D. Cappellini, K. M. Musallam, and A. T. Taher, "Iron deficiency anaemia revisited," *J. Intern. Med.*, vol. 287, no. 2, pp. 153–170, 2020.

[5] R. Lassila and J. W. Weisel, "Role of red blood cells in clinically relevant bleeding tendencies and complications," *J. Thromb. Haemost.*, 2023.

[6] T. Sonnweber, A. Pizzini, I. Tancevski, J. Löffler-Ragg, and G. Weiss, "Anaemia, iron homeostasis and pulmonary hypertension: a review," *Intern. Emerg. Med.*, vol. 15, pp. 573–585, 2020.

[7] D. T. Lee and M. L. Plesa, "Anemia," in *Family Medicine: Principles and Practice*, Springer, 2022, pp. 1815–1829.

[8] R. S. Hussien, S. I. A. Jabuk, Z. M. Altaee, and A. M. K. Al-Maamori, "REVIEW OF ANEMIA: TYPES AND CAUSES," 2023.

[9] M. Badireddy, K. M. Baradhi, and A. Wilhite Hughes, "Chronic Anemia (Nursing)," 2021.

[10] A. Al-Naseem, A. Sallam, S. Choudhury, and J. Thachil, "Iron deficiency without anaemia: a diagnosis that matters," *Clin. Med. (Northfield. Il).*, vol. 21, no. 2, p. 107, 2021.

[11] B. J. Bain, *Blood cells: a practical guide*. John Wiley \& Sons, 2021.

[12] L. Kamal and R. J. R. Raj, "Harnessing deep learning for blood quality assurance through complete blood cell count detection," *e-Prime-Advances Electr. Eng. Electron. Energy*, p. 100450, 2024.

[13] H. K. Bharadwaj *et al.*, "A review on the role of machine learning in enabling IoT based healthcare applications," *IEEE Access*, vol. 9, pp. 38859–38890, 2021.

[14] V. Singh, S.-S. Chen, M. Singhania, B. Nanavati, A. Gupta, and others, "How are reinforcement learning and deep learning algorithms used for big data based decision making in financial industries--A review and research agenda," *Int. J. Inf. Manag. Data Insights*, vol. 2, no. 2, p. 100094, 2022.

[15] S. Pullakhandam and S. McRoy, "Classification and Explanation of Iron Deficiency Anemia from Complete Blood Count Data Using Machine Learning," *BioMedInformatics*, vol. 4, no. 1, pp. 661–

672, 2024.

[16] A. Joshi, P. Saggar, R. Jain, M. Sharma, D. Gupta, and A. Khanna, "CatBoost—An ensemble machine learning model for prediction and classification of student academic performance," *Adv. Data Sci. Adapt. Anal.*, vol. 13, no. 03n04, p. 2141002, 2021.

[17] W. Jia, M. Sun, J. Lian, and S. Hou, "Feature dimensionality reduction: a review," *Complex \& Intell. Syst.*, vol. 8, no. 3, pp. 2663–2693, 2022.

[18] M. M. Nair, S. Kumari, A. K. Tyagi, and K. Sravanthi, "Deep learning for medical image recognition: open issues and a way to forward," in *Proceedings of the Second International Conference on Information Management and Machine Intelligence: ICIMMI 2020*, 2021, pp. 349–365.

[19] F. Piccialli, V. Di Somma, F. Giampaolo, S. Cuomo, and G. Fortino, "A survey on deep learning in medicine: Why, how and when?," *Inf. Fusion*, vol. 66, pp. 111–137, 2021.

[20] M. Li, Y. Jiang, Y. Zhang, and H. Zhu, "Medical image analysis using deep learning algorithms," *Front. Public Heal.*, vol. 11, p. 1273253, 2023.

[21] L. Gaur, M. Bhandari, T. Razdan, S. Mallik, and Z. Zhao, "Explanation-driven deep learning model for prediction of brain tumour status using MRI image data," *Front. Genet.*, vol. 13, p. 822666, 2022.

[22] J. Eom *et al.*, "Deep-learned spike representations and sorting via an ensemble of auto-encoders," *Neural Networks*, vol. 134, pp. 131–142, 2021.

[23] R. Gaikar, "Development of Machine Learning Algorithms for Kidney Cancer Diagnosis from Multi-Parametric MRI and Histopathology Images," University of Guelph, 2023.

[24] R. Levin, *Applications of Optimization and Machine Learning to Healthcare*. University of Washington, 2022.

[25] S. Cleveland, "The Future of Diagnosis: Achieving Excellence and Equity," *Planning*, 2023.

[26] P. Dutta, P. Upadhyay, M. De, and R. G. Khalkar, "Medical image analysis using deep convolutional neural networks: CNN architectures and transfer learning," in *2020 International Conference on Inventive Computation Technologies (ICICT)*, 2020, pp. 175–180.

[27] W. Li, M. Zuo, H. Zhao, Q. Xu, and D. Chen, "Prediction of coronary heart disease based on combined reinforcement multitask progressive time-series networks," *Methods*, vol. 198, pp. 96–106, 2022.

[28] H. Koch *et al.*, "CLIMB: High-dimensional association detection in large scale genomic data," *Nat. Commun.*, vol. 13, no. 1, p. 6874, 2022.

[29] M. S. Pingel, "Leveraging machine learning and process mining to predict anaemia with the help of biomarker data," University of Twente, 2021.

[30] O. Saidani *et al.*, "White blood cells classification using multi-fold pre-processing and optimized CNN model," *Sci. Rep.*, vol. 14, no. 1, p. 3570, 2024.

[31] M. Kirschenbaum, "Spec Acts: Reading Form in Recurrent Neural Networks," *ELH*, vol. 88, no. 2, pp. 361–386, 2021.

[32] F. H. Awad, M. M. Hamad, and L. Alzubaidi, "Robust classification and detection of big medical data using advanced parallel k-means clustering, yolov4, and logistic regression," *Life*, vol. 13, no. 3, p. 691, 2023.

[33] J. W. Asare, P. Appiahene, E. T. Donkoh, and G. Dimauro, "Iron deficiency anemia detection using machine learning models: A comparative study of fingernails, palm and conjunctiva of the eye images," *Eng. Reports*, vol. 5, no. 11, p. e12667, 2023.

[34] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electron. Mark.*, vol. 31, no. 3, pp. 685–695, 2021.

[35] J. Krois *et al.*, "Generalizability of deep learning models for dental image analysis," *Sci. Rep.*, vol. 11, no. 1, p. 6102, 2021.