Feature-enhanced residual signals with parallel hybrid network and novel loss for noninvasive blood glucose detection

Mochammad Fatchur Rahman [a,1], Karlisa Priandana [a,b,2,*], Husin Alatas [b,c,3], Renan Prasta Jenie [d,4]

[a] Computer Science Study Program, School of Data Science, Mathematics, and Informatics, IPB University, Bogor 16680, Indonesia
[b] Center for Transdisciplinary and Sustainability Sciences (CTSS), IPB University, Bogor 16127, Indonesia
[c] Department of Physics, Faculty of Mathematics and Natural Sciences, IPB University, Bogor 16680, Indonesia
[d] Department of Digital Business, Faculty of Business and Social Science, Binawan University, Jakarta 13630, Indonesia
[1] mfatchur_rahman@apps.ipb.ac.id; [2] karlisa@apps.ipb.ac.id; [3] alatas@apps.ipb.ac.id; [4] renan.prasta@binawan.ac.id
* corresponding author

ARTICLE INFO

Article history
Received February 22, 2025
Revised April 11, 2026
Accepted April 11, 2026
Available online August 31, 2026

Keywords
Noninvasive blood glucose
Residual infrared signals
Feature engineering
Hybrid deep learning
Distribution-aware loss

ABSTRACT

Diabetes remains a major global health challenge. According to the International Diabetes Federation (IDF), approximately 10.5% of the world's population is affected, with nearly half unaware of their condition. Therefore, accurate and accessible blood glucose level (BGL) monitoring remains critical. This paper presents a noninvasive BGL estimation approach using in vivo residual infrared (IR) signals collected from patients' fingertips, which is inherently challenging due to weak glucose-related signatures and interference from other physiological factors. To address this, we propose a tailored feature engineering strategy to uncover hidden relationships between residual signals and BGL. The proposed method improves BGL range discrimination, as indicated by increased Spearman correlations and consistent gains in Kendall and Pearson correlations, demonstrating its effectiveness. Furthermore, we propose a task-specific hybrid neural network architecture, consisting of parallel 1D convolutional and dense blocks to capture both local spectral–temporal patterns and global nonlinear relationships. A distribution-aware loss function is also introduced to reduce prediction bias toward the dataset mean, improving sensitivity across the full BGL range. For validation, the proposed model is compared with random forest (RF) and boosting-based methods, showing superior performance with a 14.11% mean absolute percentage error (MAPE). Clinical reliability is evaluated using Clarke error grid (CEG) analysis, where 84.7% of predictions fall within the accurate zone and 15.3% within the clinically acceptable zone. These results demonstrate the potential of the proposed approach for practical, low-cost, and portable noninvasive BGL monitoring.

1. Introduction

Diabetes emerges as one of the most dangerous health threats in the world. The International Diabetes Federation (IDF), as published in [1], reported that 10.5% of the world's adult population has diabetes, with almost half unaware that they are living with the condition. Moreover, [1] also showed a projection that approximately 783 million people will be living with diabetes by 2045, representing a 45% increase. Despite this reality, a complete cure or treatment is still not available for the disease [2]. Furthermore, the disease stages are classified into prediabetes and diabetes [2]. During the prediabetes stage, patients are usually unaware that they have the disease. This reality occurs due to a lack of

knowledge about the disease and unaffordable testing kits for most people, especially in developing countries. Consequently, low-cost measurement device inventions to predict BGLs are still relevant. Hence, developing models for non-invasive BGL measurement remains a demanding research topic to explore

Many researchers have developed BGL models, generally grouped into regression, forecasting, and classification. For regression tasks, [3] utilized photoacoustic spectroscopy (PAS) to predict BGLs. A similar sensing methodology was explored by [4], which employed convolutional neural network models and demonstrated promising results. High implementation costs remain the primary barrier to the widespread adoption of PAS methodology, especially for developing countries. In addition, the approach is not sufficiently efficient or portable because significant installation challenges prevent independent home operation by users.

Besides PAS signals, infrared-based sensing methods offer more efficient implementations because they are more practical and cost-effective. However, it is more challenging in terms of signal reliability [5], which observed near-infrared (IR) signals to predict BGL. Instead of actual blood samples, they employed synthetic laboratory samples that simulated blood using aqueous solutions of glucose, ascorbate, urea, and alanine at various concentrations, which do not adequately represent the full morphological and biochemical complexity of real human blood. Similarly, [6]–[8] developed synthetic blood solutions for the same task. In addition, experiments with similar features to [5] were conducted by [9]–[12] and concluded that neural networks performed well. Regarding IR sensing methodologies, some studies like [7], [13], [14] observed the IR residual signals, [15] deployed IR reflectance features, and [16] employed natural IR signals from the human body as features. Despite its practical advantages, IR-based sensing remains highly challenging for accurate BGL estimation. This is mainly due to the weak glucose-specific absorption in IR spectra, low signal-to-noise ratio, and strong interference from other biological components such as water, fat, and proteins. In addition, inter-subject variability and environmental factors further distort the residual signals, making it difficult to extract robust glucose-related features. These limitations explain why IR-based noninvasive BGL estimation is still an open problem despite extensive research.

Alongside PAS and IR signals, multisensor systems [17], microwaves [18]–[21], and EEG signals [22] were explored as alternative methodologies. Similar to natural human body IR, multisensor systems face inefficiencies in practical applications due to complexity and limitations. At the same time, EEG technologies pose significant barriers to widespread use because they require expensive equipment. While these modalities offer improved signal reliability, they often suffer from high cost, complexity, or poor portability, which limits real-world adoption. In contrast, IR-based sensing is more practical but faces fundamental signal quality challenges, creating a trade-off between usability and accuracy.

In terms of model comparison, [23] reported that convolutional neural network (CNN) performed better than boosting models, linear, ridge, and support vector regression. For BGL forecasting tasks, [24] implemented bi-directional long short-term memory (LSTM), and reported that Bi-LSTM performed better than the lightweight transformer. A similar model (LSTM) was applied by [25]–[28] and produced notable outcomes. The combination of LSTM and transformer also achieved notable performances as reported by [29]. Additionally, [30], [31] conducted a BGL forecasting study, where LSTM outperformed ARIMA. Moreover, for diabetes classification tasks, [32] found that extreme gradient boosting (XGBoost) outperformed convolutional neural networks. In contrast to [32], neural network models exhibited better performance than traditional models, as reported by [32], a trend that is rapidly emerging in BGL modeling. Neural network superiority predominantly happens when the observed features show spatial correlations.

While most reports confirm the superior performance of the neural network models, as stated in the previous sections, we will now examine non-neural network models for BGL predictions. For example, [33] found that light gradient boosting machines (LightGBM) demonstrated better performance compared to random forest (RF), decision tree (DT), XGBoost, and LSTM. Furthermore, [32] reported that XGBoost demonstrated better performance than RF. The boosting algorithm's superiority was

confirmed by [34], who revealed that XGBoost outperformed k-nearest neighbor (KNN), SVMs, and RF. Based on these findings, we conclude that LSTM and RNN show more promise in forecasting scenarios, while boosting models sometimes appear to have an edge in classification tasks. Furthermore, feature characteristics contribute to model performance. Neural network models mostly excel in BGL prediction for input features with spatial characteristics, such as signal sequences or images. Moreover, the superiority of LSTM for forecasting tasks stems from its ability to analyze pattern sequences, while XGBoost's dominance in classification areas confirms its reputation as a Kaggle-winning algorithm for tabular data competitions. In 2015 alone, it was reported that 17 out of 29 Kaggle winning models were XGBoost [35].

Building upon prior work, this research presents several interconnected research contributions for noninvasive BGL estimation from in vivo infrared residual signal. First, we propose a unified feature engineering pipeline, combining Fourier-based noise filtering with a relative gradient percentage transformation. This is a preprocessing strategy specifically designed to enhance the BGL range of discriminability in sparse and nonlinear data. Unlike standard normalization approaches, the proposed transformation preserves sequential signal characteristics while simultaneously achieving zero-centering and scale invariance, as demonstrated empirically through correlation analysis and ablation comparison against the well-known Standard Scaler. Second, we introduce a parallel hybrid network that combines the convolution block and the dense block. Convolutional blocks capture localized spectral-temporal patterns across consecutive residual readings, while dense blocks model global nonlinear feature-target relationships. Third, our proposed architecture is specifically designed to enable simultaneous BGL regression and range classification within a single forward pass. Moreover, we introduced a distribution-aware composite loss function combining binary cross-entropy (BCE) and mean squared error (MSE), which explicitly prevents prediction collapse toward the dataset mean. The BCE component enforces range sensitivity, while the MSE component maintains numerical precision. The proposed loss was empirically compared to the MSE loss function only. Finally, most researchers developed their models based on synthetic blood solutions, we utilized actual BGLs and IR residual features obtained in vivo from patients' fingers as in [9], supporting the feasibility of future implementation in a portable device.

2. Method

2.1. Research Workflow

The research started with dataset preparation, leveraging the dataset previously utilized in [9] to extend and build upon their previous work. Furthermore, we performed dataset descriptive explorations such as analyzing the missing values (NaN), outliers, feature correlations, and target distributions. Subsequently, we handled the NaNs and outliers, followed by data splitting. The flow of our research is presented in Fig. 1.

We treated missing values and outliers by removing the affected data points. This approach was adopted because our modeling was regression-based, where the same target values were often too sparse to estimate the actual mean or median of the data. For example, the BGL of 800 mg/dl only consists of 5 data points; if we were to replace the missing values with the mean, it would probably be inaccurate. Moreover, replacing them with these statistical metrics would introduce additional noise into the data. In terms of outlier removal, we inspected the data for each 10 mg/dl BGL range. This step is substantially critical in terms of ensuring that no noisy features adversely affect the model training. Additionally, we employed equation 1 from [36] to determine whether a feature belongs to an outlier category:

$$x < Q_1 - 1.5 \times IQR \; or \; x > Q_3 + 1.5 \times IQR \tag{1}$$

where $Q_1$is the first quartile (25^{th} percentile), Q_3 is the third quartile (75^{th} percentile), IQR is the interquartile range.

Moreover, we split the dataset into training, validation, and testing with proportions of 80%, 16%, and 4%. To avoid information leakage, dataset splitting was performed before feature engineering.

Information leakage occurs when information from the test data leaks into the training data, causing model performance to be overestimated [37]. The training and validation datasets were employed during the training process, while the test dataset was evaluated for final performance only. Subsequently, we employed feature engineering followed by model training, which is explained in more detail in the following section.

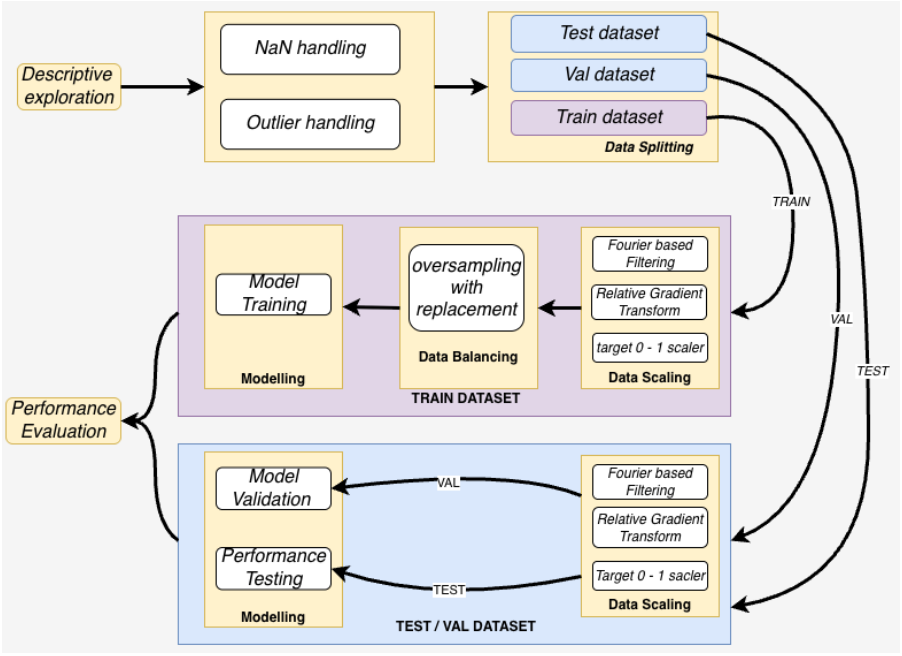


Fig. 1. Research Workflow

2.2. Data Acquisition

The data used in this research consists of residual features from infrared signals transmitted through patients' fingers, as shown in Fig. 2. The device was developed in a previous study [9], featuring two LEDs with several condition combinations: all LEDs off, an LED with 1550 nm of IR on, and an LED with 1600 nm of IR on. The use of the two wavelengths was justified by two previous studies: IR 1550 nm utilization was studied by [38], and the IR 1600- 1700 nm range was explored by [39]. When the LED was on, the sensor captured residual intensity features. Furthermore, each LED was turned on and off multiple times, producing sequential waves of residual intensities with peaks and valleys. Subsequently, the sensor captured the residual intensity, and the computer recorded the data. As shown in Fig. 2, the patient's right index finger is placed inside the device.

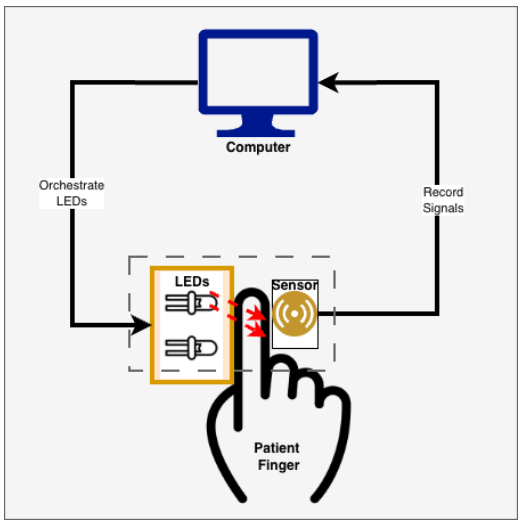


Fig. 2. Measurement Device Schematic Design

Then, the device collects the signal features 5 times for each patient. The data obtained, however, does not uniformly consist of five measurements per patient due to technical limitations. An example of the collected signal features are shown in Fig. 3.

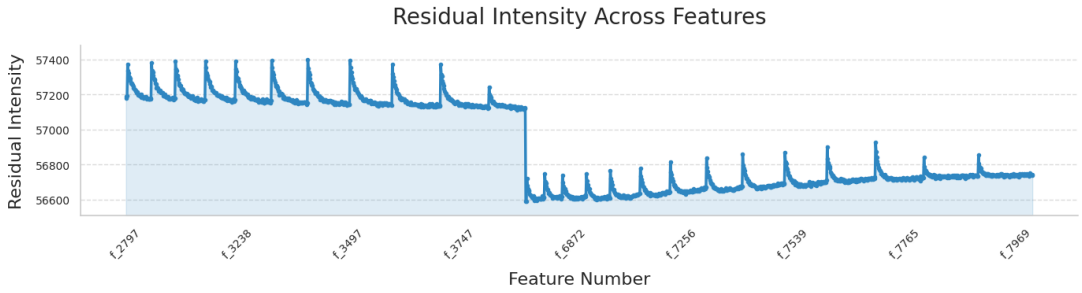


Fig. 3. Original Blood Glucose Residual Intensities

From 1400 patients, a total of 6677 data were collected. This number is significantly larger compared to [5], [11], [15]. We have obtained the required ethical clearance to conduct the experiments using the patient's data. Moreover, each patient whose data is used for this research had filled in an informed consent form to support ethical data collection procedures. The gold standard for our modelling was the actual values of the patients' BGL, measured simultaneously using a gold-standard device. The dataset collection was conducted at a hospital in Malang, East Java, Indonesia. Each patient had their BGL measured in the hospital's laboratory. However, the target data are heavily imbalanced, as shown in Fig. 4.

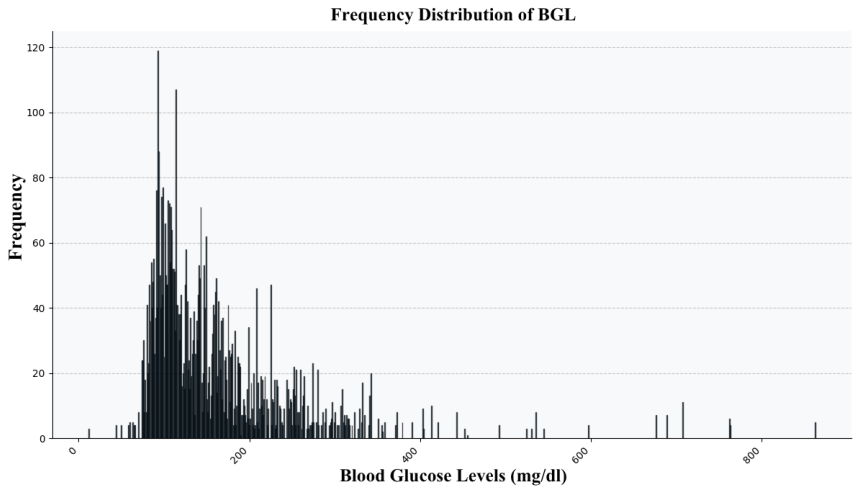


Fig. 4. Frequency Distribution of BGL

The detailed frequency breakdown for each BGL range is shown in Table 1. Nearly 80% of our data samples are concentrated among 0 –100 mg/dl and 100 – 200 mg/dl classes. This imbalance will push the model to favor predictions in that range. Meanwhile, we aim to make the model sensitive toward higher BGL ranges, where critical patient conditions are more likely to occur. To solve this issue, we proposed a data imbalance handling strategy that will be discussed in more detail in the next section.

Table 1. Frequency Distribution of BGL

Parameter	BGL Range (mg/dl)			
	0 - 100	*100 - 200*	*200 - 300*	*>300*
Frequency	1510	3780	1007	380
Frequency Percentage	22.61%	56.61%	15.08%	5.68%

Moreover, we split the dataset into training, validation, and testing sets with proportions of 80%, 16%, and 4%. The training and validation datasets were employed for the training process, while the test dataset was utilized for testing only to avoid information leaks. We also marked each data point with a patient ID, preventing data from the same person from appearing in both the training/validation set and the test set. Furthermore, the test data was only 4% (267 data points out of 6677 population). Therefore, the test set size complies with Slovin's sampling equation (a widely adopted statistical sampling equation). At this sample size, it achieves a 95% confidence level with a 6% error margin. In addition, this sample size exceeds the Central Limit Theorem threshold (n > 30), ensuring that the distribution of prediction errors approximates normality and that performance metrics such as MAPE and CEG analysis remain statistically reliable.

2.3. Imbalance Data Handling

We employed balanced sampling with replacement for each class to overcome the imbalance problem, instead of using the well-known SMOTE method. We made this choice because SMOTE relies on linear interpolation between features and was originally designed to address classification imbalance. In contrast, the study revealed that the inspected BGL features are non-linearly correlated, and the case is a sparse regression problem. More detailed analysis and empirical evidence for this decision will be presented in the discussion section. The selected imbalance handling method is illustrated in Fig. 5. We ensure that the training data is balanced across each BGL class range. Consequently, data points within the minority BGL ranges (200-300 mg/dl and > 300 mg/dl) will be exposed more frequently during training, driving the model to give more attention to minority areas.

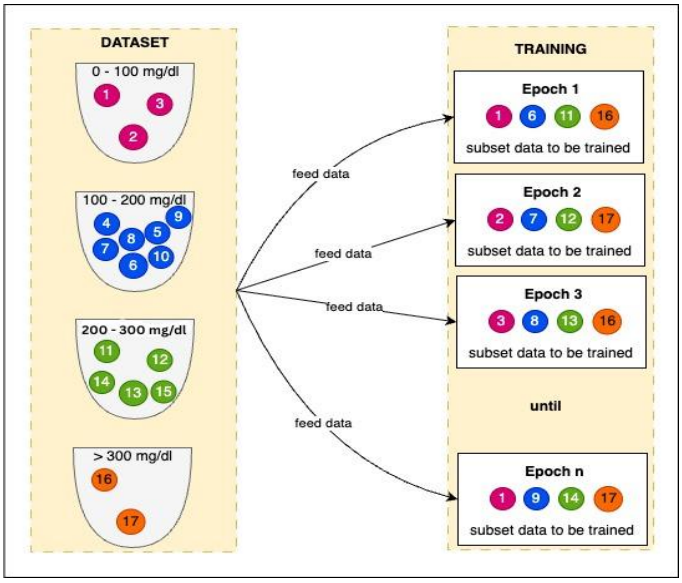


Fig. 5. Selected Imbalance Handling Method

2.4. Proposed Feature Engineering

This section presents feature engineering methods performed in our study. First, we performed a Fourier-based filtering approach. This involved converting the feature sequence to the frequency domain using the Fourier transform, removing high-frequency components through thresholding, and reconstructing the filtered signal using the inverse Fourier transform. These steps remove high-frequency noise while preserving the main characteristics of the residual features, as shown in Fig. 6. Then we applied data scaling approaches to avoid vanishing loss during training, as explained in [40]. Especially for target data, we performed zero-one scaling. The method was adopted because some machine learning models perform optimally in this range. Moreover, some neural network activation functions, such as Softmax and Sigmoid, also have output values in this range. Furthermore, for the residual features, we performed data scaling by transforming them to relative gradient percentages based on equation 2:

$$\nabla f(n) = \frac{f(n) - f(n-1)}{f(n-1)} 100\% \quad (2)$$

where $f(n)$ and f(n-1)$f(n-1)$ features of n and n-1. The benefit is that we obtain new features invariant to intensity scales. In our dataset case, the relative gradient percentage values remain in the zero-one range. Subsequently, we performed oversampling with replacement to tackle data imbalance for the training dataset only. This approach means that during training, the model pays more attention to minority samples.

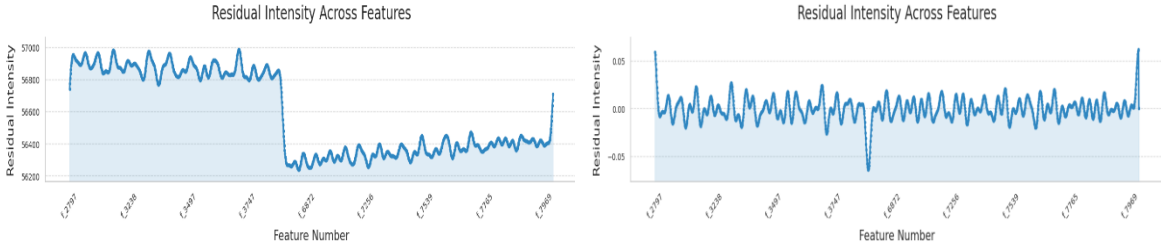


Fig. 6. Frequency-based noise filtering result (left); relative gradient percentage transformations result (right)

2.5. Proposed Neural Network Architecture

We proposed a new neural network architecture designed specifically for the study. Our architecture is a hybrid network, composed of a parallel combination of dense blocks and 1D convolutional blocks. The intuition behind the design is that dense layer operations tend to observe the residual features globally based on their values, whereas 1D convolution operations analyze these features as a sequence of interdependent values. In other words, the dense block excels at modeling based on feature intensity, such as maximum and minimum values, while convolution is strong in analyzing sequential feature patterns like cycles and forms. Furthermore, 1D convolution also excels at automatic feature engineering, enabling it to uncover hidden insights from these patterns. In terms of traditional machine learning, 1D convolutions act as filters like Gabor or Gaussian filters, but their values are tuned automatically during training. Each 1D convolutional operation consists of multiple filters, with the number determined by the filter parameter setting. The given input features are then distributed to both 1D convolutional and dense blocks. The convolutional block consists of three consecutive layers. Each layer uses 1x3 kernels and can be configured with 16, 32, or 64 filters, capturing local patterns across three consecutive readings. We adopted a 1x3 kernel size to keep computational complexity manageable. In the dense block, two dense layers are implemented with their hidden neuron number configurable to 16, 32, and 64. We chose 16, 32, and 64 as filter counts to keep the model's trainable parameters small while still being capable of learning patterns. It also satisfies the 2^n rule of thumb for optimal Nvidia GPU performance. The temporary output from the second convolutional layer was copied and then flattened into a dense layer with 64 output neurons. The outputs from the flattened layer and dense block were concatenated together. Each has 64 output neurons, maintaining balanced contributions between the convolutional and dense blocks.

The proposed architecture is illustrated in Fig. 7. Each fully connected operation in a dense or convolutional block is followed by leaky-ReLU activation, batch normalization [41], and then dropout. Leaky-ReLU is a new improvement over the ReLU activation function to avoid dead neurons during training. At the same time, we combined batch normalization and dropout to buffer neuron value spikes in hidden layers and serve as regularization to prevent overfitting.

The parameter configurations of our proposed network architecture are presented in Table 2. The dropout rate was configurable to 10%, 15%, 20%, and 25%. Consequently, some neurons in each layer were selected randomly and set to zero or dead during training, a memory loss technique to generalize patterns. The approach serves as model regularization to prevent overfitting. Dropout also creates an ensemble effect for neural networks, where different neurons learn different patterns in each iteration, creating a subnetwork combination effect during inference. The 0.0001 initial learning rate was applied during training with 10 iterations of patience and 5% decay. The configuration means that during ten

consecutive epochs, if the best model validation loss doesn't find a new better loss, then the learning rate will be reduced by 5%.

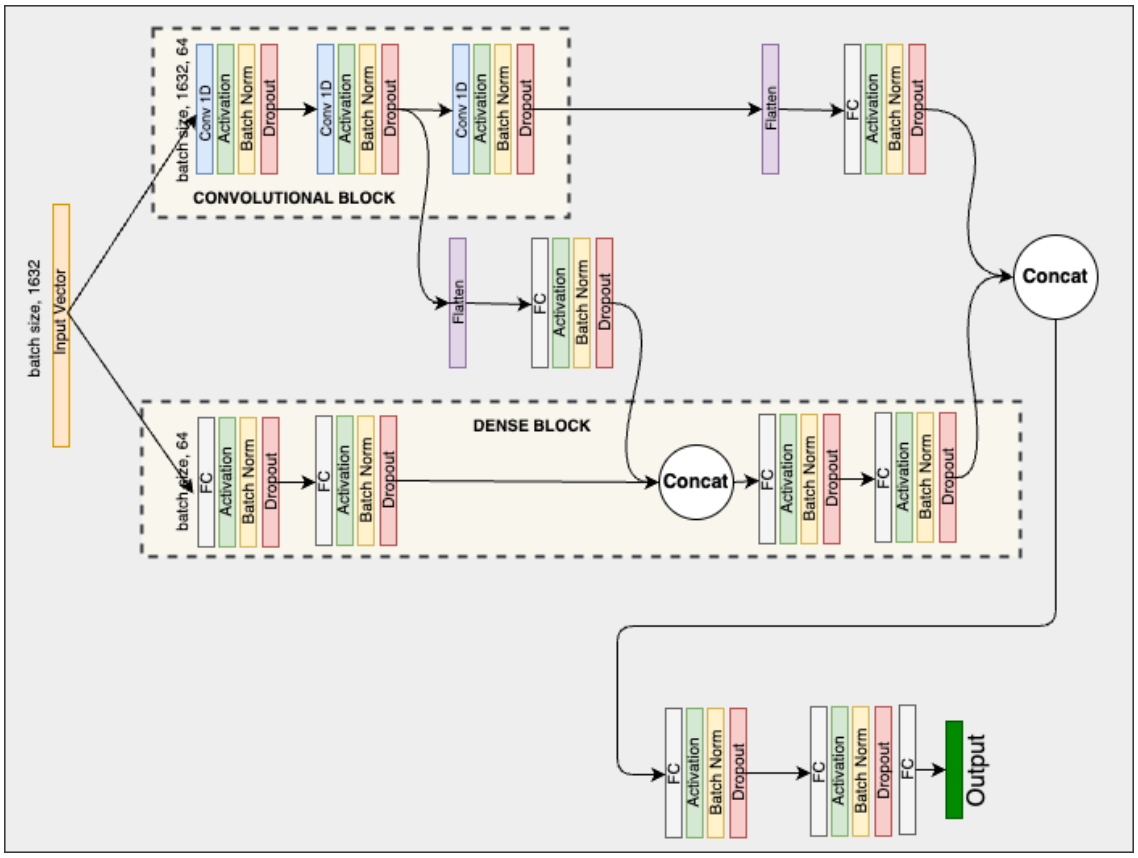


Fig. 7. Proposed Neural Network Architecture

Table 2. Neural Network Configurations

Parameter	Value
Optimizer	Adam optimizer
Regularization	Dropout
Dropout rates	0.10, 0.15, 0.20, 0.25
Variable initialization	Kaiming initialization
Conv 1D kernel size	1 x 3
Number of dense layer neurons	16, 32, 64
Num of 1D convolution kernel	16, 32, 64
Learning rates	0.0001
Learning rates decay	5%
Learning rate patience	10
Hidden layer activation	Leaky-ReLu
Output activation	Sigmoid

2.6. Proposed Output Layer

The network output network was designed distinctly as presented in Fig 8, compared to widely used regression neural networks. Instead of using a single neuron for prediction, we designed the output layer to consist of multiple neurons corresponding to predetermined BGL ranges. The study defined four BGL ranges, 0 - 100, 100 - 200, 200 - 300, and 300 - 800 mg/dl. Subsequently, we applied a Sigmoid activation function to the layer, and the result was obtained using Equation 3.

$$Output = BGL_k^{min} + p_k \cdot (BGL_k^{max} - BGL_k^{min}) \quad (3)$$

where $k = \arg\max_i(p_i)$ is the index of the neuron with the highest activation, BGL_k^{min} and BGL_k^{max} are the minimum and maximum BGL boundaries of range k, and p_k is the output of neuron k after Sigmoid activation. When the output vector is [0.0, 0.0, 0.3, and 0.0], the largest value is in the third neuron, which is 0.3, corresponding to the BGL range of 200mg/dl - 300 mg/dl. The minimum and maximum boundaries for this range are 200 mg/dl and 300 mg/dl. Applying Equation 3, the final prediction is computed as: 200 mg/dl + 0.3 (300 mg/dl - 200 mg/dl) = 230 mg/dl. The loss was inspired by object detection loss, where BGL ranges represent object grids, and Sigmoid probability represents the object center point within the range. The benefit of the output design is that the proposed network can perform regression and classification tasks simultaneously.

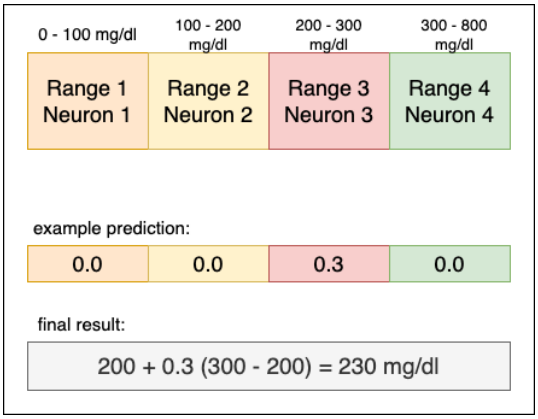


Fig. 8. Proposed Network Output

2.7. Proposed Loss

The most widely used loss function for BGL regression tasks is a mean squared error (MSE). The challenge of using this loss is that model predictions tend to concentrate near the mean or median of targets, especially for complex modeling tasks like BGL regression. To overcome this behavior, we proposed a new paradigm for BGL regression loss by adding binary cross-entropy (BCE) as a classification loss alongside MSE loss. Hence, the model can be trained for regression and classification simultaneously. The new equation is written as follows:

$$loss = \alpha\, BCE + \beta\, MSE \tag{4}$$

where α and β are hyperparameters controlling the relative contribution of each loss component. The approach was inspired by object detection by [42], where their loss is a combination of classification and regression loss. BCE loss will prevent prediction results from concentrating near the mean and push them toward their actual ranges, while MSE loss will maintain the Euclidean distance between actual and predicted values as close as possible. Additionally, based on the [42] object detection algorithm, their objectness loss represents which grid the object lies in. This is similar to our BCE loss, which represents the range in which the BGL value lies. Moreover, their center loss represents the actual value of the object's center, similar to our MSE loss, which represents the actual BGL values.

At the beginning of training, we guided the model to excel in range classification by setting α values far greater than β When the model achieved good range-classification performance, indicated by high and robust range accuracy on validation data (accuracy > 85%), we shifted the model to focus on the regression task by reducing α and increasing β.

2.8. Performance Metrics

Model performance was evaluated using Clarke Error Grid Analysis (CEG) [43]. The method is categorized as a gold standard approach to validate the measurement accuracy of BGL. Furthermore, the plot has five zones: A, B, C, D, and E. Predictions resulting in zone A indicate a clinically accurate prediction; zone B reveals clinically acceptable results with minor errors (while not perfect, these predictions are still useful for general treatment guidance); and zone C demonstrates inappropriate

predictions. The two other zones are clinically unacceptable and potentially cause mistreatment for patients. Zone A covers an area of around 20% error from actual results. Besides CEG, we adopted mean absolute percentage error (MAPE) as a model performance metric. The metric was selected due to MAPE's interpretation being invariant to BGL ranges, which is more reliable for evaluating predictions across all ranges. In contrast, MSE or RMSE metrics sometimes have different interpretability between low and high BGL ranges.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - y_i'}{y_i}\right| 100\,\% \quad (5)$$

2.9. Comparison Models

To validate our proposed network performance, model comparisons were performed between the model and other machine learning algorithms that are widely used for BGL regression tasks, such as random forest (RF) and boosting models. Random forest (RF) is an ensemble model of decision trees (DT), a machine-learning model inspired by hierarchical decision structures. RF was selected for model comparison due to its ensemble nature, which is relatively resilient to data imbalance and complex feature-target relationships. The configuration of the boosting models is shown in Table 3.

Table 3. RF Configurations

Parameter	Value
maximum depth	5, 7, and 10
Max leaf nodes	20, 30, and 40
Min sample split	2, dan 5
Random state	42
Number of estimators	100

Gradient boosting decision tree (GBDT) is an improved version of decision tree ensemble models, consisting of sequential weak classifiers. Each weak classifier learns to minimize errors from previous classifiers using gradient descent. The final prediction is obtained by combining all tree outputs, weighted by their learning rates. For implementations, our study employed XGBoost [35] and a light gradient boosting machine (LightGBM) as comparison models. The decision to use XGBoost and LightGBM as comparison models was based on their boosting nature, which tends to be resilient to imbalanced data. It also has a strong reputation as a Kaggle-winning algorithm for tabular data competitions [35]. The parameter configuration of our boosting implementations is presented in Table 4.

Table 4. XGBoost and LightGBM Configurations

Parameter	Value
maximum depth	5, 7, and 10
Max leaf nodes	20, 30, and 40
Min sample split	2, dan 5
Random state	42
Number of estimators	100
Num leaves	20, 30, and 40
Learning rates	0.05 and 0.1

3. Results and Discussion

3.1. Feature Engineering Insights

The proposed feature engineering successfully centered our feature sequence around zero while keeping the data range between -1 and 1, as shown in Fig. 9.

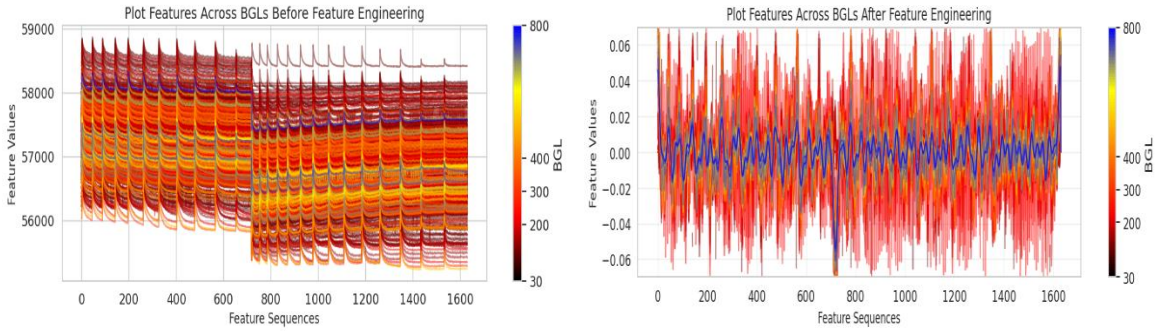


Fig. 9. Plot Features Across Blood Glucose Levels, Before and After Feature Engineering

Thus, the transformation effectively combines data scaling, zero mean, and normalization into a single operation. The proposed feature engineering demonstrates promising results, indicated by the improvement in residual feature-target correlations, measured by Spearman, Pearson, and Kendall, as shown in Fig. 10.

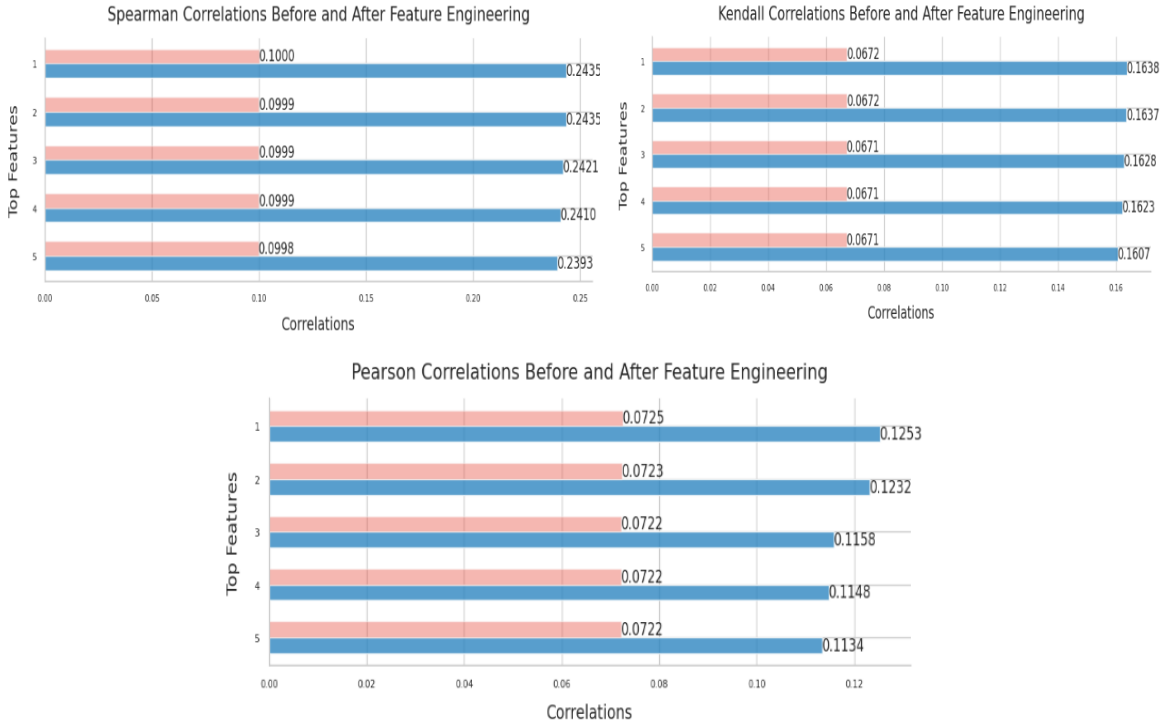


Fig. 10. Correlations between Features and BGLs, Before and After Feature Engineering

Fig. 9 visualizes feature sequences across BGLs, before and after feature engineering. The figure shows that before feature engineering, discrimination between low and high BGLs was challenging, indicating no clear pattern between residual features and BGLs. After feature engineering, the figure demonstrates that residual features across BGLs become more separable. Additionally, higher BGLs tend to be more compact in gradient-relative percentage and more separable from lower BGLs. These findings suggest that feature engineering produced features with better discrimination across BGLs, confirming the results of the correlation analysis. Furthermore, the figure also reveals that high BGLs yield clear signal patterns, similar to BGL signals reported by [14], [44]. Lastly, the figure also shows that residual features become zero-centric after feature engineering, successfully achieving the initial goals.

Fig. 10 illustrates that Spearman correlations improve significantly after feature engineering, from around 0.1 to 0.243, an approximately 2.5-fold improvement. The improvement suggests that the methodology uncovers monotonic and hidden nonlinear relationships between residual features and

BGLs. Meanwhile, Kendall and Pearson correlations improve moderately. These improvements across all metrics indicate that the proposed feature engineering has robust positive impacts. Furthermore, the relatively higher Spearman and Kendall correlations than Pearson correlations suggest that features and targets have nonlinear relationships. Consequently, instead of using the synthetic minority oversampling technique (SMOTE), we prefer using oversampling with replacement to handle data imbalance during training.

3.2. Proposed Data Scaling Performances

As discussed in the previous section, we proposed a feature engineering approach that includes Fourier-based noise filtering and data scaling. Particularly for data scaling, it is a conventionally recommended procedure for a neural network architecture, as mentioned by [39]. To evaluate the proposed data scaling performance, we conducted an ablation study by comparing it with the well-known Standard Scaler algorithm, as shown in Fig. 11. The other parameters for this comparison are set to default, meaning both use Fourier-based noise filtering, the selected imbalance handling, the proposed architecture, and the proposed loss function. Fig. 11 illustrates that the proposed scaler method outperforms the Standard Scaler algorithm in both accuracy and MSE loss. Furthermore, the proposed scaler achieves 95% BGL range accuracy, indicating that it can recognize the minority area of higher BGL levels.

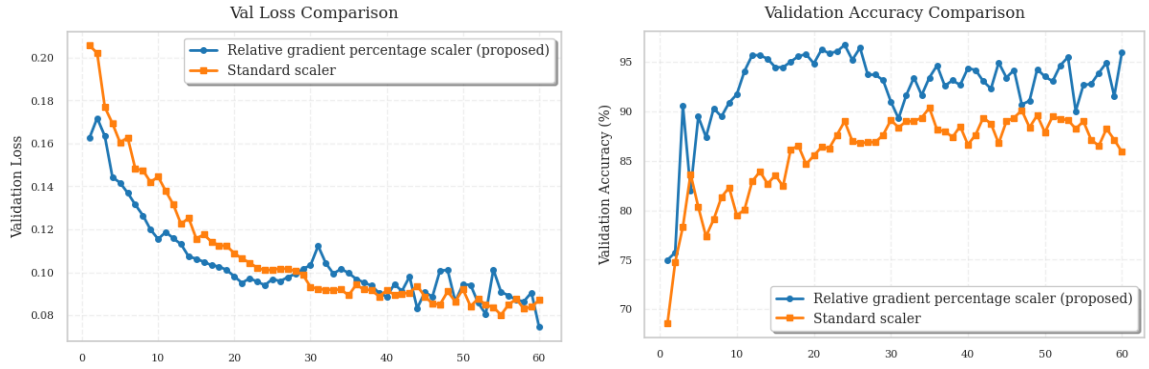
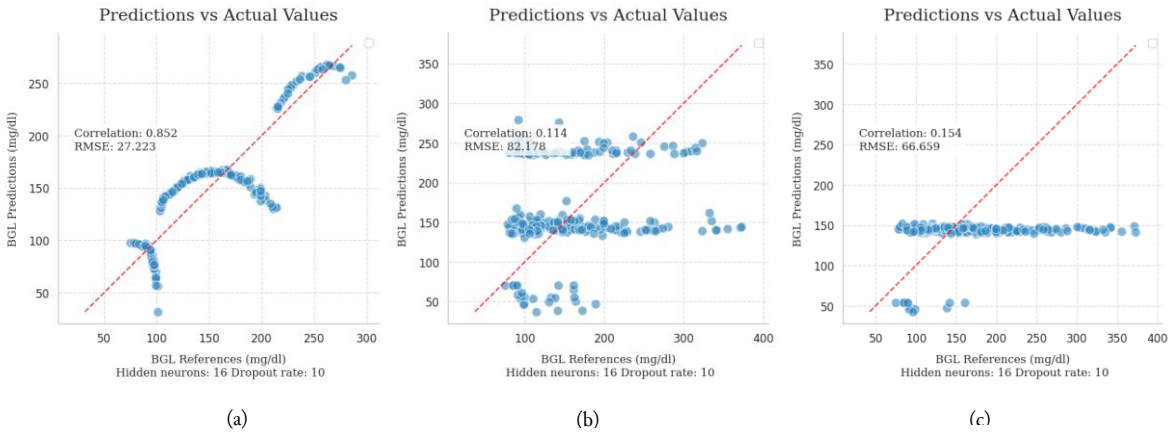


Fig. 11. Performance Comparison of the Proposed Scaling and Standard Scaler

3.3. Selected Balancing Method Performances

Regarding the correlation analysis in the previous section, we revealed that the Spearman and Kendall correlations of features and BGL are significantly higher than Pearson. The finding indicates that the relationships between features and BGL are nonlinear. Consequently, instead of employing SMOTE to tackle imbalance problems, we proposed range-balanced random sampling. As additional evidence, we explored some empirical studies regarding the performance comparison of the selected imbalance handler, SMOTE, and without a balance handler algorithm, as illustrated in Fig. 12. The other parameters were set to default, meaning we employed Fourier-based noise filtering, the proposed data scaling, the proposed hybrid network, and the proposed loss function. For the architecture, we set each convolutional operation to have only 16 filters, and each dense layer to have only 16 hidden neurons, small enough to observe the performance quickly.

The figure uncovers some insights, which are the no-balancing model results in predictions favoring the median of the data. This outcome is reasonable since 80% of our data points lie in the range of 0 – 100 mg/dl and 100 – 200 mg/dl. A similar pattern comes from the SMOTE balancing method, but it tends better, predicting the median values of class ranges. This occurs because SMOTE preforms data balancing, but suboptimal due to its reliance on linear interpolations, which does not suit the nature of BGL features. In contrast, the selected imbalance handling method was performed well, where the plot between actual and predicted BGL begins to synchronize, although not yet perfectly. This is expected given the very small architecture used.



(a) (b) (c)

Fig. 12. (a) Proposed balancing, (b) SMOTE, (c) No-Balancing Performance

3.4. Proposed Loss Performances

To validate the proposed loss function, we compared its performance with mean squared error (MSE) loss. We set the other parameters to their defaults. For the architecture, we set each convolutional operation to have only 16 filters, and each dense layer to have only 16 hidden neurons, small enough to observe the performance quickly. The proposed loss outperforms the MSE loss, both in RMSE and correlation metrics, as presented in Fig. 13.

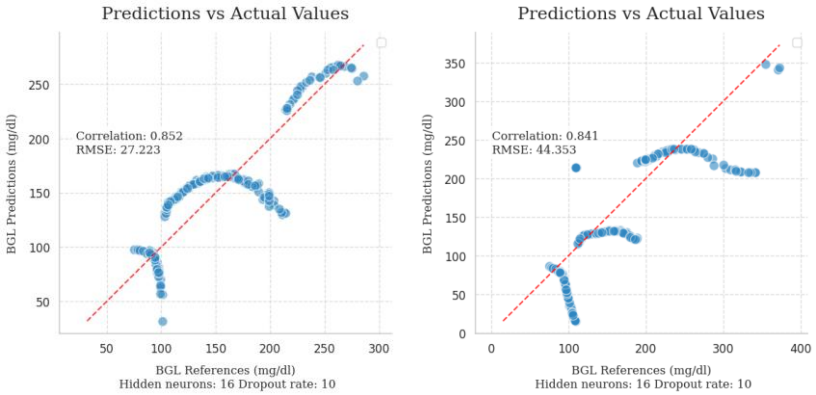


Fig. 13. Proposed loss performance and MSE loss performance

3.5. Model Learning Interpretations

We trained the proposed model and recorded the loss and accuracy profiles across epochs as shown in Fig. 14. The loss profile reveals the model's capability to analyze patterns. The training loss decreased steadily across epochs, suggesting that feature engineering strengthened the relationships between features and BGLs. Consequently, the model learns rapidly.

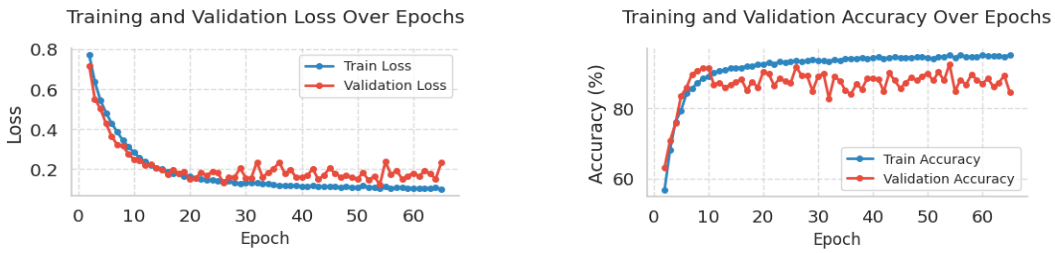


Fig. 14. Training and Validation Loss and Accuracy

Furthermore, the validation loss decreases alongside training loss, suggesting the model's capability to generalize patterns across features. Once the validation loss improved, indicating the best model performance was achieved, we saved the model. Furthermore, the accuracy profile describes our proposed loss's ability to prevent model predictions from clustering around the dataset mean during training. Fig.

14 shows that the accuracy profile increased steadily across epochs, revealing the proposed loss was achieving our expected goal. When the training accuracy achieved robust performance, we adjusted the loss function hyperparameters by lowering α and increasing the β parameter, shifting the model to focus more on regression tasks.

3.6. Proposed Architecture Evaluations

We compared the performance of the proposed hybrid, dense, and conv1d architectures. We set other parameters such as feature engineering, balancing method, and the loss function to default. We also used the same batch size, optimizer, momentum, dropout rate, learning rate decay, and learning rate patience. For the Conv1D training, we simply removed the dense block from the proposed architecture, while for the dense training, we excluded the Conv1D block of the architecture. The performance results are illustrated in Fig. 15.

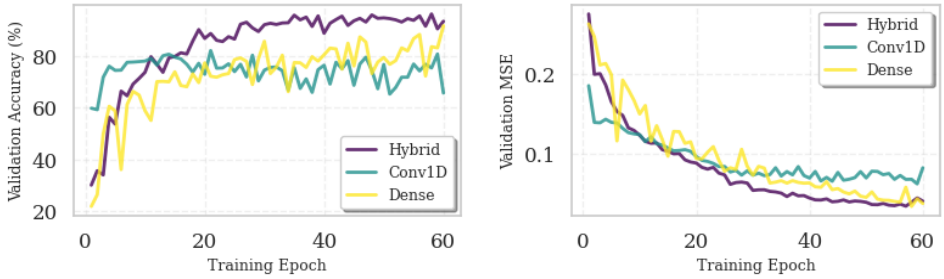


Fig. 15. Performance comparison of the hybrid, conv1d, and dense architecture

Based on validation data, the study found that the proposed hybrid architecture performs better than the dense-only or conv1d-only architectures. Furthermore, we evaluated model performance on the test dataset, as shown in Table 5, and reached a similar conclusion.

Table 5. Performance Comparisons of various Architecture Types

ML Model	Performance Matrice	
	MAPE	*RMSE*
Hybrid	17.79	27.22
Conv1D	27.86	38.05
Dense	20.68	36.75

In addition, we explored hybrid-architecture performance by increasing the number of convolutional filters and hidden neurons in each layer across various dropout rates, then evaluated performance, as shown in Table 6. The result reveals that a larger model size yields better predictions, achieving a MAPE of 14.11%. However, the study limits the model size to 64 filters for the convolutional layers and 64 neurons for each dense block layer. Future work will assess larger models.

Table 6. Performance Comparisons of Various Architecture Sizes

Conv1D Filter Number	Dense Hidden Neuron Number	Dropout Rate (%)	Performance Matrice		
			MAPE	*RMSE*	R^2
16	16	10	17.79	27.25	0.77
		15	20.73	27.22	0.85
		20	25.88	32.09	0.80
		25	25.76	35.88	0.74
32	32	10	15.26	26.46	0.90
		15	13.15	27.60	0.86
		20	19.98	35.32	0.81
		25	23.75	39.11	0.79
64	64	10	14.80	21.54	0.93
		15	17.75	30.59	0.89
		20	14.11	23.12	0.91
		25	14.15	23.72	0.89

3.7. CEG Analysis Evaluations

To evaluate our model's clinical safety and acceptability, we mapped the prediction results to the Clarke error grid (CEG), as illustrated in Fig. 16.

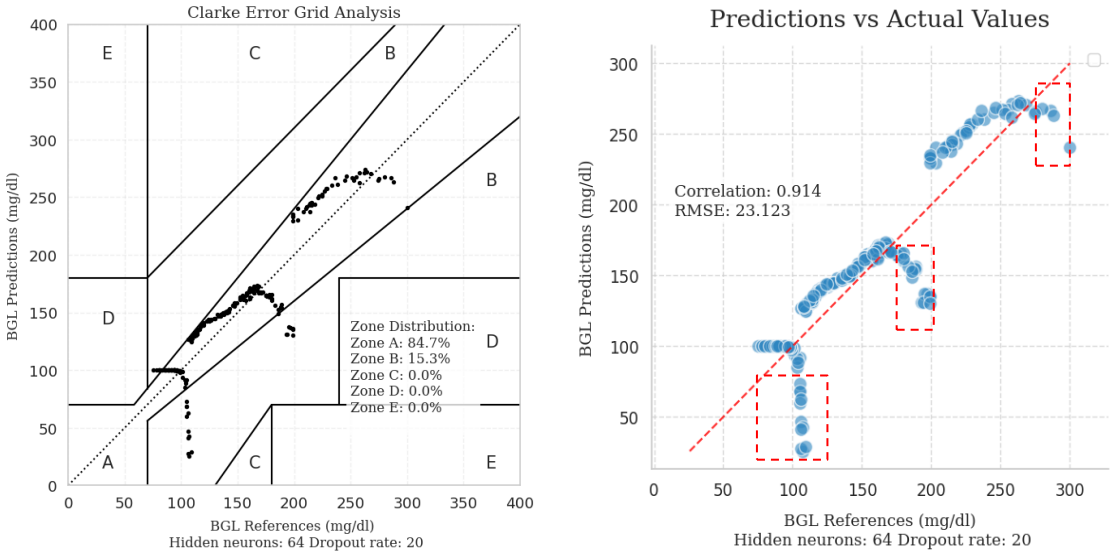


Fig. 16. Clarke Error Grid Analysis and Plot Model Predictions vs Actual References

According to the plot, 84.7% of the model predictions fall in zone A, which is clinically accurate, and the remaining 15.3% lie in zone B, which is clinically acceptable with minor errors. Based on these findings, we conclude that 100% of model predictions are clinically accepted and safe. Some other studies reported a higher percentage of predictions in zone A [7]. However, their studies were conducted in vitro and employed synthetic blood. Furthermore, the regression plot between the prediction and target values shows a 0.914 correlation, demonstrating the model's robust performance in predicting the BGLs. Of course, some data points are still predicted incorrectly, as indicated by red rectangles in Fig. 16. These discrepancies are likely caused by variations in patients' finger morphology (either too big or too small) or the presence of thick pigmentation. In addition, the figure shows that our model tends to be more precise at higher BGLs. This is because higher BGLs contribute more strongly to residual feature patterns than lower BGLs. It also confirms the illustration in Fig. 9, which shows higher BGLs have clearer signal patterns.

3.8. Comparison Results

In addition to CEG analysis, we validated our model with other machine learning algorithms, such as RF, XGBoost, and LightGBM. We trained these comparison models with MSE loss, reflecting the widely used approach for BGL regression tasks. The results (Table 7) show that our proposed model with modified loss achieves superior performance compared with RF, XGBoost, and LightGBM based on the MAPE evaluation metric. Moreover, the proposed model's percentage error is 14.11%, outperforming LGBM (38.75%), XGBoost (39.59%), and RF (41.71%) by more than 24 percentage points. The 14.11% MAPE is considered clinically acceptable because the defined accurate zone is 20% around actual BGL references based on the CEG analysis.

Table 7. Performance Comparisons

ML Model	MAPE (%)	
	Range 0 - 800 mg/dl	*Range 0 - 400 mg/dl*
LGBM	38.75	37.81
XGBoost	39.59	38.75
RF	41.71	40.64
Proposed model	14.11	14.15

4. Conclusion

This paper presents noninvasive BGL modeling, based on residual infrared signals, obtained in vivo throughout patients' fingers. The study presents substantial complexity due to the features not solely dependent on BGL. To address this, we performed feature engineering to assess hidden relationships between features and BGLs. Our proposed feature engineering discriminates against BGLs better, indicated by significant improvements in Spearman correlations. The evidence reveals that the methodology uncovers features with strong monotonic relationships with BGLs. In addition, the Pearson and Kendall correlations also demonstrate moderate improvements, indicating that the approach has robust positive impacts. These outcomes are validated by visualizing feature sequences across various BGLs and show clear separability. Furthermore, our proposed hybrid neural network architecture with a custom loss function demonstrates superior performance compared to other machine learning models including LightGBM, XGBoost, and RF, achieving 14.11% MAPE. To analyze clinical acceptability, we conducted CEG analysis, which revealed that 84.7% of our model predictions fell within zone A or clinically accurate, while the remaining 15.3% resided in zone B, or clinically acceptable. Finally, these results validate that our model achieved 100% clinically acceptable predictions, supporting its potential implementation for BGL measurements. Future work could explore the model's generalization across more diverse patient populations.

Acknowledgment

The authors would like to express their gratitude to IPB University and the LPDP endowment fund for supporting this study.

Declarations

Author contribution. All authors contributed equally in different ways and approved the paper.
Funding statement. The research was supported by IPB University and LPDP Endowment Fund
Conflict of interest. The authors declare no conflict of interest.
Additional information. The authors utilized artificial intelligence (AI) tools to assist in improving the language clarity and readability of the manuscript. The use of AI was limited to editorial support, and all content, interpretations, and conclusions remain the responsibility of the authors.

Data and Software Availability Statements

The code for our proposed model is available on GitHub.

References

[1] "Diabetes facts and figures," *International Diabetes Federation (IDF)*, 2023. [Online]. Available at: https://idf.org/about-diabetes/facts-figures/.

[2] R. Khan, Z. Chua, J. Tan, Y. Yang, Z. Liao, and Y. Zhao, "From Pre-Diabetes to Diabetes: Diagnosis, Treatments and Translational Research," *Medicina (B. Aires).*, vol. 55, no. 9, p. 546, Aug. 2019, doi: 10.3390/medicina55090546.

[3] J. Wu, Z. Peng, X. Zhang, W. He, and C. Yang, "Wavelet-ResNet: A deep residual network combined with wavelet transform for photoacoustic blood glucose detection," *Biomed. Signal Process. Control*, vol. 97, p. 106718, Nov. 2024, doi: 10.1016/j.bspc.2024.106718.

[4] C. Xiong, Z. Ren, and T. Liu, "Quantitative blood glucose detection influenced by various factors based on the fusion of photoacoustic temporal spectroscopy with deep convolutional neural networks," *Biomed. Opt. Express*, vol. 15, no. 5, p. 2719, May 2024, doi: 10.1364/BOE.521059.

[5] J. Parab, M. Sequeira, M. Lanjewar, C. Pinto, and G. Naik, "Backpropagation Neural Network-Based Machine Learning Model for Prediction of Blood Urea and Glucose in CKD Patients," *IEEE J. Transl. Eng. Heal. Med.*, vol. 9, pp. 1–8, 2021, doi: 10.1109/JTEHM.2021.3079714.

[6] A. Aloraynan, S. Rassel, M. R. Kaysir, and D. Ban, "Dual quantum cascade lasers for noninvasive glucose detection using photoacoustic spectroscopy," *Sci. Rep.*, vol. 13, no. 1, p. 7927, May 2023, doi: 10.1038/s41598-023-34912-3.

[7] S. Vanaja *et al.*, "Non-invasive glucometer monitoring system through optical based near-infrared sensor method," *Comput. Methods Biomech. Biomed. Eng. Imaging Vis.*, vol. 12, no. 1, p. 12, Dec. 2024, doi: 10.1080/21681163.2024.2327423.

[8] Satish, K. Sen, and S. Anand, "Analysis of glucose-dependent dielectric properties of aqueous-based solution: A proof of concept," *IET Sci. Meas. Technol.*, vol. 15, no. 7, pp. 562–568, Sep. 2021, doi: 10.1049/smt2.12057.

[9] R. P. Jenie *et al.*, "Non-Invasive Optical Blood Glucose Measurement based on Discrete Fourier Transform and Fast Artificial Neural Network: Fasting Normal Glucose Participants Case Study," *J. Med. Device.*, vol. 13, no. 4, p. 7, Jul. 2019, doi: 10.1115/1.4044336.

[10] Y. Yang *et al.*, "Noninvasive Blood Glucose Detection System With Infrared Pulse Sensor and Hybrid Feature Neural Network," *IEEE Sens. J.*, vol. 24, no. 8, pp. 13385–13394, Apr. 2024, doi: 10.1109/JSEN.2024.3373048.

[11] M. Naresh, V. S. Nagaraju, S. Kollem, J. Kumar, and S. Peddakrishna, "Non-invasive glucose prediction and classification using NIR technology with machine learning," *Heliyon*, vol. 10, no. 7, p. e28720, Apr. 2024, doi: 10.1016/j.heliyon.2024.e28720.

[12] G. Han, S. Chen, X. Wang, J. Wang, H. Wang, and Z. Zhao, "Noninvasive blood glucose sensing by near-infrared spectroscopy based on PLSR combines SAE deep neural network approach," *Infrared Phys. Technol.*, vol. 113, p. 103620, Mar. 2021, doi: 10.1016/j.infrared.2020.103620.

[13] L. M. Pires and J. Martins, "Design and Implementation of a Low-Power Device for Non-Invasive Blood Glucose," *Designs*, vol. 8, no. 4, p. 63, Jun. 2024, doi: 10.3390/designs8040063.

[14] Y. Zheng *et al.*, "Highly Sensitive Perovskite Photoplethysmography Sensor for Blood Glucose Sensing Using Machine Learning Techniques," *Adv. Sci.*, vol. 11, no. 43, pp. 1–10, Nov. 2024, doi: 10.1002/advs.202405681.

[15] S. V. K. R. Rajeswari and P. Vijayakumar, "Development of sensor system and data analytic framework for non-invasive blood glucose prediction," *Sci. Rep.*, vol. 14, no. 1, p. 9206, Apr. 2024, doi: 10.1038/s41598-024-59744-7.

[16] T. Kitazaki *et al.*, "Glucose emission spectra through mid-infrared passive spectroscopic imaging of the wrist for non-invasive glucose sensing," *Sci. Rep.*, vol. 12, no. 1, p. 20558, Nov. 2022, doi: 10.1038/s41598-022-25161-x.

[17] J. Hanna *et al.*, "Wearable flexible body matched electromagnetic sensors for personalized non-invasive glucose monitoring," *Sci. Rep.*, vol. 12, no. 1, p. 14885, Sep. 2022, doi: 10.1038/s41598-022-19251-z.

[18] S. Kiani and P. Rezaei, "Microwave substrate integrated waveguide resonator sensor for non-invasive monitoring of blood glucose concentration: Low cost and painless tool for diabetics," *Measurement*, vol. 219, p. 113232, Sep. 2023, doi: 10.1016/j.measurement.2023.113232.

[19] S. Harnsoongnoen and B. Buranrat, "Advances in a Microwave Sensor-Type Interdigital Capacitor with a Hexagonal Complementary Split-Ring Resonator for Glucose Level Measurement," *Chemosensors*, vol. 11, no. 4, p. 257, Apr. 2023, doi: 10.3390/chemosensors11040257.

[20] C. G. Juan *et al.*, "On the Selectivity of Planar Microwave Glucose Sensors with Multicomponent Solutions," *Electronics*, vol. 12, no. 1, p. 191, Dec. 2022, doi: 10.3390/electronics12010191.

[21] A. Kandwal *et al.*, "Designing Highly Sensitive Microwave Antenna Sensor with Novel Model for Noninvasive Glucose Measurements," *Prog. Electromagn. Res.*, vol. 176, pp. 129–141, 2023, doi: 10.2528/PIER22113002.

[22] T. Igbe, O. W. Samuel, J. Li, F. Kulwa, A. Kandwal, and Z. Nie, "Inspection of EEG Signals for Noninvasive Blood Glucose Monitoring in Prediabetes Diagnosis," in *2023 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, Jun. 2023, pp. 1–5, doi: 10.1109/MeMeA57477.2023.10171941.

[23] A. Othmane, I. Youkana, L. Kahloul, and S. Bourekkache, "AI-Based Prediction for Glucose Levels: A Comparative Study of Machine Learning and Deep Learning Approaches," 2024, pp. 231–245, doi: 10.2991/978-94-6463-496-9_18.

[24] X. Huang *et al.*, "Comparison of feature learning methods for non-invasive interstitial glucose prediction using wearable sensors in healthy cohorts: a pilot study," *Intell. Med.*, vol. 4, no. 4, pp. 226–238, Nov. 2024, doi: 10.1016/j.imed.2024.05.002.

[25] M. F. Rabby, Y. Tu, M. I. Hossen, I. Lee, A. S. Maida, and X. Hei, "Stacked LSTM based deep recurrent neural network with kalman smoothing for blood glucose prediction," *BMC Med. Inform. Decis. Mak.*, vol. 21, pp. 1–15, 2021, doi: 10.1186/s12911-021-01462-5.

[26] A. Gade, V. Vijaya Baskar, and J. Panneerselvam, "Hybrid model with optimal features for non-invasive blood glucose monitoring from breath biomarkers," *Biomed. Signal Process. Control*, vol. 88, p. 105036, Feb. 2024, doi: 10.1016/j.bspc.2023.105036.

[27] J. M. Lee, R. Pop-Busui, J. M. Lee, J. Fleischer, and J. Wiens, "Shortcomings in the Evaluation of Blood Glucose Forecasting," *IEEE Trans. Biomed. Eng.*, vol. 71, no. 12, pp. 3424–3431, Dec. 2024, doi: 10.1109/TBME.2024.3424665.

[28] Y. Shen and S. Kleinberg, "Personalized Blood Glucose Forecasting From Limited CGM Data Using Incrementally Retrained LSTM," *IEEE Trans. Biomed. Eng.*, vol. 72, no. 4, pp. 1266–1277, Apr. 2025, doi: 10.1109/TBME.2024.3494732.

[29] Q. Bian, A. As'arry, X. Cong, K. A. bin M. Rezali, and R. M. K. bin Raja Ahmad, "A hybrid Transformer-LSTM model apply to glucose prediction," *PLoS One*, vol. 19, no. 9, p. e0310084, Sep. 2024, doi: 10.1371/journal.pone.0310084.

[30] N. Niako, J. D. Melgarejo, G. E. Maestre, and K. P. Vatcheva, "Effects of missing data imputation methods on univariate blood pressure time series data analysis and forecasting with ARIMA and LSTM," *BMC Med. Res. Methodol.*, vol. 24, no. 1, p. 320, Dec. 2024, doi: 10.1186/s12874-024-02448-3.

[31] H. Nemat, H. Khadem, J. Elliott, and M. Benaissa, "Data-driven blood glucose level prediction in type 1 diabetes: a comprehensive comparative analysis," *Sci. Rep.*, vol. 14, no. 1, p. 21863, Sep. 2024, doi: 10.1038/s41598-024-70277-x.

[32] M. A. R. Refat, M. Al Amin, C. Kaushal, M. N. Yeasmin, and M. K. Islam, "A Comparative Analysis of Early Stage Diabetes Prediction using Machine Learning and Deep Learning Approach," in *2021 6th International Conference on Signal Processing, Computing and Control (ISPCC)*, Oct. 2021, pp. 654–659, doi: 10.1109/ISPCC53510.2021.9609364.

[33] A. Zarghani, "Comparative Analysis of LSTM Neural Networks and Traditional Machine Learning Models for Predicting Diabetes Patient Readmission," *arxiv Artif. Intell.*, pp. 1–21, 2024. [Online]. Available at: https://arxiv.org/pdf/2406.19980.

[34] A. Prabha, J. Yadav, A. Rani, and V. Singh, "Design of intelligent diabetes mellitus detection system using hybrid feature selection based XGBoost classifier," *Comput. Biol. Med.*, vol. 136, p. 104664, Sep. 2021, doi: 10.1016/j.compbiomed.2021.104664.

[35] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, vol. 13-17-August-2016, pp. 785–794, Aug. 2016, doi: 10.1145/2939672.2939785.

[36] M. Komorowski, D. C. Marshall, J. D. Salciccioli, and Y. Crutain, "Exploratory data analysis," *Second. Anal. Electron. Heal. Rec.*, pp. 185–203, Jan. 2016, doi: 10.1007/978-3-319-43742-2_15.

[37] R. Joeres, D. B. Blumenthal, and O. V. Kalinina, "DataSAIL: Data Splitting Against Information Leakage." pp. 1–18, Nov. 17, 2023, doi: 10.1101/2023.11.15.566305.

[38] T. Han *et al.*, "In Vivo Near-Infrared Noninvasive Glucose Measurement and Detection in Humans," *Appl. Spectrosc.*, vol. 76, no. 9, pp. 1100–1111, Sep. 2022, doi: 10.1177/00037028221092474.

[39] M. Yang *et al.*, "Miniaturized Optical Glucose Sensor Using 1600–1700 nm Near-Infrared Light," *Adv. Sens. Res.*, vol. 4, no. 3, Mar. 2025, doi: 10.1002/adsr.202300160.

[40] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient BackProp," 1998, pp. 9–50, doi: 10.1007/3-540-49430-8_2.

[41] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *32nd Int. Conf. Mach. Learn. ICML 2015*, vol. 1, pp. 448–456, Feb. 2015. [Online]. Available at: https://arxiv.org/abs/1502.03167v3.

[42] C. Li *et al.*, "YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications," *arxiv Artif. Intell.*, pp. 1–17, 2022. [Online]. Available at: https://github.com/meituan/YOLOv6.

[43] W. L. Clarke, "The Original Clarke Error Grid Analysis (EGA)," *Diabetes Technol. Ther.*, vol. 7, no. 5, pp. 776–779, Oct. 2005, doi: 10.1089/dia.2005.7.776.

[44] M. Zeynali, K. Alipour, B. Tarvirdizadeh, and M. Ghamari, "Non-invasive blood glucose monitoring using PPG signals with various deep learning models and implementation using TinyML," *Sci. Rep.*, vol. 15, no. 1, p. 581, Jan. 2025, doi: 10.1038/s41598-024-84265-8.