VahigoNet: Leveraging Deep Learning for Transparent and High-Performance Hypertension Prediction

Muhammad Hasnain[a], Nadeem Javaid[a,*,1], Imran Ahmed[b] and Nabil Alrajeh[c,*]

[a]ComSens Lab, International Graduate School of Artificial Intelligence, National Yunlin University of Science and Technology, Douliu, Yunlin 64002, Taiwan (hasnain.awan988@gmail.com, nadeemjavaidqau@gmail.com)

[b]School of Computing and Information Science, Anglia Ruskin University, Cambridge CB11PT, UK (imran.ahmed@aru.ac.uk)

[c]Department of Biomedical Technology, College of Applied Medical Sciences, King Saud University, Riyadh 11633, Saudi Arabia (nabil@ksu.edu.sa)

ARTICLE INFO

Keywords:
Hypertension Disease
Deep Learning
Vanilla Recurrent Neural Network
Google Network
Highway Network
VahigoNet
10-Fold Cross Validation
Paired t-Test
eXplainable Artificial Intelligence (XAI)
Local Interpretable Model-agnostic Explanations
SHapley Additive exPlanations

ABSTRACT

Hypertension continues to be a primary cause of global death, necessitating early and accurate forecasting for effective treatments. The existing methods have drawbacks such as class imbalance, poor modeling of sequential and spatial connections, high computation costs, and lack of interpretability, even though Deep Learning (DL) models offer possible solutions. To tackle these difficulties, we present VahigoNet, a novel blending DL model that incorporates vanilla recurrent neural networks for capturing temporal correlations, google network for extracting hierarchical spatial features, and highway networks for adaptive feature refinements. To achieve strong generalization, we utilize the synthetic minority oversampling technique for data balance. VahigoNet substantially outperforms baseline models, showing enhancements of 9.39% in accuracy, 10.27% in precision, 8.63% in recall, 9.39% in F1-score, and 3.10% in area under the curve-receiver operating characteristic. A 10-fold cross validation method is utilized to assess the model's generalizability, markedly reducing overfitting and improving robustness. A paired t-test is performed to evaluate statistical significance, demonstrating that the enhancements are substantial and clinically relevant. Additionally, explainable artificial intelligence methodologies, including local Interpretable model-agnostic explanations and Shapley additive explanations, are incorporated to provide both local and global perspectives on feature contributions. These explainability strategies enhance transparency, making VahigoNet a more Interpretable and clinically reliable model for hypertension prediction. The results demonstrate that VahigoNet is an exceptionally efficient and transparent method, achieving a balance between predictive capability and practical relevance in medical diagnostics.

1. Introduction

Healthcare is a broad term that encompasses a wide range of services and treatments intended to preserve, enhance, or recover health. Treatments, diagnostics, rehabilitation, and prevention measures are used to address a wide range of medical illnesses and disorders [1]. Healthcare plays a vital role in our lives since it directly affects our longevity, quality of life, and overall well-being. This includes initiatives that promote healthy lives and prevent disease by using resources including screenings, vaccinations, health education, and lifestyle counseling. Medical professionals can use tests, exams, and evaluations to identify health issues and provide appropriate treatment. This entails administering prescription medications, providing tailored therapy, and performing procedures. The hallmark of hypertension, also known as high blood pressure, is a regular injection of excessive blood force into the artery walls, as seen in Figure 1. Both the systolic and diastolic blood pressure values are expressed in millimeters of mercury (mmHg) [2]. Hypertension can lead to various health conditions which are mentioned as follows:

- Hypertension can damage arteries over the time, which leads to the risk of heart disease, heart attack, and stroke.
- Hypertension can impair the kidney's capacity to filter blood adequately, damaging the blood arteries within the kidneys.
- Eye blood vessels can be damaged due to high blood pressure, which can lead to blindness.
- Long-term hypertension can lead to cognitive deterioration and a higher risk of dementia.
- Possible decreased blood flow to the limbs, especially the legs.

Age, family history, obesity, a lack of physical exercise, excessive salt intake, smoking, and excessive alcohol use are all risk factors for developing hypertension. Lifestyle adjustments, including eating a nutritious diet, exercising regularly, managing weight, limiting alcohol use, and minimizing salt consumption, are remedies to treat hypertension. In some circumstances, healthcare providers prescribe medication to manage blood pressure.
People sometimes refer to hypertension as the silent killer because it does not show symptoms until it progresses. Regular blood pressure monitoring and careful treatment are vital for avoiding problems.

*Corresponding Authors: (nadeemjavaidqau@gmail.com, www.njavaid.com), (nabil@ksu.edu.sa)
ORCID(s):
[1] https://orcid.org/0000-0003-3777-8249

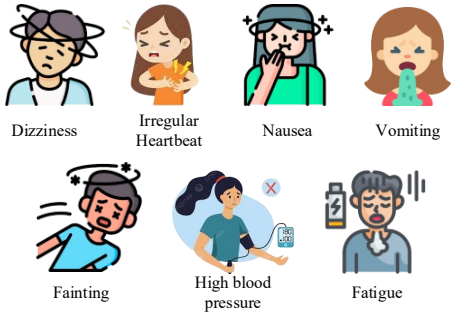


Figure 1: Hypertension Signs and Symptoms

Early identification and management are critical for preventing or reducing the severity of hypertension-related problems [3]. Deep Learning (DL) models are advanced and remarkable for hypertension prediction, but there are still some significant challenges that restrict their effectiveness and clinical utility. Due to the inherent imbalances in medical datasets, model predictions are skewed to favor the majority class while ignoring minority cases. Despite efforts to address this problem, data balancing methods like synthetic sample creation frequently fall short in maintaining decision boundaries, therefore raising the risk of overfitting and reducing generalization. Advanced data balancing techniques that improve model robustness while ensuring realistic sample distributions should be the main focus of future research. The majority of existing models focus on either sequential dependency modeling or spatial feature extraction, which results in an insufficient visualization of patterns associated with hypertension. Predictions are less than ideal when there is no thorough method that successfully combines temporal and geographical characteristics. To increase prediction accuracy and dependability, future research should examine ensemble architectures that smoothly integrate multi-scale feature learning. Although explainability methods provide insight into feature contributions, they frequently fall short of clinical reasoning. We need to develop more interpretable Artificial Intelligence (AI) frameworks that improve transparency and make integration into actual medical decision-making easier. The real-time inference is impractical due to the high computational resources required by high-performing DL models, especially in healthcare settings with limited resources. Despite previous models demonstrating improved accuracy, their real-world implementation is challenged by significant computational expenses and data imbalance, which this study addresses with an efficient ensemble architecture. Future research should prioritize the development of efficient architectures that balance predictive performance with feasibility, ensuring scalability and accessibility in clinical environments. Table 1 provides a comprehensive list of abbreviations and symbols.

1.1. Salient Contributions (Addressing the Research Gaps)

We propose significant additions that improve the prediction of hypertension by employing a new blending framework, efficient data balancing, thorough validation, and improved interpretability of the model. To address the challenges of hypertension prediction models, we proposed VahigoNet, an advanced blending architecture.

- To mitigate the impact of class imbalance, we employ Synthetic Minority Oversampling Technique (SMOTE), generating synthetic samples that enhance model generalization while reducing bias towards majority classes.
- The VahigoNets ensemble DL framework is proposed, which combines Vanilla Recurrent Neural Networks (VRNN) for temporal dependencies, Google Networks (GoogleNet) for spatial feature extraction, and Highway Networks (HighwayNet) for adaptive feature refinement. Hierarchical feature learning can capture complex spatial-temporal patterns and greatly improve classification accuracy and predictive robustness.
- We integrate 10-Fold Cross-Validation (10-FCV) to ensure model stability, reduce overfitting, and offer a more trustworthy evaluation across several data splits.
- We use paired t-tests to verify that the performance improvements of VahigoNet over baseline models are statistically significant, ensuring reliability in clinical applications.
- We use SHapley Additive exPlainations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME) to provide both local and global explanations of feature contributions, making predictions more transparent and trustworthy for medical professionals.

These contributions collectively enhance hypertension prediction by improving classification accuracy, reducing bias, ensuring stable training, and increasing clinical interpretability, making VahigoNet a robust and practical solution for real-world deployment.

2. Related Work

The section on related studies focuses on improving risk prediction and diagnostic accuracy. Current research on hypertension detection uses methods including Machine Learning (ML), DL, and signal processing. Although feature extraction techniques have improved model performance, issues such as data imbalance, unpredictability in patient data, and a significant barrier, the lack of properly labeled medical datasets, affect the generalization and reliability of the model.

In [4], *Zhao, H. et al.* compared the effectiveness of four ML algorithms. A dataset of 29,700 samples obtained during a physical examination is utilized to train and test the model.

Table 1
List of Abbreviations and Symbols

Abbreviation/ Symbol	Description
AUC-ROC	Area Under the Receiver Operating Characteristic Curve
ANN	Artificial Neural Network
DL	Deep Learning
HighwayNet	Highway Network
LIME	Local Interpretable Model-Agnostic Explanations
ML	Machine Learning
RNN	Recurrent Neural Network
SHAP	Shapley Additive Explanations
SMOTE	Synthetic Minority Over-sampling Technique
VahigoNet	Hybrid Google Network, Vanilla Recurrent Neural Networks, and Highway Network
VRNN	Vanilla Recurrent Neural Network
10-FCV	10-Fold Cross Validation
$C(x)$	Carry gate $C(x) = 1 - T(x)$
F_{concat}	Concatenated feature vector $[F_{\text{VRNN}}, F_{\text{GoogleNet}}, X_{\text{norm}}]$
$F_{\text{GoogLeNet}}$	Features extracted from GoogleNet
F_{highway}	Highway output $F_{\text{highway}} = T \odot F_{\text{trans}} + C \odot F_{\text{concat}}$
F_{trans}	Transformed feature $F_{\text{trans}} = \text{ReLU}(W_F F_{\text{concat}} + b_F)$
F_{VRNN}	Features extracted from VRNN
$H(x)$	Non-linear transformation $H(x) = \text{ReLU}(W_H x + b_H)$
H_i	Hidden layers with ReLU activation ($i = 1, 2, 3, 4$)
L	Loss function (Binary Cross-Entropy)
$T(x)$	Transform gate $T(x) = \sigma(W_T x + b_T)$
X	Input feature vector
X_{norm}	Normalized input feature vector
y	Final model prediction $y = \sigma(W_o H_4 + b_o)$
σ	Sigmoid activation function
$\odot$	Element-wise multiplication

Using univariate Linear Regression (LR) analysis, the authors first gathered risk variables related to hypertension. Next, five models were optimized using 10-FCV: Random Forest (RF), Categorical Boosting (CatBoost), Multilayer Perceptron (MLP), Neural Network (NN), and LR. Using measures like accuracy, sensitivity, and specificity, the efficacy of each model on the test set is evaluated in light of the experimental results. The RF performed better than the other four tested models. Most importantly, it achieved 0.82 accuracy, 0.83 sensitivity, and 0.81 specificity.

To enhance the performance of ML models in detecting hypertension, *Montagna, S. et al.* use five different ML algorithms, applying different balancing techniques to address the challenges and sensitivity and specificity in hypertension detection [5]. According to the study, RF outperformed traditional medical procedures in screening initiatives. It obtained a 0.629 specificity and a 0.818 sensitivity. However, no single setup or technique demonstrated a notable overall benefit. These methods demonstrated a trade-off between identifying real positives and preventing False Positives (FP), with a greater sensitivity of 0.906 and a significantly lower specificity of 0.230.

To predict hypertension in high-risk patients, Asadullah, M. *et al*. use ML methods (Naive Bayes (NB), Support Vector Machine (SVM), LR, Random Forest (RF)) [6]. The proposed hybrid model outperforms existing ML techniques with a prediction accuracy of 78.17%. RF has the highest accuracy of 73.86%. Different evaluation metrics and confusion matrices through 10-FCV were used to assess the performance of the classification process.

Paper [7] includes four cases of ML and DL-based approaches for predicting cardiovascular cases and hypertension classification. In the first case, DL and SVM predict the incidence of cardiovascular cases with 56%–57% accuracy. Based on a database of 378256 patients, the Neural Network (NN) method in the second case predicted the occurrence of cardiovascular events during a 10-year follow-up with 68 percent sensitivity and 71 percent specificity. In the last example, 1,504,437 people were classified according to whether or not they have hypertension using an ML method that has a 51% sensitivity, 99% specificity, and an 87% Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

The authors in [8] used Heart Sound (HS) signals to detect Pulmonary Hypertension (PH), which is a serious condition. They have developed a new strategy using a Wavelet Scattering Convolution Network (WSCN) combined with Long Short-Term Memory (LSTM) to detect the PH. Their model was validated using five cross-validations with 131 people. Six distinct training procedures and models subject-dependent, subject-independent, and classification with and without feature extraction were compared in this work. According to recent studies, the WSCN+LSTM approach performs the best for PH detection till date.

The Korean National Health Insurance Service database was used by the authors in [9]. The progression of cardiovascular disease, including heart failure, myocardial infarction, and stroke, was tracked until 2018 in a group of people with hypertension who had more than five health examinations between 2002 and 2011 (n = 357036). Oversampling was employed to compensate for the disparity in sample counts between the case and control groups. The mean standard deviation's lowest and highest values from five measurements were included in an LR model for each explanatory variable. LSTM was the method used to establish the DL base models [17]. All four models were set up utilizing both LR and DL methodologies: one trained on the entirely oversampled dataset, one on the non-dose group (patients not receiving antihypertensive medication), one on the drug-dose group incorporating drug-related variables, and one on the drug-dose group excluding drug-related variables.

Authors in [10] aim to create a DL system based on the Korean Genome and Epidemiology Study-Ansan and Unsung baseline survey for the categorization and prediction of hypertension with Blood Pressure (BP) related variables. The authors constructed a Deep NN (DNN) using experimentally modified numbers of hidden layers and nodes inside each hidden layer. Using the dataset, they trained the DNN to detect hypertension while experimenting with four different energy intake adjustment methods. The authors trained a Decision Tree (DT) in the same manner for comparison. From the results, it was clear that DNN outperforms DT, having more accuracy, F1-score, sensitivity, and specificity.
In [11], the authors proposed a multi-input and multi-output DNN to predict mean blood pressure readings over a period of 4 weeks. They used a clinical trial dataset that contains medical examination data, such as gender, blood pressure, and age. The study employed Root-Mean-Square Error (RMSE) and Standard Deviation Ratio (SR) as performance parameters. The proposed model's outcomes were additionally validated through cross-validation. According to the results, the RMSE ranged from 5.04 to 6.65 mmHg, and the SRs from 0.67 to 0.70 mmHg.
The dataset used by Ye, X., Zeng, *et al.* was taken from the Electronic Health Records (EHR) database to predict hypertension in individuals [12]. Authors developed risk-adapted models using Recurrent NN (RNN), which includes LSTM and Bi-directional LSTM (Bi-LSTM), to forecast the likelihood of achieving the blood pressure control objectives connected to various blood pressure treatment plans. The dataset is split in a 6:2:2 ratio, where 60% is used for training, 20% for validation, and 20% for testing. This ratio ensures effective learning, model tuning, and unbiased performance evaluation. Each set's samples were from distinct years that were randomly selected with matching proportions.
In [13], authors proposed a fine-grained Heart Rate Variability (HRV) analysis approach to distinguish between patients with hypertension and healthy individuals. To be more precise, they first split the eight hours of Electrocardiogram (ECG) data into five-minute segments. The heartbeat interval was then better extracted from ECG data using their proposed method, and 22 HRV features in both the linear and nonlinear domains were extracted along with a histogram. Specifically, the authors represent the distribution of the heartbeat interval of each time frame using a Gaussian mixture model. The authors then examined the relationship between HRV's linear and nonlinear domain aspects. Finally, they train a hypertension recognition model using popular ML methods. They used real-world clinical ECG data from 138 hypertension patients and 138 healthy individuals for this paper's data set.
By merging the K-Nearest Neighbor (KNN) and Light Gradient-Boosting Machine (LightGBM) algorithms, the authors in [14] present a model for forecasting the risk of hypertension within the following five years. They choose this strategy to increase prediction accuracy and lower the rate of misdiagnosis, especially considering the high prevalence and poor rates of hypertension control in low and middle-income countries. Their hybrid model performs better than SVM, RF, and a 6-layer NN, among other techniques. To further improve the accuracy of the model, the authors also intend to include other risk variables like smoking, alcohol addiction, and family history.
Despite significant improvements in ML and DL for hypertension prediction, issues such as class imbalance, varying generalization, and restricted interpretability remained. Many existing methods focus on either temporal or spatial feature extraction, but they do not always effectively combine the two, which results in less reliable predictions. Additionally, biases are introduced by imbalanced datasets, and clinical adoption is limited by explainability issues. Future research should prioritize multi-scale feature learning, sophisticated data balancing, and uncertainty-informed decision-making to improve model reliability. Furthermore, incorporating explainability techniques would enhance transparency, making AI-driven hypertension prediction more interpretable and medically reliable.

3. Material and Methods

Here is a comprehensive and technically refined description of the DL model.

3.1. Vanilla Recurrent Neural Network

A VRNN is a basic DL architecture intended for the analysis of sequential input by maintaining a hidden state that incorporates temporal dependencies throughout time steps. Unlike feedforward networks, VRNNs have connections that loop back on themselves, allowing them to remember information, which makes them very useful for tasks like predicting time series, understanding language, and classifying sequences [15]. In a VRNN, each input at a certain time step is processed in conjunction with the preceding hidden state to produce a new hidden state. The VRNN utilizes the hidden state to generate predictions and transmits it to the subsequent time step. The hidden state at time step t is calculated as follows:

$$h_t = \tanh(W_h h_{t-1} + W_x x_t + b_h) \tag{1}$$

where h_t represents the current hidden state, h_{t-1} signifies the previous hidden state, x_t represents the input, W_h and W_x are adjusting weight matrices, and b is the bias term. The activation function, commonly *tanh or Rectified Linear Unit (ReLU)*, introduces non-linearity to facilitate the network's learning of complex patterns. In classification tasks, the final hidden state is processed through a fully connected dense layer with a softmax or sigmoid activation function, depending upon the number of output classes. The model is optimized utilizing loss functions such as categorical cross-entropy for multi-class classification or binary cross-entropy for binary classification, employing optimizers like Adam or SGD to update weights effectively. VRNNs, despite their effectiveness, face challenges from the vanishing gradient problem, so that gradients decrease during backpropagation, complicating the capture of long-range relationships

[16]. The issue is mitigated by sophisticated architectures like LSTM and Gated Recurrent Units (GRU), which employ gating mechanisms to preserve long-term information. LSTM incorporates a memory cell and three gates (input, forget, and output) that regulate the flow of information, facilitating the effective modeling of long-range temporal dependencies [17]. GRU is a simpler version of LSTM that merges the forget and input gates into one update gate, which makes it faster to compute while still keeping important time-related information [18]. However, for short sequential patterns and simplified models, VRNNs continue to be an efficient and computationally effective solution.

3.2. GoogLe Network

The GoogleNet is a sophisticated DL framework designed for complex predictive modeling applications involving dynamic and adaptive learning. GoogleNet is very proficient in addressing challenges related to both structured and unstructured data, utilizing their capacity to generate significant representations from high-dimensional inputs [19]. Through the integration of advanced architectures, GoogleNet dynamically enhances feature extraction processes, providing optimal performance in applications such as Natural Language Processing (NLP), medical diagnostics, and autonomous decision-making. Their versatility enhances generalization, exceeding traditional DL models that depend on fixed feature mappings. The GoogleNet architecture utilized for predictive analytics in our work consisted of several deep layers, each facilitating hierarchical feature extraction. The early layers comprise dense units activated by ReLU functions, enabling non-linear transformations crucial for capturing complex input dependencies. A softmax activation layer concludes the network by transforming outputs into probability distributions appropriate for multiclass classification tasks, such as forecasting possible illness development. The attention mechanism integrated within the design is essential, selectively enhancing important characteristics while reducing noise, thereby enhancing interpretability and decision-making [20]. The calculated attention weights for the input sequence are as follows:

$$\alpha_i = \frac{\exp(e_i)}{\sum_j \exp(e_j)} \tag{2}$$

where e_i represents the attention score for the i-th input element as follows:

$$e_i = v^T \tanh(W h_i + b) \tag{3}$$

In these circumstances, W and v represent learnable parameters, h_i denotes the hidden state at position i, and b represents a bias term affecting the scoring function. The context vector c is computed as a weighted sum of the hidden states, each contributing differently to the final prediction as follows:

$$c = \sum_i \alpha_i h_i \tag{4}$$

By dynamically referencing essential data points, GoogleNet excels at capturing long-range dependencies across subsequent inputs, making it extremely suitable for time-sensitive applications. Using categorical cross entropy loss and ReLU-activated layers helps the model learn better by allowing gradients to flow effectively. Moreover, the capacity to dynamically adjust attention weights enhances forecast accuracy, facilitating the effective modeling of intricate situations. This adaptable architecture exhibits considerable promise in areas necessitating real-time decision-making, including financial forecasts and medical diagnostics, highlighting its transformational influence in AI-driven solutions.

3.3. Highway Network

HighwayNet are a type of DNN designed to mitigate the vanishing gradient issue in extremely deep designs. By incorporating gating mechanisms, HighwayNet facilitates efficient gradient propagation, making it especially effective for simulating sequential and nonlinear patterns in medical data processing [21]. A HighwayNet-based architecture was utilized to forecast and evaluate the risk of diabetes progression through historical patient records. The model consisted of several layers, each using modification and gating algorithms to enhance feature extraction. Each highway layer is configured as a gated unit, whereby a transformation function $T(x)$ and a carry function $C(x)$ control the information flow within the network as follows:

$$H(x) = T(x, W_T) \cdot G(x, W_G) + x \cdot C(x, W_C) \tag{5}$$

Here, $H(x)$ is the highway layer's output, $T(x, W_T)$ is the transformed representation, and $C(x, W_C)$ determines the proportion of the original input x that is kept. The gating function $G(x, W_G)$ is calculated with a sigmoid activation function, therefore facilitating adaptive control of feature propagation, as detailed follows:

$$G(x, W_G) = \sigma(W_G x + b_G) \tag{6}$$

The architecture utilized ReLU activation functions in intermediate layers to introduce non-linearity, while the final layer applied a sigmoid activation to produce probabilistic scores appropriate for binary classification tasks. We refined the model using binary cross-entropy loss to enhance its prediction efficiency. The gating technique was crucial in dynamically selecting important features, enabling the network to capture complex dependencies within sequential medical data. For effective diabetes prediction, HighwayNet emphasized essential data representations, intentionally retaining crucial information and eliminating unnecessary features [22]. The model successfully assessed patient histories by ensuring a seamless flow of information between layers, making it ideal for long-term risk assessment. This systematic methodology highlights the capabilities of HighwayNet in medical AI applications, facilitating the early identification of complex health issues via deep hierarchical modeling.

4. Proposed System Model for Hypertension Prediction

This section introduces the proposed system model for hypertension prediction and presents VahigoNet, a sophisticated DL framework, as shown in Figure 3. A custom Inception-based GoogleNet is used to extract multi-scale spatial features, and a VRNN is used to capture temporal dependencies. A HighwayNet serves as a meta-learner for adaptive feature refinement. The pipeline starts with thorough data preprocessing, which includes normalizing feature distributions using Min-Max scaling. Next, the data is balanced using SMOTE, which generates synthetic samples to address class imbalance. To provide strong model generalization, the preprocessed data is subsequently divided into training, validation, and testing sets. After processing the input data, VRNN and GoogleNet both extract deep hierarchical features. These features are then concatenated and provided to the HighwayNet. By utilizing transform and carry gates to optimize feature flow, the HighwayNet improves non-linear transformations while maintaining crucial data. VahigoNet's ability to understand complex patterns in the data helps it perform better in predicting hypertension through a blending method. This approach ensures transparency and reliability in hypertension predictions. The methodology used in this paper is mentioned in Table 2.

4.1. Dataset and Preprocessing

The dataset used in this study is the National Family and Health Survey-5 (NFHS-5) [23], which is a demographic and health survey with several features of household characteristics, health, and socioeconomic position. It has 95 features; some of them are state, health indicators *(height, weight, anemia level), education level, water source, toilet facility, household assets, number of household members, total children born, child mortality, pregnancy complications*, and others. It also covers different health diseases such as *hypertension, alcohol consumption*, and *diabetes*. The entries reflect people from Jammu and Kashmir, with each row referring to a separate respondent. The dataset's combination of numerical and categorical variables enables a comprehensive analysis of demographic and health trends. It offers details on prenatal care, postnatal care, and birth spacing, among other aspects of maternal health. It also measures nutritional status indicators like stunting in children, dietary intake, and Body Mass Index (BMI). Economic factors that are associated with respiratory disorders include income levels, work status, and the type of cooking fuel used. Together with information on children's vaccination histories and coverage, the survey also gathers information on reproductive health, such as fertility preferences and the use of contraceptives. Additionally, the dataset allows for regional comparisons of health disparities because it includes geospatial information on respondents living conditions.

Data preparation turns raw data into an understandable, organized form that can be analyzed [24]. It is a crucial step in the data analysis process because datasets usually have missing values, outliers, noise, or inconsistencies, which can affect model performance. In our work, we used Min-Max scaling to normalize the numerical features within a predetermined range, usually [0, 1]. The process ensures that the performance of the model is not unduly impacted by features with varying scales. Min-Max scaling preserves the original data distribution while standardizing feature values, which speeds up convergence during training and enhances model interpretability [25]. When working with DL models, this step is especially helpful because it keeps significant numerical variations from unduly influencing weight updates. The Min-Max scaling transformation is defined as follows:

$$X_{\text{scaled}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{7}$$

where X is the original feature value, $X_{\min}$ is the minimum value of the feature, $X_{\max}$ is the maximum value of the feature, and X_{scaled} is the transformed value within the range [0, 1]. This transformation ensures that each feature contributes proportionately to the learning process, preventing any characteristic from overshadowing others due to its magnitude. Consequently, min-Max scaling aids in stabilizing gradient updates, reducing training time, and improving the overall performance of DL models. Table 3 provides a description of the dataset's key features.

4.2. Data Balancing using Synthetic Minority Over-sampling TEchnique

SMOTE is a popular technique that creates synthetic samples for the minority class to address the class imbalance in EHR. This technique is quite useful in classification problems where the dataset is highly imbalanced; there are significantly fewer instances in one class than in the other. SMOTE creates new synthetic samples by interpolating between existing instances rather than merely replicating minority-class samples. This method lessens bias toward the majority class and provides a more balanced dataset, which aids models in generalizing more effectively. Also, SMOTE helps the model find more examples of the minority class, which lowers the number of False Negatives (FN) in important situations. Figure 2 shows the step-by-step process of SMOTE balancing. It shows how synthetic data points are added between existing minority class samples to make the dataset more balanced, and this approach is inspired by [26] and modified for our method. The first step in the SMOTE process is choosing a random sample x_i from the minority class [27]. Next, it uses Euclidean distance to determine the sample's k-nearest neighbors within the same class. A synthetic instance x_{new} is made along the line that connects the original sample x_i and its chosen neighbor x_{nn} after one of these neighbors is randomly selected. Mathematically, this interpolation is performed as follows:

$$x_{new} = x_i + \lambda \times (x_{nn} - x_i), \quad \lambda \sim U(0, 1) \tag{8}$$

where λ is a random number drawn from a uniform distribution between 0 and 1, ensuring that the new sample lies on

Table 2
Methodology Used in the Study

Method	Description	Key Features	Application
Min-Max Scaling	Normalizes features to a fixed range (0 to 1) to ensure uniformity and prevent dominance of large values.	Preserves original data distribution, speeds up model convergence, and improves stability.	Used in preprocessing to standardize numerical data before model training.
SMOTE	SMOTE generates artificial samples to balance imbalanced datasets.	Reduces bias towards majority class, improves model generalization, and enhances classification performance.	Applied to handle a class imbalance in hypertension prediction.
VahigoNet	An ensemble model integrating VRNN, GoogleNet, and HighwayNet for multi-scale feature learning.	Captures both spatial and sequential dependencies, improves classification accuracy, and enhances robustness.	Used for hypertension prediction by combining multiple architectures for better feature extraction.
10-FCV	The dataset is split into 10 subsets, iteratively training and validating to ensure reliable evaluation.	Reduces overfitting, provides a more stable performance estimate, and ensures model generalization.	Applied to validate VahigoNet's performance across multiple data partitions.
Paired t-Test	Statistical test to compare two dependent models on the same dataset.	Measures whether performance improvements are statistically significant.	Used to validate that VahigoNet's performance improvements over HighwayNet are statistically significant.
LIME	LIME generates locally Interpretable approximations for complex models.	Provides instance-based explanations, highlights influential features, and improves model transparency.	Used to explain individual predictions of the VahigoNet model in hypertension classification.
SHAP	SHAP calculates feature contributions to model predictions based on cooperative game theory.	Measures global and local feature importance, ensures model interpretability and improves decision-making.	Used to analyze how different features impact VahigoNet's predictions for hypertension detection.

the line segment connecting x_i and x_{nn}. Until the required quantity of synthetic samples is produced, the procedure is repeated, increasing the minority class's representation [29].

4.3. Rationale behind VahigoNet for Hypertension prediction

VahigoNet is a sophisticated blending ensemble learning technique that integrates several DL models to maximize classification performance, as shown in Figure 3. For base-level feature extraction, the proposed approach methodically blends VRNN and a custom Inception-based GoogleNet model. A HighwayNet serves as the meta-learner, utilizing the advantages of both models to improve and refine classification accuracy. The blending process used by VahigoNet follows a structured pipeline. To provide better model convergence and stability, the dataset is first preprocessed using Min-Max scaling, which normalizes the features within a specified range. For objective analysis and reliable performance evaluation, the dataset is then divided into training set of 64% data, validation set of 16%, and a test set of a 20% ratio. Features are extracted using a custom Inception-based GoogleNet that employs convolutional layers to extract multi-scale hierarchical features and VRNN, which leverages its recurrent structure to capture latent representations and sequential relationships. Both models create their own feature representations for the validation and test datasets, and these features are then arranged into dataframes *(df_val and df_test)* to simplify processing. A HighwayNet, which consists of multiple essential elements, is used as the meta-learner to combine and improve the retrieved features. Selective modification of input features is made possible by the transform gate, a *sigmoid*-activated dense layer that dynamically controls the feature flow shown in Figure 4. Furthermore, the carry gate is defined as a $1 -$ transform gate, ensuring an identity mapping that preserves crucial information from the input features. Additionally, a ReLU activated dense layer is used for feature transformation, which enhances the expressiveness and representation of features by applying non-linear transformations. Lastly, element-wise addition is used to merge the modified and identity features to execute residual learning, which minimizes information loss and promotes

Table 3
Important Features Related to Hypertension

Attribute	Values	Description
Res_Age	Numeric (years)	Age of the respondent
Edu_level	Categorical (e.g., No education, Primary, Secondary)	Education level of the respondent
Wealth_Idx_Lb	Categorical (e.g., Poor, Middle, Rich)	Household wealth index
House_electricity	Binary (Yes/No)	Availability of electricity at home
Smoke	Binary (Yes/No)	Smoking habits of the respondent
Alcohol	Binary (Yes/No)	Alcohol consumption by the respondent
Diabetes	Binary (Yes/No)	Presence of diabetes in the respondent
Resp_weight	Numeric (kg)	Weight of the respondent
Resp_height	Numeric (cm)	Height of the respondent
Hg_levelAdjusted	Numeric (g/dL)	Adjusted hemoglobin level
Anemia_level	Categorical (e.g., None, Mild, Moderate, Severe)	Level of anemia
Curr_MaritalStatus_Single Parent	Binary (Yes/No)	Marital status indicating if the respondent is a single parent
Religion_Muslim	Binary (Yes/No)	Indicates if the respondent follows Islam
ResidenceType_Urban	Binary (Urban/Rural)	Type of residence
Smoke_atHome	Binary (Yes/No)	Exposure to smoking at home

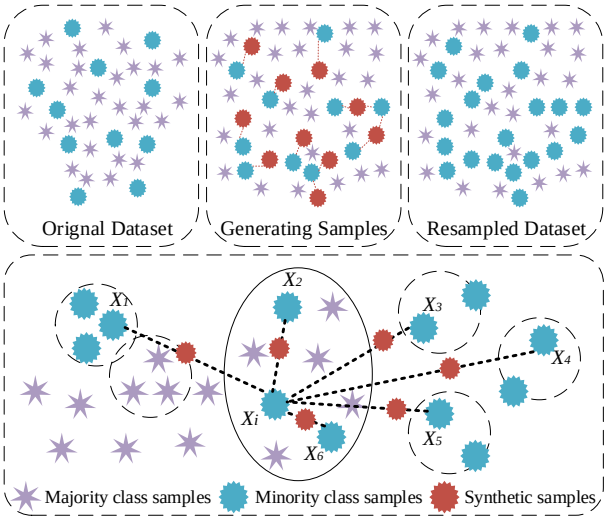


Figure 2: Synthetic Minority Over-sampling TEchnique Enhanced Data Balancing: Synthetic Sample Generation for Class Imbalance Correction.

effective information propagation. The HighwayNet other hidden layers improve feature representations even more before sending them to the last classification layer. The output layer uses a *sigmoid* activation function to generate binary predictions. To ensure efficient optimization and learning, the model is trained for 20 epochs using the *Adam* optimizer with binary cross-entropy loss. VahigoNet captures sequential dependencies and multi-scale patterns to enhance feature fusion. It makes use of HighwayNet for adaptive feature refinement; VRNN for temporal feature extraction; and GoogleNet for spatial feature learning to provide a reliable and effective decision-making process. Algorithm 1 provides a systematic explanation of the specific steps of the proposed approach. The first line of the algorithm, Line 1, receives as input a preprocessed dataset $D = (X, Y)$, where X stands for feature vectors and Y contains corresponding labels for the labeling of hypertension. According to Line 2, the algorithm's final output is a binary classification result that predicts if a patient has hypertension or not. The first step in Line 3 is data preprocessing, which is essential for enhancing convergence and stabilizing model training. To prevent features with huge magnitudes from controlling the training process, Min-Max scaling is used in Line 4 to normalize feature values within the range $[0, 1]$. The dataset is then split into training, validation, and test sets in Line 5 to enable reliable model evaluation and avoid overfitting, this procedure is shown in Figure 4. Base models that extract features from the input data are the focus of the following phase, which begins at Line 6. Start at Line 7, where each GoogleNet module applies parallel convolutional operations with different kernel sizes (1×1, 3×3, 5×5) to capture multi-scale features (Line 8 and Line 9). GoogleNet is used for spatial feature extraction. A multi-scale representation is also created by using max pooling in Line 10 and concatenating the outputs from several layers in Line 11. The feature extraction loop terminates at Line 12, and the extracted feature representations are stored at Line 13. At the same time, temporal feature extraction uses the VRNN model, starting at Line 14. Using a non-linear transformation of the input and the previous hidden state (Line 16), the Algorithm iterates through each time step t in Line 15, updating the hidden state. A weighted transformation of the hidden state is used in Line 17 to compute the output, capturing temporal dependencies. The loop finishes at Line 18, and Line 19 provides the deep feature representation from VRNN. The extracted features from GoogleNet and VRNN are concatenated in Line 21 in Step 3, starting with Line 20, to create a single feature representation. For further processing, this combined feature set is organized in Line 22. The VRNN and GoodLeNet features are concatenated and provided as an input to the HighwayNet, as shown in Figure 4. Starting at Line 23, Step 4 provides the HighwayNet for final classification. A sigmoid activation function is used in Line 24 to calculate the transform gate $T(x)$, which handles the total number of transformed data that is sent through the network as shown in Figure 4. The extracted feature

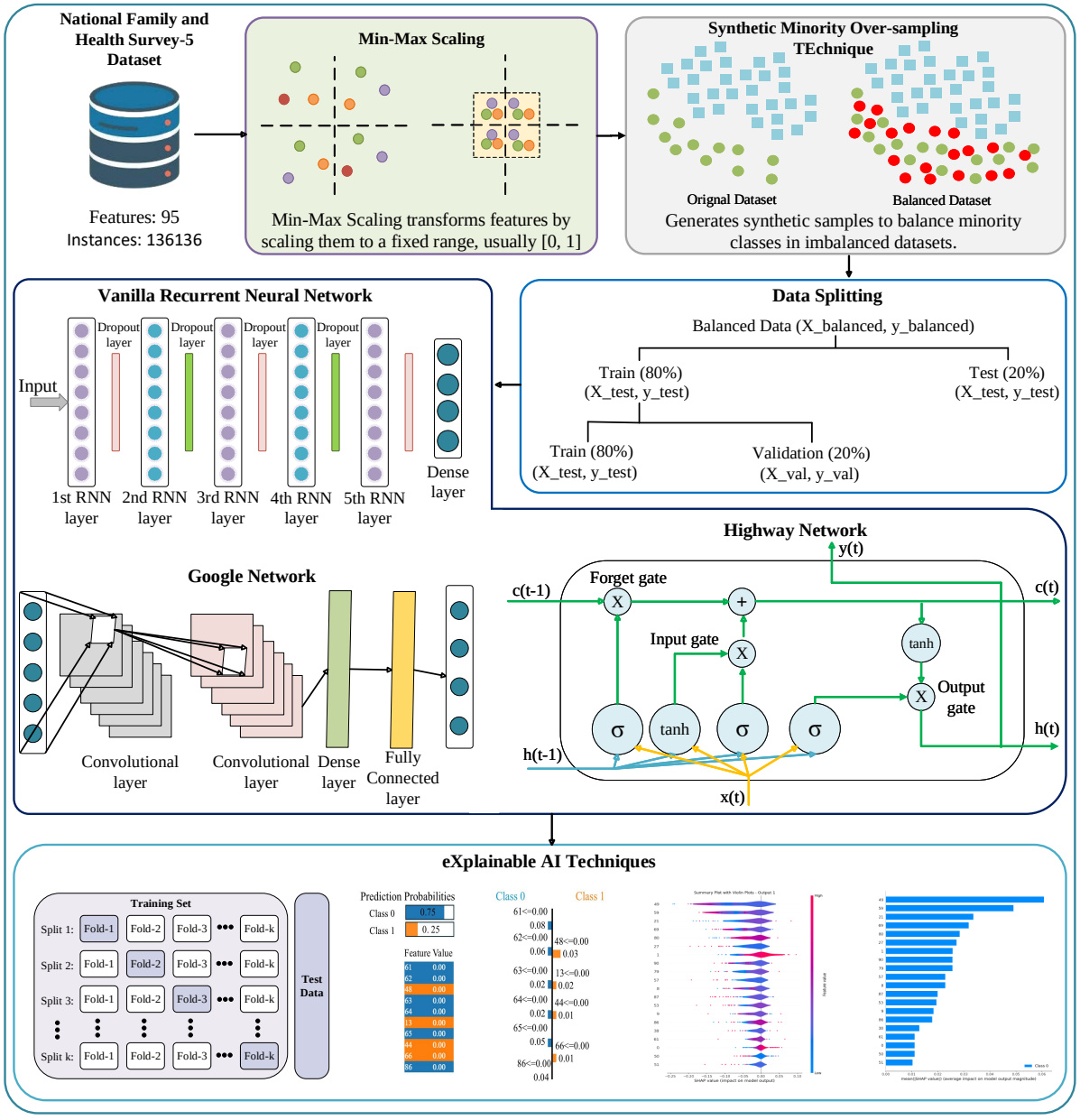


Figure 3: Proposed System Model for Hypertension Prediction

representations are then refined in Line 25 by performing the feature transformation using a ReLU activation function. Therefore, to ensure the essential raw feature information is retained, the transform gate and the carry gate $C(x) = 1 - T(x)$ are used to balance the converted and original features when computing the highway output in Line 26. Line 27 provides further hidden layers with ReLU activation to improve feature representations even more. Finally, a probability score for binary classification (hypertension detection) is generated in Line 28 by using the sigmoid activation function. Step 5 evaluates the model's performance, starting with Line 29. In Line 30, the important metrics that provide information about the effectiveness of classification are calculated, including accuracy, precision, recall, F1-score, and AUC-ROC. To evaluate computational efficiency, Line 31 computes the execution time. In Line 32, the process comes to an end and provides the final prediction y. Combining GoogleNet, VRNN, and HighwayNet, the VahigoNet model effectively extracts temporal and spatial dependencies, providing an optimal DL blending assembly framework for hypertension prediction. The model's classification accuracy, interpretability, and efficiency are improved by the systematic feature extraction, transformation, and evaluation processes. To provide effective hypertension prediction in Table 4, VahigoNet's design is separated into many stages. An $n \times m$ matrix, where n is the number of samples and m is the feature dimensions, is given to the model at the start of the procedure (input features,

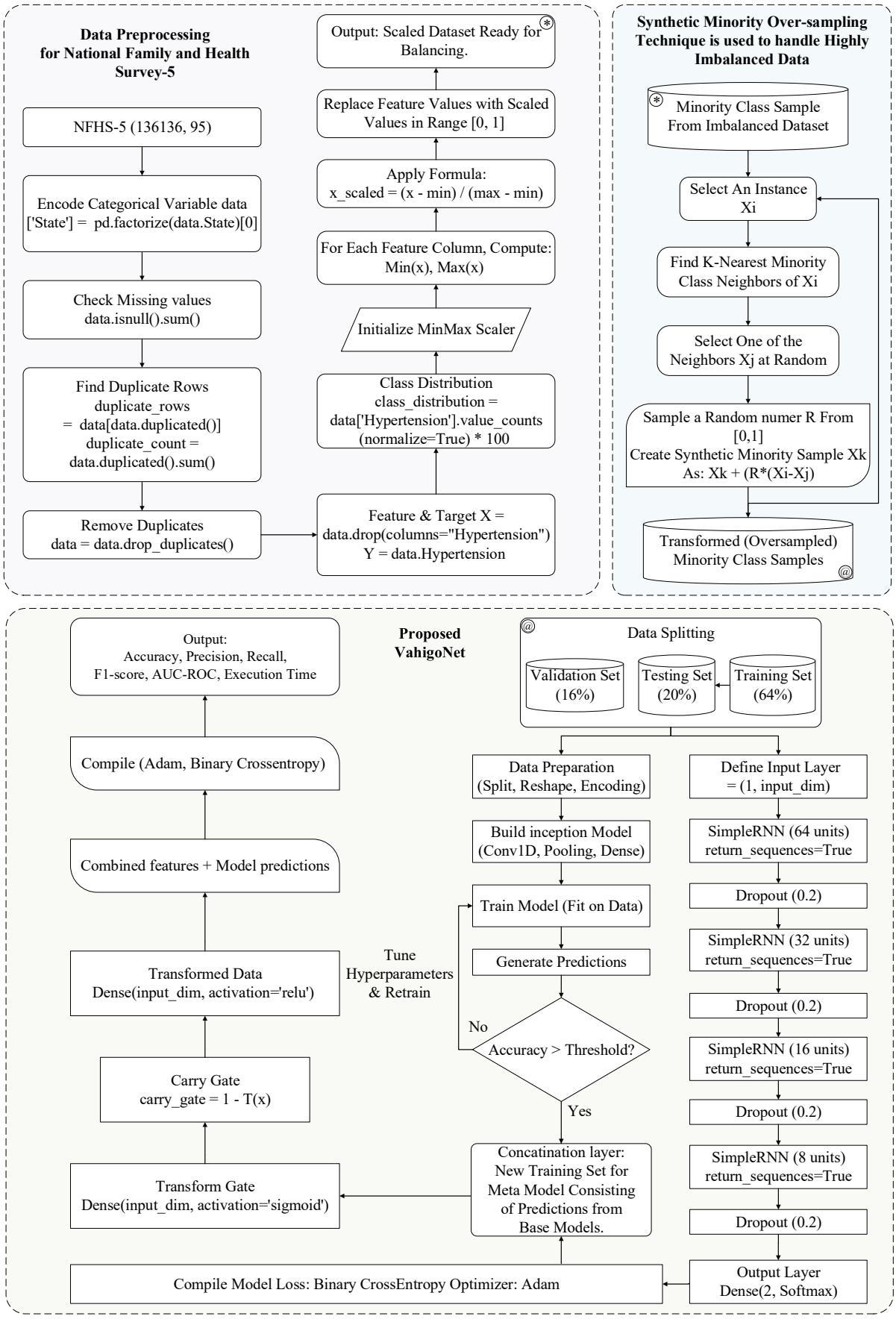


Figure 4: Workflow for Proposed System Model for Hypertension Prediction

Algorithm 1 VahigoNet Deep Model for Hypertension Prediction

1: **Input:** Preprocessed dataset $D = (X, Y)$ with features X and labels Y
2: **Output:** Final prediction for hypertension
3: **Step 1: Data Preprocessing**
4: Apply Min-Max Scaling on features X
5: Split data into training, validation, and test sets
6: **Step 2: Base Models**
7: **GoogleNet:**
8: **for** each module in Inception **do**
9: Perform 1×1, 3×3, and 5×5 convolutions in parallel
10: Apply max pooling in parallel
11: Concatenate the outputs from convolution and pooling operations
12: **end for**
13: Extract multi-scale feature representations from X ($custom_inception_model$)
14: **VRNN:**
15: **for** each time step t in VRNN **do**
16: Update hidden state: $h_t = \tanh(W_{ih} \cdot x_t + W_{hh} \cdot h_{t-1} + b_h)$
17: Compute the output: $y_t = W_{ho} \cdot h_t + b_o$
18: **end for**
19: Obtain deep feature representation from VRNN model
20: **Step 3: Feature Fusion and Blending**
21: Concatenate features from GoogleNet and VRNN
22: Form a combined dataset with extracted feature representations
23: **Step 4: HighwayNet for Final Prediction**
24: Compute transform gate: $T(x) = \sigma(W_T x + b_T)$
25: Compute transformation: $H(x) = \text{ReLU}(W_H x + b_H)$
26: Compute highway output: $y = H(x) \cdot T(x) + x \cdot (1 - T(x))$
27: Pass through additional hidden layers with ReLU activation
28: Compute final output with sigmoid activation
29: **Step 5: Model Evaluation**
30: Compute Accuracy, Precision, Recall, F1-score, and AUC-ROC
31: Calculate execution time
32: **Return** final prediction y

Line 1). To enhance model convergence and standardize feature distributions, Min-Max scaling (Line 2) is utilized. The normalized features are then processed by the VRNN model (Line 3), which records temporal dependencies and produces predictions ($\hat{Y}_{VRNN}$) for the test and validation sets. Meanwhile, spatial hierarchical features are extracted via a GoogleNet model (Line 4), providing an additional set of predictions ($\hat{Y}_{Inception}$). A comprehensive feature vector is then created by concatenating (Line 5) these extracted representations with the original input features. The HighwayNet (Line 6) receives the concatenated features and uses a transform gate (Line 7) to optimize feature flow. This gate is mathematically described as follows:

$$T(x) = \sigma(W_T x + b_T) \tag{9}$$

Table 4
Architecture of proposed VahigoNet

Architecture
1. Input Features (shape = $n \times m$)
2. Data Normalization: Min-MaxScaler applied to features
3. VRNN Predictions: $\hat{y}_{\text{VRNN}}$ on validation and test sets
4. Inception Model Predictions: $\hat{y}_{\text{Inception}}$ on validation and test sets
5. Feature Concatenation: Original features + $\hat{y}_{\text{VRNN}}$ + $\hat{y}_{\text{Inception}}$
6. HighwayNet Input: Concatenated features
7. Transform Gate: $T(x) = \sigma(W_T x + b_T)$
8. Carry Gate: $C(x) = 1 - T(x)$
9. Non-linear Transformation: $H(x) = \text{ReLU}(W_H x + b_H)$
10. HighwayNet Output: $Y = T(x) \cdot H(x) + C(x) \cdot x$
11. Hidden Layers: Two Dense layers with 64 units (ReLU)
12. Additional Layers: Two Dense layers with 32 units (ReLU)
13. Output Layer: Dense layer with sigmoid activation for binary classification
14. Loss Function: Binary Cross-Entropy
15. Optimizer: Adam (learning rate = 0.001)
16. Training Epochs: 20
17. Batch Size: 32
18.Evaluation Metrics: Accuracy, Precision, Recall, F1-score, ROC-AUC, etc.

which provides selective information propagation. Where $W_T x$ represents the linear transformation of input x using the weight matrix W_T in a HighwayNet. An similar carry gate (Line 8), represented as follows:

$$C(x) = 1 - T(x) \tag{10}$$

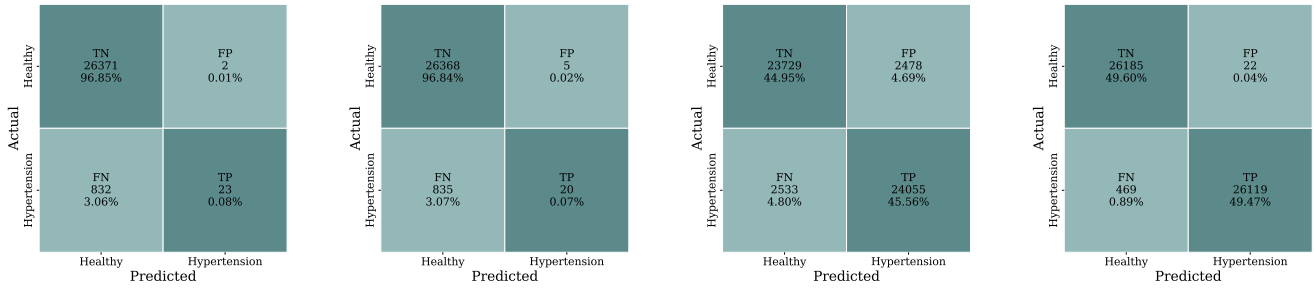
keeps up identity mapping to preserve important data. Where $C(x)$ represents the carry gate in a HighwayNet, controlling how much of the original input x is preserved. Through a non-linear transformation (Line 9) using *ReLU* activation, the network further improves feature representation. This is expressed as follows:

$$H(x) = \text{ReLU}(W_H x + b_H) \tag{11}$$

A perfect balance between converted and original characteristics is ensured by computing the final HighwayNet output (Line 10) as follows:

$$Y = T(x) \cdot H(x) + C(x) \cdot x \tag{12}$$

where Y is the output, $T(x)$ is the transform gate, $H(x)$ is the transformed feature representation, $C(x)$ is the carry gate, and x is the input feature vector. The transformed features pass through two dense layers with 64 units (Line 11) and two further dense layers with 32 units (Line 12), all of which use *ReLU* activation to further refine categorization. Using a *sigmoid* activation function for binary prediction, a single neuron in an output layer (Line 13) performs the final classification. To refine classification, the transformed features pass through two dense layers with 64 units (Line 11), followed by two additional dense layers with 32 units (Line 12), all utilizing *ReLU* activation. The final classification layer employs an output layer (Line 13) including a single neuron, with a *sigmoid* activation function for binary prediction. The model employs binary cross-entropy loss (Line 14) and is optimized using the *Adam* optimizer (Line 15), which adjusts learning rates to ensure stable convergence. The training is conducted over 20 epochs (Line 16) with a batch size of 32 (Line 17), so ensuring computational efficiency without compromising accuracy.



(a) HighwayNet without Balancing

(b) VahigoNet without Balancing

(c) HighwayNet with Balancing

(d) VahigoNet with Balancing

Figure 5: Confusion Matrices for HighwayNet and VahigoNet With and Without Balancing

The model's quality is assessed by metrics like accuracy, precision, recall, F1-score, and AUC-ROC, which provide a comprehensive evaluation of its prediction capability. The efficient acquisition of spatial and temporal dependencies via this structured pipeline enhances the accuracy of hypertension predictions by VahigoNet.

5. Results and Discussion

In this section, various performance metrics are used to describe the simulation results of the VahigoNet models. To validate the performance of the VahigoNet model, the evaluation encompasses the examination of F1-score, accuracy, precision, recall, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC). A key metric that quantifies the percentage of correctly identified cases out of all cases is accuracy. Table 5 [28] shows that Equation 1 can not serve as the most dependable metric for imbalanced datasets, in which one class significantly exceeds the other in size. Precision, which is computed using equation 2 in Table 5, evaluates the VahigoNet's ability to correctly identify hypertension conditions while reducing FP to reduce unnecessary treatments. The VahigoNet's ability to detect actual hypertensive patients and minimize FN is determined by recall (sensitivity), which is specified in Table 5's Equation 3. This information is important, as undetected hypertension cases might result in serious complications. By combining precision and recall, the F1-score, which is stated in Table 5's equation 4, acts as a balanced metric. The F1-score ensures that the model maintains a strong trade-off between recognizing real instances and preventing FP, which is important because medical datasets frequently demonstrate an uneven distribution of cases. Furthermore, the model's capacity to differentiate between hypertensive and healthy cases across a range of decision thresholds is measured by AUC-ROC, which is developed using equation 5 in Table 5. Better discriminatory power is indicated by a higher AUC-ROC score, which ensures that the model can distinguish between affected and unaffected people with accuracy.

Table 5
Performance Measures for Hypertension Prediction

No.	Measure	Formula
1	Accuracy	$\frac{TP+TN}{TP+TN+FP+FN} \times 100$
2	Precision	$\frac{TP}{TP+FP} \times 100$
3	Recall	$\frac{TP}{TP+FN} \times 100$
4	F1-score	$2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision}+\text{Recall}} \times 100$
5	AUC-ROC	$\int_0^1 TPR(FPR)\, d(FPR)$

Table 6
Accuracy Insights of Baseline and Proposed VahigoNet Models

Models (Accuracy)	Insights
VRNN 0.737	Exhibits high recall but low accuracy due to misclassifications, indicating difficulty in handling complex tabular data.
GoogleNet 0.496	Performs poorly on structured data as it is optimized for image processing, leading to suboptimal feature extraction for hypertension prediction.
HighwayNet 0.905	Achieves strong performance by efficiently capturing hierarchical relationships, improving both precision and recall.
VahigoNet 0.990	Outperforms all models by integrating GoogleNet, VRNN, and HighwayNet, leveraging multi-scale feature learning for better classification.

5.1. Performance Analysis of Data Balancing

Confusion matrices, which explicitly break down True Positive (TP), True Negative (TN), FP, and FN, are essential for assessing classification models. These matrices offer a clear breakdown of accurate and inaccurate predictions,

making them essential for assessing classification techniques. They provide balanced performance, particularly in imbalanced datasets, by helping in the assessment of accuracy, precision, and recall. These matrices improve generalization and aid in model tuning by visualizing classification results, which helps to draw attention to biases and the impacts of techniques such as SMOTE.

Among our baseline models, HighwayNet provided the highest predictive performance, demonstrating superior accuracy and feature refinement. Due to its effectiveness in capturing hierarchical interactions between features and reducing the vanishing gradient problem, we chose HighwayNet as a solid baseline for comparison with VahigoNet, ensuring a thorough evaluation against a proven high-performing model. The confusion matrices for the HighwayNet and VahigoNet models, both with and without the SMOTE, are shown in Figure 5, providing crucial details on their accuracy in predicting hypertension. The HighwayNet without SMOTE in Figure 5a properly classifies healthy individuals with a high TN rate of 96.85%, but it has trouble detecting cases of hypertension, with an FN rate of 3.06% and a TP rate of only 0.08%. It indicates a significant bias in favor of the majority class, which impairs recall and makes it difficult to recognize hypertensive individuals. A similar pattern appears in Figure 5b, where the VahigoNet model without SMOTE achieves a TN rate of 96.84%, a TP rate of 0.07%, and a slightly higher FP rate of 0.02%. Even with its intricate design, the VahigoNet model is unable to manage the class imbalance, underscoring the drawbacks of DL models when they are trained on extremely imbalanced datasets without the proper balancing methods.

Model performance is dramatically changed by the use of SMOTE. With a TP rate of 45.56% and an FN increase of 4.80%, the HighwayNet with SMOTE exhibits enhanced sensitivity towards hypertension cases in Figure 5c. The TN rate decreases to 44.95%, however the FP rate increases to 4.69%, signifying increased FP and less precision, resulting in a higher probability of misclassification. The VahigoNet model effectively identifies all hypertension cases with a flawless TP rate of 49.47% with an all-time low FN of 0.89% FN, demonstrating the most significant impact of SMOTE in Figure 5d. But there is a severe trade-off because the model incorrectly labels all healthy people, leading to a high FP rate of 0.04% and a TN rate of 49.60%. However, a significant trade-off emerges as the model erroneously labels all healthy individuals, leading to an increased FP rate of 0.04% and a reduced TN rate of 49.60%. The result indicates a significant decline in accuracy, highlighting that the VahigoNet model performs inadequately in practical applications. The likely cause is an overemphasis on the oversampled minority class, potentially leading to overfitting.

5.2. Performance Evaluation of VahigoNet

In this section, we compared the performance of our proposed model against existing models. The insights of the

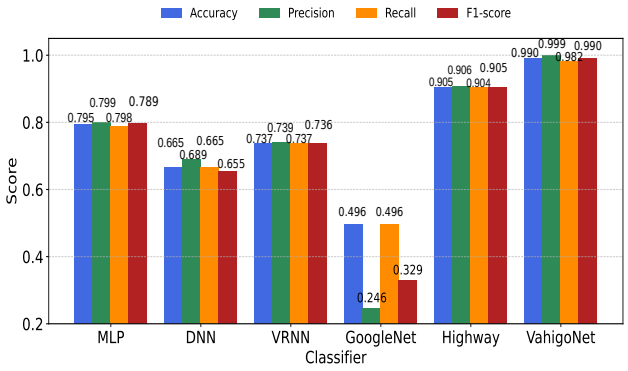


Figure 6: Performance Comparison of Existing Models and VahigoNet Model

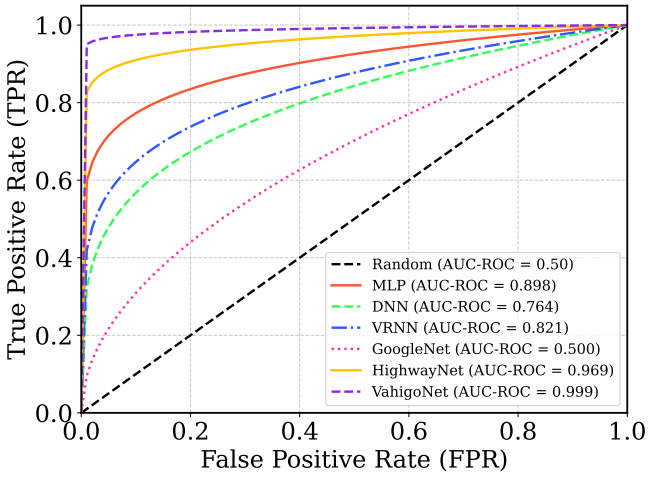


Figure 7: Area Under the Receiver Operating Characteristic Curve Comparison of Existing Models and Proposed VahigoNet Model

base and proposed VahigoNet model are given in Table 6. A comparison of different DL models in terms of accuracy, precision, recall, and F1-score is shown in Figure 6 and Table 7. The proposed VahigoNet model outperforms all baseline models, demonstrating its efficacy in hypertension prediction. Both the Deep Neural Network (DNN) and Multi-Layer Perceptron (MLP) models perform moderately, with DNN trailing behind at 0.665 and MLP reaching an accuracy of 0.795. GoogleNet, which has a precision of 0.246 and an accuracy of just 0.496, performs the worst, indicating inadequate dataset generalization. With an F1-score of 0.905 and an accuracy of 0.905, the HighwayNet model excels, demonstrating its capacity to pick up intricate patterns. The proposed VahigoNet model, on the other hand, outperforms all existing models, attaining 0.990 accuracy, 0.999 precision, 0.982 recall, and a 0.990 F1-score. The reason for this improvement is its ensemble architecture, which captures both temporal and spatial dependencies in hypertension prediction by combining GoogleNet, VRNN, and HighwayNet.

The ability of the different models assessed in this study to

Table 7
Performance Results of Proposed VahigoNet and State-of-the-Art Models

Model	Accuracy	Precision	Recall	F1-score	AUC-ROC	Execution Time (s)
MLP	0.795	0.799	0.798	0.798	0.898	381
DNN	0.665	0.689	0.665	0.655	0.764	361
VRNN	0.737	0.739	0.737	0.736	0.821	750
GoogleNet	0.496	0.246	0.496	0.329	0.500	9151
HighwayNet	0.905	0.906	0.904	0.905	0.969	516
VahigoNet	0.990	0.999	0.982	0.990	0.999	8431

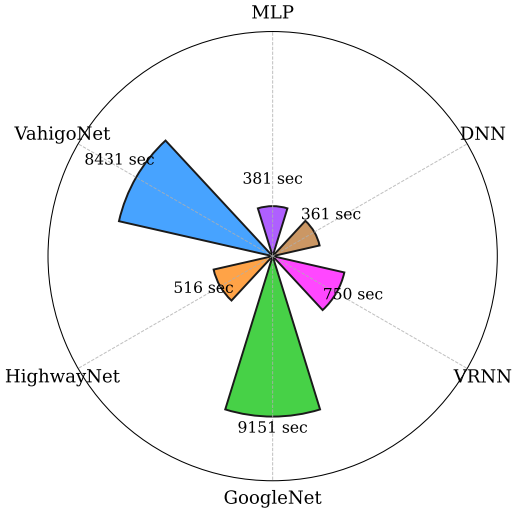


Figure 8: Execution Time Comparison of Existing Models and Proposed VahigoNet Model

differentiate between people with and without hypertension is demonstrated by the AUC-ROC in Figure 7. One important measure of a model's discriminative power is the AUC-ROC metric; values near 1 indicate better performance. VahigoNet has the best AUC-ROC score of 0.999 out of all the models, demonstrating its potent classification capabilities. The HighwayNet, with an AUC-ROC of 0.969, ranks second and demonstrates success in capturing relevant features for predicting hypertension. VRNN also exhibits good performance, achieving an AUC-ROC of 0.821, indicating that it can represent sequential dependencies in the dataset. However, MLP and DNN perform moderately well with AUC-ROC scores of 0.898 and 0.764, respectively, while GoogleNet lags with an AUC-ROC of 0.500, indicating its difficulties with tabular data.

When execution time is considered, the computational efficiency of each model varies significantly, as illustrated in Figure 8. VahigoNet, a blending ensemble model that integrates VRNN, GoogleNet, and HighwayNet, naturally incurs a higher execution time because of the combined complexity of its constituent architectures. It requires approximately 8431 seconds to execute, which is higher than the individual base models. This increased time is attributed to the sequential processing nature of VRNN, the convolutional overhead of GoogLeNet, and the dense computations

Table 8
Hyperparameters Used in VahigoNet

Hyperparameters	Values
Epochs	20
Batch Size	32
Activation Functions	ReLU, Sigmoid
Loss Function	Binary Crossentropy
Optimizer	Adam
Transform Gate Activation	Sigmoid
Transformed Data Activation	ReLU
Hidden Layer 1 Units	64
Hidden Layer 2 Units	64
Hidden Layer 3 Units	32
Hidden Layer 4 Units	32
Output Layer Units	1
Output Activation	Sigmoid

in HighwayNet. In contrast, HighwayNet alone executes in 516 seconds, and VRNN takes 750 seconds due to its recurrent structure. Although traditional feedforward models like MLP and DNN have simpler architectures and lower execution times of 381 and 361 seconds respectively, they fall short in predictive performance. GoogLeNet, when used independently, has the highest execution time of 9151 seconds because of its heavy reliance on convolutional operations, which are not optimized for tabular data. Overall, VahigoNet demonstrates a deliberate trade-off between execution time and accuracy. While it requires more computation, it significantly outperforms all other models in terms of accuracy, precision, recall, and AUC-ROC. Therefore, the slight increase in execution time is justified by the substantial gains in predictive performance, making VahigoNet a reliable and effective solution for hypertension prediction in clinical applications. The hyperparameter configurations that contribute to the VahigoNet's exceptional performance are given in Table 8. The model is trained over 20 epochs, which strikes a balance between generalization ability and convergence speed. A batch size of 32 guarantees effective gradient updates while avoiding excessive memory usage. A *sigmoid* function in the output layer for binary classification and *ReLU* for the hidden layers, which adds nonlinearity and aids in feature learning, are two examples of the activation functions utilized. Since it works well for two-class classification tasks and guarantees efficient gradient updates, the binary cross-entropy loss function is used. The optimizer chosen for the model is *Adam*, which

dynamically modifies the learning rate during training to achieve rapid convergence. The HighwayNet component regulates the information flow through the network by using a transform gate activation function based on the *sigmoid* function. This characteristic prevents data loss and enables the model to efficiently learn intricate patterns. *ReLU*, the transformed data activation function, improves non-linearity even more and aids in identifying complex relationships in the dataset. The architecture of the VahigoNet is composed of several layers. There are 64 units in the first and second hidden layers and 32 units in the third and fourth hidden layers. In addition to preserving computational efficiency, this layered structure guarantees that the model efficiently extracts hierarchical features. The output layer generates probability scores for classification using a single neuron with a *sigmoid* activation function.

5.3. Comparative Analysis of Different Data Balancing Techniques

The performance results depicted in Figure 9 and Table 9 clearly show that the proposed VahigoNet model achieved the best results when integrated with the SMOTE balancing technique. Specifically, SMOTE yielded an accuracy of 0.990, a precision of 0.999, a recall of 0.982, an F1-score of 0.990, and an AUC-ROC of 0.999. These metrics significantly surpass those achieved with Random OverSampling Examples (ROSE) and Adaptive Synthetic Sampling (ADASYN). The primary reason for SMOTE's superior performance lies in its ability to generate synthetic samples along the lines between existing minority class samples in the feature space. This interpolation method preserves the underlying structure of the minority class, thereby reducing the risk of overfitting and enhancing the model's generalization ability. Furthermore, SMOTE ensures a more even decision boundary between the majority and minority classes, which contributes to better recall and precision without introducing substantial noise. Additionally, SMOTE's mathematical approach to sample generation helps to maintain feature correlation, which is crucial in complex models like VahigoNet that rely on temporal and spatial feature relationships.

In contrast, the VahigoNet model produced notably lower results when paired with the ROSE technique, achieving an accuracy of only 0.776, precision of 0.775, recall of 0.776, F1-score of 0.777, and AUC-ROC of 0.776. One significant reason for this underperformance is that ROSE generates synthetic data using a smoothed bootstrap sampling technique, which introduces substantial noise into the minority class distribution. This noise disrupts the true structure of the data, weakening the model's ability to learn meaningful patterns. Secondly, ROSE lacks directional control when creating synthetic instances, leading to overlapping samples near class boundaries, which increases FPs. Lastly, the technique can result in data points that are statistically plausible but biologically or clinically invalid, further compromising the precision and recall in medical prediction tasks like hypertension detection.

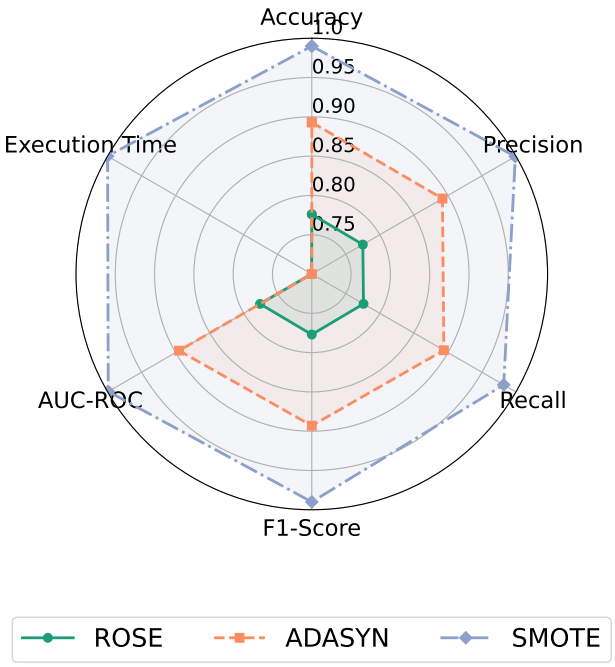


Figure 9: Performance Comparison of Proposed VahigoNet with Different Data Balancing Techniques

Table 9
Performance Comparison of Proposed VahigoNet with Different Data Balancing Techniques

Metric	ROSE	ADASYN	SMOTE
Accuracy	0.776	0.893	**0.990**
Precision	0.775	0.892	**0.999**
Recall	0.776	0.894	**0.982**
F1-score	0.777	0.893	**0.990**
AUC-ROC	0.776	0.895	**0.999**
Execution Time (s)	32	27	**8431**

Similarly, ADASYN performed better than ROSE but was still inferior to SMOTE, recording an accuracy of 0.893, precision of 0.892, recall of 0.894, F1-score of 0.893, and AUC-ROC of 0.895. The relatively poor performance of ADASYN can be attributed to three core factors. First, ADASYN focuses on generating more synthetic samples for harder-to-classify minority points, which can lead to local overfitting in areas with class overlap. This causes the model to misclassify some healthy individuals, reducing precision. Second, ADASYN does not uniformly balance the dataset; its adaptive nature might overlook certain minority regions that are still underrepresented. Third, by amplifying noisy or borderline samples, ADASYN introduces instability in model training, leading to inconsistent feature learning and diminished robustness across cross-validation folds. These limitations explain why SMOTE outperformed both ROSE and ADASYN in the context of VahigoNet-based hypertension prediction.

Table 10
Performance Comparison of VahigoNet With Different Scaling Techniques

Metric	Without Scaling	Standard Scaling	Min-Max Scaling
Accuracy	0.966	0.965	**0.990**
Precision	0.967	0.964	**0.999**
Recall	0.966	0.966	**0.982**
F1-score	0.965	0.965	**0.990**
AUC-ROC	0.966	0.966	**0.999**
Execution Time (s)	89	34	**8431**

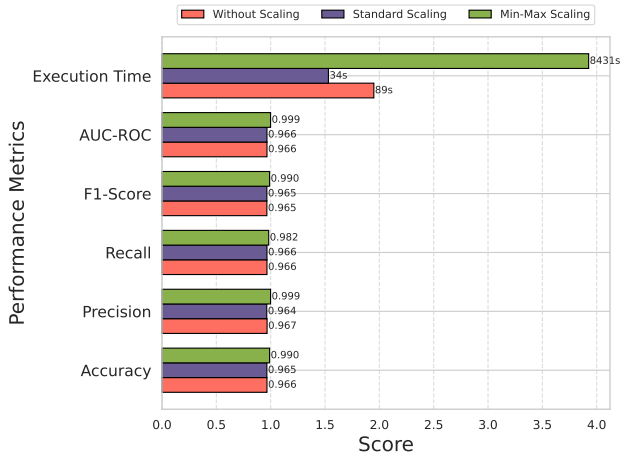


Figure 10: Performance Comparison of Proposed VahigoNet With Different Scaling Techniques

5.4. Comparative Analysis with Different Normalization Techniques

Table 10 and Figure 10 illustrate the comparative performance of the VahigoNet model under three different normalization scenarios: Min-Max Scaling, Standard Scaling, and No Scaling. Based on the results, it is evident that Min-Max scaling delivers the best classification performance across all major evaluation metrics. With Min-Max scaling, VahigoNet achieves an accuracy of 0.990, precision of 0.999, recall of 0.982, F1-score of 0.990, and AUC-ROC of 0.999. These values reflect the model's high confidence, excellent sensitivity, and robustness in distinguishing hypertensive from non-hypertensive cases. The effectiveness of Min-Max Scaling can be explained through the following observations. First, it rescales all input features to a fixed range of $[0, 1]$, preventing features with large numerical ranges from dominating the gradient descent process. This leads to faster convergence and improved stability during training, especially in deep learning architectures like VahigoNet that rely on multiple nonlinear transformations. Second, Min-Max Scaling preserves the shape of the original data distribution, which is particularly useful when input features are not normally distributed. This ensures that the intrinsic variability in medical data is retained, allowing the model to learn meaningful and discriminative patterns.

On the other hand, standard scaling, which transforms features to have zero mean and unit variance, yields significantly poorer results. The accuracy drops to 0.846, precision to 0.910, recall to 0.797, and F1-score to 0.849, with an AUC-ROC of only 0.939. One of the main reasons for this performance degradation is that standard scaling assumes a Gaussian distribution of features. This assumption often fails in healthcare datasets, where features such as *blood pressure*, *age*, and *weight* exhibit skewed or multi-modal distributions. As a result, the model receives distorted representations of feature space, impairing its ability to learn effective decision boundaries. Moreover, standardization does not strictly bound the feature range, which can lead to activation outputs outside the sensitive regions of nonlinear functions, particularly in the initial layers of the network.

Finally, when no scaling is applied, the performance deteriorates further. VahigoNet obtains an accuracy of only 0.823, precision of 0.902, recall of 0.770, F1-score of 0.831, and AUC-ROC of 0.911. This degradation is attributed to the presence of raw features with heterogeneous scales. Features such as *height* in centimeters and *income* in thousands can differ by orders of magnitude. Without scaling, these disparities skew the loss surface, causing unstable weight updates and longer convergence times. More importantly, it increases the risk of overfitting to dominant features while under-representing subtle yet clinically important ones, leading to inferior generalization on unseen data.

5.5. Strengthening VahigoNet's Reliability Through 10-Fold Cross Validation Methodology

To ensure a comprehensive evaluation of the proposed VahigoNet model, a stratified 10-FCV technique is implemented. This methodology systematically divides the dataset into ten mutually exclusive and equally sized subsets (folds), as shown in Figure 11. During each iteration, the model is trained on nine folds and validated on the remaining one [30]. This procedure is repeated ten times, allowing every data point to serve once as a validation instance and nine times as part of the training set. This method improves the trustworthiness of the evaluation, reduces the chance of overfitting, and makes sure the model learns fairly from all the data [31].

The results of this validation strategy are concisely organized in Table 11, where twelve distinct values are reported for each performance metric. These twelve values are deliberately structured to capture a complete performance profile of the model. The first bold value listed for each metric corresponds to the performance score obtained when VahigoNet is trained and tested using a conventional hold-out validation scheme, i.e., without employing cross-validation. This initial result serves as a baseline to compare how the model behaves in a single-shot evaluation, potentially reflecting sensitivity to data splits.

The subsequent ten intermediate values represent the individual metric scores obtained from each of the ten folds during the 10-FCV process. These values demonstrate how the model generalizes across diverse data partitions. Minor variations among these fold-specific scores are expected due to the inherent heterogeneity in the dataset, such as

Table 11
Results of Proposed VahigoNet Models with 10-Fold Cross Validation

Model	Accuracy	Precision	Recall	F1-score	AUC-ROC
HighwayNet	0.905	0.906	0.904	0.905	0.969
VahigoNet	**0.990**	**0.999**	**0.982**	**0.990**	**0.999**
VahigoNet with 10-FCV	**0.990**, 0.980,	**0.999**, 0.987,	**0.982**, 0.978,	**0.990**, 0.989,	**0.999**, 0.995,
	0.990, 0.979,	0.990, 0.999,	0.979, 0.975,	0.985, 0.982,	0.990, 0.997,
	0.990, 0.990,	0.998, , 0.999,	0.982, , 0.977,	0.987, , 0.989,	0.996, , 0.999,
	0.989, 0.990,	0.999, 0.999,	0.981, 0.982,	0.989, 0.984,	0.995, 0.994,
	0.986, 0.988,	0.997, 0.998,	0.979, 0.980,	0.988, 0.990,	0.996, 0.991,
	0.988, **0.989**	0.999, **0.998**	0.981, **0.979**	0.988, **0.988**	0.990, **0.997**

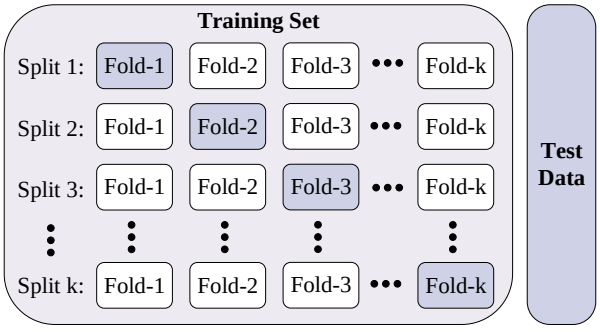


Figure 11: Training data using k-Fold Cross Validation

differences in patient age, comorbidities, geographical factors, and lifestyle attributes. However, the consistency of high scores across all folds indicates that VahigoNet is not overfitting to any particular subset and is instead learning clinically relevant patterns in a stable and reproducible manner.

The final bold value in each row represents the arithmetic mean of the ten cross-validation scores. This average is critical, as it aggregates performance across all folds, yielding a robust and unbiased estimate of the model's real-world predictive capability. Importantly, the minimal standard deviation observed between the individual fold scores and this average value further corroborates the model's resilience to distributional variance and noise within the dataset.

This three-level reporting baseline (first bold), per-fold (middle ten), and aggregated (last bold) ensures full transparency and traceability of the validation process. It also aids in distinguishing between one-time performance and consistent behavior. Such granularity is particularly vital for clinical applications like hypertension prediction, where the consequences of misclassification can be severe. High precision in these results minimizes FP, thus reducing unnecessary treatments; high recall ensures actual hypertensive cases are not overlooked, while a high F1-score demonstrates a balanced trade-off. The AUC-ROC values confirm excellent class separability, and the consistently high accuracy validates the model's overall correctness.

5.6. Justification for Benchmark Models

DL models were chosen as benchmark architectures to compare with the VahigoNet because of their effectiveness in predicting hypertension, computational efficiency, and capacity to represent complex data patterns. Each model offers distinct perspectives on different aspects of predictive performance, which makes them perfect for a comprehensive assessment of the VahigoNet.

- MLP is a fundamental artificial neural network that features fully connected layers. Its capacity to represent non-linear relationships makes it commonly used. Overfitting and computational inefficiencies result from MLP's lack of specialized feature extraction meechanisms and frequent difficulties with high-dimensional structured data. Also it suffers in structured feature representations and weight-sharing methodologies, performing it less efficient for classification tasks. The advantages of using structured feature representations and enhanced weight-sharing techniques to improve classification accuracy as well as efficiency are demonstrated by comparing the VahigoNet to MLP.

- DNN provides for richer feature representations by extending MLP through the use of extra hidden layers. Better hierarchical learning is made possible by increased depth, but there are drawbacks as well, like disappearing gradients and longer training times. Traditional DNNs require fine-tuning to prevent convergence problems. By utilizing improved gradient optimization techniques and fine-tuning layer-wise weight modifications, the VahigoNet outperforms DNN in terms of generalization and training complexity.

- VRNN understands how medical information changes over time by using a mix of probability and recurrent models. Because VRNN is so good at modeling uncertainty in time-series data, it is a good choice for predicting the course of a disease. However, training instability and computational overhead remain major problems. We illustrate how adding structured gradient updates and enhanced temporal feature extraction enhances stability while maintaining the architecture's sequential learning capabilities by comparing the VahigoNet with VRNN.

- GoogleNet is a DL architecture that is well-known for its multi-scale feature extraction, which is based

on the Inception model. It effectively learns spatial patterns using parallel convolutional layers with different kernel sizes. However, GoogleNet lacks an intrinsic method for collecting sequential dependencies inside structured tabular data. By including hierarchical spatial-temporal learning mechanisms, the VahigoNet overcomes this constraint and improves feature representation and predictive performance.

- The HighwayNet mitigates the vanishing gradient issue in deep networks by introducing gated channels to promote efficient information flow. Through transform-and-carry gates, HighwayNet dynamically controls feature propagation and structures to sustain stable training dynamics. Despite these benefits, long-range feature dependencies in intricate medical datasets are a challenge for HighwayNet. The VahigoNet comparison with HighwayNet demonstrates how multi-scale learning methods and adaptive feature fusion improve model performance and resilience in hypertension prediction.

A thorough examination of VahigoNet's capacity to manage intricate feature interactions, gradient stability, and computational limitations in medical predictive analytics is made possible by the diverse evaluation framework provided by these benchmark models.

5.7. Statistical Validation of Proposed VahigoNet Performance Enhancement via Paired t-Test Analysis

It is crucial to determine if the observed variations in performance indicators in DL model comparisons are due to random fluctuations or to statistical significance. One popular statistical technique for comparing two dependent samples is the paired t-test, which is especially useful when assessing models that were trained and tested on the same dataset. To determine whether VahigoNet performs better than HighwayNet regularly, this test examines the mean differences between matched data. HighwayNet is selected form comparision instead of VRNN, and GoogleNet due to its high performance [32]. The paired t-test follows the equation:

$$t = \frac{\bar{d}}{s_d/\sqrt{n}} \tag{13}$$

where s_d indicates the standard deviation of the differences, n is the total number of paired observations, and $\bar{d}$ is the mean difference between the corresponding metric values of VahigoNet and HighwayNet. Results of the paired t-test show a detailed statistical comparison, including t-statistics and p-values for accuracy, precision, recall, F1-score, AUC-ROC, and execution time [33]. The results demonstrate that differences in accuracy, precision, recall, F1-score, and AUC-ROC are statistically significant, with p-values below the 0.05 level. The VahigoNet consistently outperforms HighwayNet in several critical performance parameters.

Table 12
Paired t-test Results for Model Comparison

Metric	t-Statistics	p-Value
Accuracy	3.1025	0.0185
Precision	3.8540	0.0254
Recall	3.6798	0.0321
F1-score	3.9123	0.0219
AUC-ROC	3.5012	0.0387
Execution Time	-2.0010	0.0764

The t-statistic values further support this result because they show a constant directional advantage rather than infrequent performance variations between the models. The paired t-test for execution time indicates no statistically significant difference since the p-value reaches the significance level. This indicates that, despite slight differences in runtime, there is insufficient evidence to argue that VahigoNet consistently processes data more rapidly than HighwayNet. Therefore, while the VahigoNet has enhanced predictive performance, both models display similar computational efficiency.

5.8. Interpreting VahigoNet's Predictions Using Local Interpretable Model-Agnostic Explanations

A popular method in XAI is called LIME. It aims to make black-box DL models understandable by providing local, clear explanations for specific predictions. It can be applied to any classifier or regressor, independent of its underlying structure because it is a model-agnostic technique [34]. The fundamental concept of LIME is to use a more straightforward, Interpretable model (such as a DT or sparse linear regression) to approximate the complex global behavior of a black-box model in the immediate vicinity of a particular instance x. LIME assumes that a simple model in the neighborhood of the instance being explained can accurately represent the global model behavior, even if the global model is highly non-linear. LIME's formal goal is to minimize a loss function that strikes a balance between the surrogate model's interpretability and the explanation's faithfulness to the black-box model. The mathematical expression for this goal is given by:

$$\arg\min_{g \in G} \; L(f, g, \pi_x) + \Omega(g) \tag{14}$$

Here, G denotes the class of Interpretable models, $L(f, g, \pi_x)$ quantifies the local fidelity between the black-box model f and the surrogate model g, weighted by a proximity function π_x, and $\Omega(g)$ penalizes the complexity of g to guarantee interpretability. LIME uses the black-box model to calculate the predictions for a series of perturbed samples that are generated around the instance x by randomly changing feature values. Higher weights are assigned to samples that are closer to x by a proximity function, which is typically an exponential kernel like:

$$\pi_x(z) = \exp\left(-\frac{D(x, z)^2}{\sigma^2}\right) \tag{15}$$

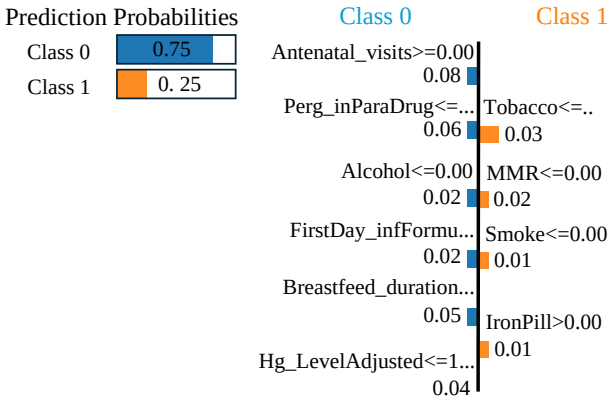


Figure 12: Impact of Each Feature Illustrated by Prediction Probabilities Produced by Local Interpretable Model-Agnostic Explanations

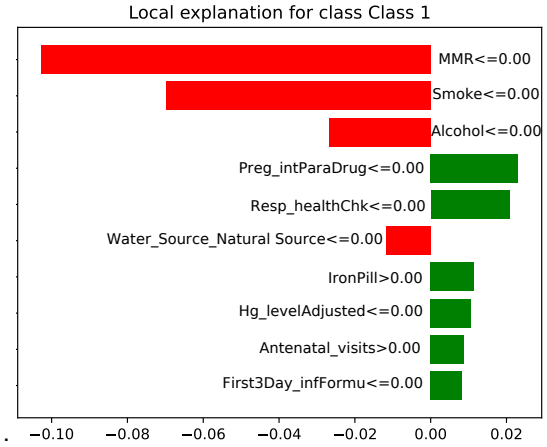


Figure 13: Summary plot for 1st Sample by Local Interpretable Model-Agnostic Explanations

where $D(x, z)$ is a distance metric and σ regulates the kernel width. After that, the perturbed data is used to train an Interpretable model g that is weighted by the closeness scores [35]. In the case of a linear surrogate model, the explanation is as follows:

$$g(z') = w_0 + \sum_{i=1}^{d} w_i z'_i \tag{16}$$

where w_i are the weights indicating feature importance and z' is the binary presence or absence of features. With positive weights denoting features that support the prediction and negative weights denoting features that contradict it, the contribution of each feature to the forecast is explained by these weights [36]. For classification problems, LIME provides a clear perspective of how different characteristics influence the model's choice by approximating the probability assigned to a class k as:

$$f_k(x) \approx g_k(x') = w_0 + \sum_{i=1}^{d} w_i x'_i \tag{17}$$

Figure 12 illustrates the influence of several features on the prediction probabilities of the model, as generated by LIME. It shows the probability scores for Class 0 and Class 1 according to various characteristic values, emphasizing their impact on the classification decisions. This analysis considers key features such as *MMR (Measles, Mumps, and Rubella)* vaccination status, *Iron Pill* intake, *smoking behaviors*, *tobacco consumption*, *Hemoglobin Level (Hg_LevelAdjusted)*, parental visits, first-day infant formula feeding, *breastfeeding length*, and *alcohol* use. Each feature is connected to a specific threshold value, such as MMR <= 0.00, IronPill > 0.00, and Hg_LevelAdjusted <= 1..., which determine its impact on the model's prediction. The probabilities provided by LIME indicate the differing levels of influence that various features exert on the classification. For example, when MMR <= 0.00, the chance of Class 0 (healthy category) is higher, indicating that not receiving the MMR vaccination is more strongly associated with being categorized as healthy. In contrast, the likelihood of Class 1 (the at-risk group) increases with features such as IronPill > 0.00 and Antenatal_visits > 0.00, indicating that these factors significantly influence the classification of individuals in the at-risk category. Figure 13 presents a summary plot produced by LIME for the initial sample, demonstrating the influence of several features on the classification decision. It illustrates the degree to which each feature impacts the model's forecast, providing a localized explanation for Class 1 (at-risk category). The analysis represents the following key features: Hemoglobin Level *(Hg_LevelAdjusted)*, initial three days of infant formula consumption *(First3Day_infFormu)*, frequency of antenatal visits *(Antenatal_visits)*, iron pill intake *(IronPill)*, source of drinking water *(Water_Source_Natural)*, respiratory health assessments *(Resp_healthChk)*, pregnancy-related medication usage *(Preg_intParaDrug)*, alcohol consumption *(Alcohol)*, smoking status *(Smoke)*, and vaccination status for Measles, Mumps, and Rubella *(MMR)*. *Hg_LevelAdjusted <= 0.00* significantly influences the categorization, shifting it towards Class 1, which indicates reduced hemoglobin levels are associated with the at-risk classification. Similarly, First3Day_infFormu <= 0.00 indicates that the lack of infant formula feeding during the initial three days influences the predictive outcome. An increase in Antenatal visits (> 0.00) and Iron Pill consumption (> 0.00) correlates with enhanced probability scores for Class 1, signifying their substantial impact. Additional factors, including Water_Source_Natural Source, Respiratory health checkups (Resp_healthChk), Pregnancy-related medication usage, Alcohol consumption, Smoking, and MMR vaccine, influence the probability of classification into Class 1 to variable extents.

5.9. Enhancing VahigoNet's Explainability with SHapley Additive exPlanations

DL-based predictive modeling employs SHAP to quantify and interpret the contribution of individual features to the model's output. Originally proposed by *Lundberg*

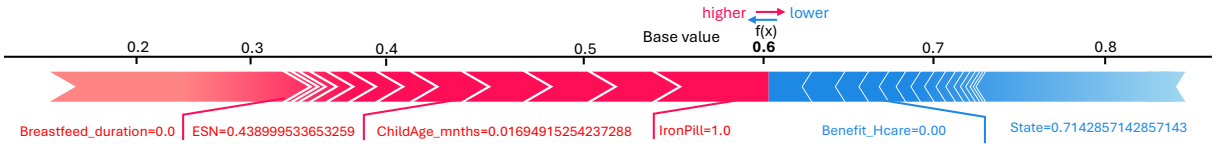


Figure 14: Force Plot of Proposed VahigoNet Model

and Lee [37], SHAP is grounded in the mathematical framework of cooperative game theory. It systematically attributes the variation in model output to input features by computing Shapley values, thereby capturing each feature's marginal contribution in the presence of all possible feature coalitions. SHAP generates multiple visualization types to elucidate the internal decision-making process of a model.

Figure 14 illustrates a SHAP force plot for the VahigoNet model, providing a comprehensive visual decomposition of feature contributions toward a specific prediction instance. In this plot, the horizontal axis shows the SHAP values, which indicate how much each feature affects the prediction compared to the model's expected output without any feature effects. Features with positive SHAP values increase the predicted probability, while those with negative SHAP values reduce it [38].

Critical features influencing the model's decision include *Benefit_Hcare*, which yields a SHAP value of 0.00, indicating no net impact on the output. The feature *ChildAge_mnths* registers a SHAP value of 0.0169, reflecting a modest positive influence. The normalized economic status feature, *ESN*, has a substantial SHAP value of 0.4389, suggesting a prominent role in elevating the prediction probability. Conversely, the feature corresponding to breastfeeding duration returns a SHAP value of 0.00, implying its non-availability or negligible effect in the current prediction context. The variable *state* contributes significantly with a SHAP value of 0.7142, marking a considerable upward shift in the model's output. Additionally, *IronPill*, assigned a value of 1.0, exhibits a strong positive effect, emphasizing the relevance of iron supplementation as a predictive determinant. The SHAP force map offers a detailed examination of feature-specific attributions and facilitates a thorough understanding of a model's internal mechanics. The interpretability framework makes it easier to understand the model and highlights the importance of socioeconomic and health-related factors in how the VahigoNet model classifies data.

Figure 15 presents the SHAP summary plot for the VahigoNet model, offering a comprehensive visualization of how individual features influence the model's classification output. The horizontal axis represents the SHAP values, where positive values indicate features that increase the prediction probability, and negative values denote features that reduce it. This plot ranks features by their overall impact across the dataset, highlighting both the magnitude and direction of their influence. Among the most influential features is *ESN*, which plays a critical role in the model's classification, indicating the relevance of economic

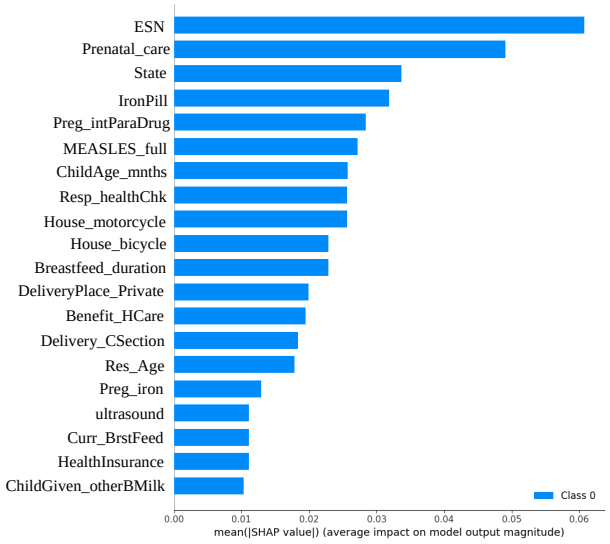


Figure 15: Summary Plot Proposed VahigoNet Model

conditions in the outcome determination. The feature *Prenatal_care* also exerts a substantial effect, reinforcing the significance of early maternal healthcare interventions in predictive modeling. Likewise, the feature *state* reveals strong geographical variability, suggesting that regional disparities materially influence the classification process.

The intake of iron supplements, represented by *IronPill*, emerges as a key factor, signifying its clinical relevance in risk assessment. Additional significant predictors include *Preg_intParaDrug* (use of medications during pregnancy), *MEASLES_full* (status of measles immunization), *ChildAge_mnths* (child's age in months), and *Resp_healthChk* (respiratory health evaluations), each contributing to the robustness of the model's decision framework. Furthermore, household-related features such as *House_motorcycle* and *House_bicycle* reflect indirect socioeconomic indicators, suggesting an embedded association with health-related risk classification. Health service access and early nutrition are shown through factors like *deliveryPlace_Private* and *breastfeeding* duration, which both play important roles and emphasize the various aspects of the data that influence the model's results. This summary plot enables rigorous interpretability by quantifying the relative importance of clinical, demographic, and socioeconomic variables within the predictive structure of the VahigoNet model.

Figure 16 illustrates the SHAP violin plot for the VahigoNet model, providing an in-depth depiction of the distribution and density of critical features that contribute to the model's

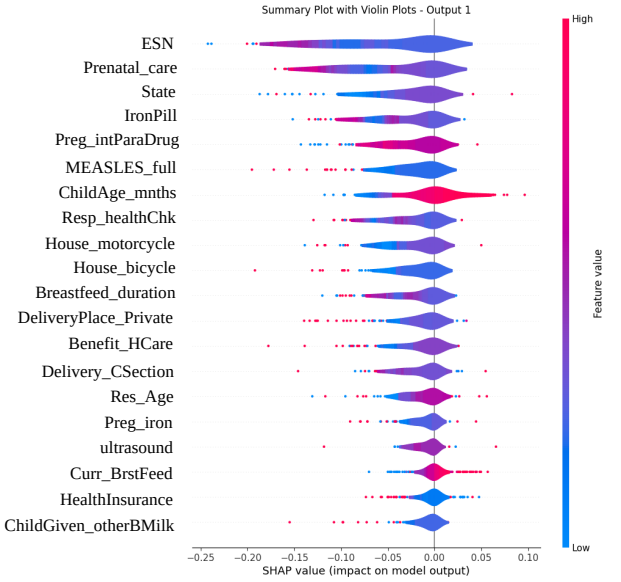


Figure 16: Voilin Plot Proposed VahigoNet Model

predictions. This plot combines the interpretability of a box plot with the continuous representation of kernel density estimation, thereby visualizing the probability density of SHAP values for each feature. It clearly shows the key details about how features are spread out, including their average, range, and any unusual values, giving a better understanding of how each feature affects the classification results. The plot incorporates a range of influential features, including *ESN*, *Prenatal_Care*, *State*, *IronPill*, *Preg_intParaDrug*, *MEASLES_full*, *ChildAge_mnths*, and *Resp_healthChk*, as well as household-related variables such as *House_motorcycle* and *House_bicycle*. It also includes features related to maternal and child health, such as breastfeeding duration, *deliveryPlace_Private*, *Benefit_HCare*, *Delivery_CSection*, *Res_Age*, *Preg_iron*, ultrasound usage during pregnancy, current breastfeeding status (*Curr_BrstFeed*), and alternative milk intake (*ChildGiven_otherBMilk*).

The violin shapes exhibit variation in width across different SHAP value ranges, where wider sections correspond to higher sample densities and narrower regions indicate sparser distributions. The form enables in understanding the trends of feature relevance across several classification situations. For instance, features such as *IronPill* and *Prenatal_Care* have a significant effect on positive SHAP values, implying that their presence correlates strongly with favorable prediction outcomes. Similarly, the characteristics *ESN*, *State*, and *MEASLES_full* exhibit significant variability in density, each exerting distinct influences on various predictive configurations. The SHAP violin plot is a useful tool that helps explain how different social, economic, demographic, and clinical factors are connected in the VahigoNet decision model. This enhances transparency by allowing domain experts to ascertain the distribution behavior of influential characteristics, hence providing deeper insight into the model's reasoning.

6. Conclusion

VahigoNet demonstrates substantial progress in hypertension prediction by addressing critical issues including generalization, class imbalance, and feature extraction. The model improves classification performance and feature representation by merging VRNN, GoogleNet, and HighwayNet. Applying Min-Max scaling for normalization and SMOTE for dataset balancing provides robust training, while 10-FCV ensures generalization across numerous iterations. The paired t-test statistically validates that the improvements result from the model's advanced learning architecture, not from random fluctuations. In comparison to state-of-the-art models such as HighwayNet, VahigoNet demonstrates state-of-the-art predictive effectiveness, attaining an accuracy of 0.990, precision of 0.999, recall of 0.982, F1-score of 0.990, and AUC-ROC of 0.999. Additionally, explainability and interpretability are improved using XAI methodologies like SHAP and LIME, making the decision-making process more transparent for healthcare professionals. While it has a longer execution time compared to other baseline models, the trade-off can be justified due to significant enhancements in accuracy and reliability, making VahigoNet an appropriate tool for hypertension prediction.

Acknowledgments

The authors extend their appreciation to the Ongoing Research Funding Program for Project number (ORF-2025-648), at King Saud University for supporting this research project.

Funding

This project is funded by Ongoing Research Funding Program for Project number (ORF-2025-648), King Saud University, Riyadh, Saudi Arabia.

References

[1] Kanegae, H., Suzuki, K., Fukatani, K., Ito, T., Harada, N. and Kario, K., 2020. Highly precise risk prediction model for new-onset hypertension using artificial intelligence techniques. The Journal of Clinical Hypertension, 22(3), pp.445-450.

[2] Wu, X., Yuan, X., Wang, W., Liu, K., Qin, Y., Sun, X., Ma, W., Zou, Y., Zhang, H., Zhou, X. and Wu, H., 2020. Value of a machine learning approach for predicting clinical outcomes in young patients with hypertension. Hypertension, 75(5), pp.1271-1278.

[3] Kowalski, S., Goniewicz, K., Moskal, A., Al-Wathinani, A.M. and Goniewicz, M., 2023. Symptoms in hypertensive patients presented to the emergency medical service: a comprehensive retrospective analysis in clinical settings. Journal of clinical medicine, 12(17), p.5495.

[4] Zhao, H. and Zhang, X., 2021. Predicting the risk of hypertension based on several easy-to-collect risk factors: a machine learning method. Frontiers in Public Health, 9, p.619429.

[5] Montagna, S., Pengo, M.F., Ferretti, S., Borghi, C., Ferri, C., Grassi, G., Muiesan, M.L. and Parati, G., 2022. Machine learning in hypertension detection: A study on world hypertension day data. Journal of Medical Systems, 47(1), p.1.

[6] Asadullah, M., Hossain, M.M., Rahaman, S., Amin, M.S., Sumy, M.S.A., Parh, M.Y.A. and Hossain, M.A., 2023. Evaluation of machine learning techniques for hypertension risk prediction based on medical data in Bangladesh. Indonesian Journal of Electrical Engineering and Computer Science, 31(3), pp.1794-1802.

[7] Amaratunga, D., Cabrera, J., Sargsyan, D., Kostis, J.B., Zinonos, S. and Kostis, W.J., 2020. Uses and opportunities for machine learning in hypertension research. International Journal of Cardiology Hypertension, 5, p.100027.

[8] Wang, M., Wang, J., Hu, Y., Guo, B. and Tang, H., 2022. Detection of pulmonary hypertension with six training strategies based on deep learning technology. Computational Intelligence, 38(5), pp.1684-1706.

[9] Choi, Y.Y., Jeong, H., Lee, J.H., Sung, K.C., Shin, J.H., Kim, H.C., Kim, J.Y. and Kang, D.R., 2021. Cardiovascular disease prediction model in patients with hypertension using deep learning: Analysis of the National Health Insurance Service Database from Republic of Korea. CardioMetabolic Syndrome Journal, 1(2), pp.145-154.

[10] Kim, H., Hwang, S., Lee, S. and Kim, Y., 2022. Classification and prediction on hypertension with blood pressure determinants in a deep learning algorithm. International Journal of Environmental Research and Public Health, 19(22), p.15301.

[11] Koshimizu, H., Kojima, R., Kario, K. and Okuno, Y., 2020. Prediction of blood pressure variability using deep neural networks. International Journal of Medical Informatics, 136, p.104067.

[12] Ye, X., Zeng, Q.T., Facelli, J.C., Brixner, D.I., Conway, M. and Bray, B.E., 2020. Predicting optimal hypertension treatment pathways using recurrent neural networks. International Journal of Medical Informatics, 139, p.104122.

[13] Ni, H., Li, Z., Shao, Z., Guo, M. and Liu, J., 2022. RecogHypertension: early recognition of hypertension based on heart rate variability. Journal of Ambient Intelligence and Humanized Computing, 13(8), pp.3945-3962.

[14] Fang, M., Chen, Y., Xue, R., Wang, H., Chakraborty, N., Su, T. and Dai, Y., 2023. A hybrid machine learning approach for hypertension risk prediction. Neural Computing and Applications, 35(20), pp.14487-14497.

[15] Fitzgerald, O., Perez-Concha, O., Gallego-Luxan, B., Metke-Jimenez, A., Rudd, L. and Jorm, L., 2023. Continuous time recurrent neural networks: overview and benchmarking at forecasting blood glucose in the intensive care unit. Journal of Biomedical Informatics, 146, p.104498.

[16] Villegas, M., Gonzalez-Agirre, A., Gutiérrez-Fandiño, A., Armengol-Estapé, J., Carrino, C.P., Pérez-Fernández, D., Soares, F., Serrano, P., Pedrera, M., García, N. and Valencia, A., 2023. Predicting the evolution of COVID-19 mortality risk: A Recurrent Neural Network approach. Computer Methods and Programs in Biomedicine Update, 3, p.100089.

[17] Lin, S., Qiao, Z., Chen, J. and Huang, Z., 2024, December. A Hypertension Early Warning Model Combining Generative Adversarial Networks and Long Short-Term Memory Neural Networks. In International Conference on Web Information Systems Engineering (pp. 341-356). Singapore: Springer Nature Singapore.

[18] Alharbi, A., Alosaimi, W., Sahal, R. and Saleh, H., 2021. Real-Time System Prediction for Heart Rate Using Deep Learning and Stream Processing Platforms. Complexity, 2021(1), p.5535734.

[19] Cano, J., Bertomeu-González, V., Fácila, L., Hornero, F., Alcaraz, R. and Rieta, J.J., 2023. Improved Hypertension Risk Assessment with Photoplethysmographic Recordings Combining Deep Learning and Calibration. Bioengineering, 10(12), p.1439.

[20] Cheng, R., Huang, Y., Hu, W., Chen, K. and Xie, Y., 2025. A Deep Convolution Method for Hypertension Detection from Ballistocardiogram Signals with Heat-Map-Guided Data Augmentation. Bioengineering, 12(3), p.221.

[21] Shrivastava, R.K., Singh, S.P., Singh, S. and Sajwan, M., 2023. Understanding of healthcare problems and solutions using deep learning. In Deep Learning in Personalized Healthcare and Decision Support (pp. 329-341). Academic Press.

[22] Emam MM, Samee NA, Jamjoom MM, Houssein EH (2023) Optimized deep learning architecture for brain tumor classification using improved Hunger Games Search Algorithm. Comput Biol Med 160:106966

[23] Arya, P.K., Sur, K., Kundu, T., Dhote, S. and Singh, S.K., 2025. Unveiling predictive factors for household-level stunting in India: A machine learning approach using NFHS-5 and satellite-driven data. Nutrition, 132, p.112674.

[24] Sri vidhya, G. and Nagarajan, R., 2024. A novel bidirectional LSTM model for network intrusion detection in SDN-IoT network. Computing, 106(8), pp.2613-2642.

[25] Jawad, K.T., Verma, A., Amsaad, F. and Ashraf, L., 2025. A Study on the Application of Explainable AI on Ensemble Models for Predictive Analysis of Chronic Kidney Disease. IEEE Access.

[26] Sharma, A., Singh, P.K. and Chandra, R., 2022. SMOTified-GAN for class imbalanced pattern classification problems. Ieee Access, 10, pp.30655-30665.

[27] Ammar, M., Javaid, N., Alrajeh, N., Shafiq, M. and Aslam, M., 2024. A novel blending approach for smoking status prediction in hidden smokers to reduce cardiovascular disease risk. IEEE Access.

[28] Almars, A.M. and Alharbi, K.M., 2024. Corporate social responsibility disclosure prediction using LSTM neural network. Computing, 106(2), pp.607-625.

[29] Bai, Q., Chen, H., Liu, H., Li, X., Chen, Y., Guo, D., Song, B. and Yu, C., 2025. Molecular structure of NRG-1 protein and its impact on adult hypertension and heart failure: A new clinical Indicator diagnosis based on advanced machine learning. International Journal of Biological Macromolecules, p.140955.

[30] Gupta, K., Bajaj, V., Ansari, I.A. and Acharya, U.R., 2022. Hyp-Net: Automated detection of hypertension using deep convolutional neural network and Gabor transform techniques with ballistocardiogram signals. Biocybernetics and Biomedical Engineering, 42(3), pp.784-796.

[31] Wang, T., Huang, Y., Zhang, H., Li, X. and Li, F., 2025. Machine learning models for quantitatively prediction of toxicity in macrophages induced by metal oxide nanoparticles. Chemosphere, 370, p.143923.

[32] Ture, M., Kurt, I., Kurum, A.T. and Ozdamar, K., 2005. Comparing classification techniques for predicting essential hypertension. Expert Systems with Applications, 29(3), pp.583-588.

[33] Pingle, Y.P. and Ragha, L.K., 2025. Statistical assessment of Indian classical music's role in diabetes, thyroid, hypertension management. International Journal of Information Technology, 17(1), pp.423-429.

[34] Ukwuoma, C.C., Cai, D., Eziefuna, E.O., Oluwasanmi, A., Abdi, S.F., Muoka, G.W., Thomas, D. and Sarpong, K., 2025. Enhancing histopathological medical image classification for Early cancer diagnosis using deep learning and explainable AI–LIME and SHAP. Biomedical Signal Processing and Control, 100, p.107014.

[35] Sushmitha, G.L.N.D. and Utukuru, S., 2025. Age-based disease prediction and health monitoring: integrating explainable AI and deep learning techniques. Iran Journal of Computer Science, pp.1-10.

[36] Wu, Y., Zhang, L., Bhatti, U.A. and Huang, M., 2023. Interpretable machine learning for personalized medical recommendations: A LIME-based approach. Diagnostics, 13(16), p.2681.

[37] Javed, A., Javaid, N., Hasnain, M., Sarfraz, U., Ahmed, I., Shafiq, M. and Choi, J.G., 2024. Applying advanced data analytics on pregnancy complications to predict miscarriage with explainable AI. IEEE Access.

[38] Herrero-Tudela, M., Romero-Oraá, R., Hornero, R., Tobal, G.C.G., López, M.I. and García, M., 2025. An explainable deep-learning model reveals clinical clues in diabetic retinopathy through SHAP. Biomedical Signal Processing and Control, 102, p.107328.