Harnessing Artificial Intelligence for Heart Disease: Comparing Machine Learning, Deep Learning, and Bayesian Network Approaches

1st Chaganti Varshini
School of Computing
Amrita Vishwa Vidyapeetham
Amaravati, India
av.en.u4aie22008@av.students.amrita.edu

2nd Vanga Navya Sri
School of Computing
Amrita Vishwa Vidyapeetham
Amaravati, India
av.en.u4aie22044@av.students.amrita.edu

3rd Nakka Sravya
School of Computing
Amrita Vishwa Vidyapeetham
Amaravati, India
av.en.u4aie22023@av.students.amrita.edu

4th Dr. M. Srinivas
School of Computing
Amrita Vishwa Vidyapeetham
Amaravati, India
m_srinivas@av.amrita.edu

Abstract*—Heart disease is still a global health challenge; thus, the model at an early stage of prediction in a timely intervention remains a challenge. It compares various techniques in machine learning, deep learning, and Bayesian networks against the UCI Cleveland Heart Disease dataset, whereby it compares with the Kaggle Heart Disease dataset. Attempts have been made for the following models: XGBoost, Random Forest (RF), Gradient Boosting, Logistic Regression (LR), Support Vector Classifier (SVM), Convolutional Neural Networks (CNN), and Bayesian Networks. One-hot encoding, ANOVA F-test-based feature selection, and SMOTE will be used for advanced preprocessing for higher performance. In ensemble methods, XGBoost was the best method with accuracy of about 97.32% while the second highest accuracy rate was given by Random Forest with about 95.89%. This result probably justifies the applicability of ensemble models in medical decisions. A very high performance was observed while using a measure that is derived from precision, recall, and F1-score. The paper puts into the limelight the necessity for feature engineering, missing data correction, and class imbalance correction for higher predictive reliability. Future research should be placed on DL with a more deep architecture and multiple data sets for the practical application in healthcare with improved prediction accuracy.

***Keywords*—Heart Disease Prediction, Machine Learning (ML), Deep Learning (DL), Bayesian Networks, Ensemble Techniques, XGBoost, Random Forest, SMOTE, ANOVA-F- Test Predictive Analytics, Medical Diagnostic**

I. Introduction

Cardiovascular diseases (CVDs) are responsible for nearly 31% of global mortality, causing approximately 17.9 million deaths annually, as cited by WHO. Early and accurate diagnosis is crucial as CVD significantly impacts global health. Traditionally, heart disease diagnosis relies on a physician's judgment, which, while effective, can be time-consuming and expensive. This has led to a growing demand for automated systems that can quickly and accurately assess clinical data and serve as decision-support tools for clinicians. Machine learning (ML) techniques are promising in this regard, uncovering complex patterns in large datasets that conventional methods may overlook. The biomedical domain has witnessed rapid development in using ML for prediction and diagnosis, particularly for diseases like diabetes and cardiovascular disorders. ML algorithms can develop predictive models from healthcare data, aiding clinical decisions efficiently in terms of time and cost. Additionally, integrating machine learning models with real-time healthcare systems, such as electronic health records (EHRs) and wearable devices, has the potential to revolutionize heart disease prediction by providing continuous monitoring, real-time risk assessment, and personalized recommendations. This study focuses on applying classification techniques in ML to predict heart disease, evaluating algorithms such as ANN, SVM, logistic regression, kNN, decision trees, and Naïve Bayes. The performance of these methods will be assessed using metrics like accuracy, sensitivity, specificity, and ROC curves to determine their clinical applicability. The objectives are to (1) develop an integrated workflow combining ML models for heart disease prediction and (2) evaluate the most dependable algorithms for clinical use. This work aims to contribute meaningfully to medical informatics by enhancing ML's role in predicting heart disease and supporting healthcare providers in making precise diagnoses and quicker decisions.

979-8-3315-2751-8/25/$31.00 ©2025 IEEE

II. Literature Survey

Dwivedi (2016) has carried out an in-depth review of machine learning approaches for heart disease prediction using the UCI StatLog Heart Disease Dataset. The work considers various algorithms, such as Logistic Regression, Support Vector Machines (SVM), Artificial Neural Networks (ANN), and K-Nearest Neighbors (KNN). The best result was obtained by Logistic Regression with an accuracy of 85%, sensitivity of 89%, and specificity of 81%. This study emphasized the ability of machine learning algorithms to improve diagnostic accuracy and provide reliable decision support in medical applications.[1]

Mohan et al. (2019) presented a new approach for enhancing the accuracy of prediction by integrating feature selection techniques with hybrid models. Utilizing the UCI Cleveland Heart Disease Dataset, they introduced a hybrid Random Forest model integrated with a linear model. This hybrid model performed better than the traditional machine learning methods, with an accuracy of 88.7%. The study highlighted the significance of feature selection in reducing dimensionality and improving the performance of the model. It also showed that hybrid models can exploit the strengths of individual models to make them very suitable for complex prediction tasks.[2]

Kwon et al. (2019) proposed a deep learning-based model to predict in-hospital mortality among heart disease patients. Using echocardiography data, their deep learning model achieved an area under the receiver operating characteristic curve of 0.912 during internal validation. The study also compared the performance of the deep learning model against traditional scoring systems, such as GRACE and TIMI, commonly used in clinical practice. The results showed that this deep learning approach was significantly better than the traditional ones, which makes it possible to analyze complex data of high dimensions and provide much better predictive accuracy.[3]

Sharma et al. (2020) compared different machine learning algorithms, such as Random Forest, SVM, Naive Bayes, and Decision Tree, for predicting heart disease. Using the UCI Cleveland Heart Disease Dataset, Sharma et al. found that the most accurate algorithm for heart disease prediction was Random Forest. This method demonstrated superior performance in terms of accuracy, precision, and recall, establishing its utility in supporting clinical decision-making. Additionally, the study emphasized the importance of incorporating explainable AI techniques to ensure transparency and trust in machine learning applications in healthcare.[4]

Zhou et al. (2024) performed a systematic review of deep learning-based models for heart disease prediction using datasets such as Cleveland, Framingham, and others. Their study highlighted how CNN was powerful, with high accuracy of 96% on balanced datasets. Second, the researchers discussed the setbacks of deep learning, such as data scarcity and class imbalance, in handling diverse populations of patients, requiring integration of multiple models. As a conclusion to the review, the study recommended focusing on developing integrated deep learning frameworks to improve prediction accuracy while addressing these setbacks.[5]

Patel et al. (2021) pointed out hybrid approaches to enhance the performance of heart disease prediction models. Using the Framingham Heart Study Dataset, they hybridized the approach of Decision Trees and SVM to form a hybrid ensemble. The overall accuracy was 90.2%. They proved that the combination of complementary algorithms indeed enhances the prediction performance. The study further focused on the inclusion of real-world clinical data and handling problems like feature correlation and overfitting in order to come up with models that are sound enough for real-world implementation.[6]

III. Methodology

A. *Dataset*

This paper uses Heart disease dataset from UCI Machine Learning Respository. It consists of 902 rows and 16 columns. They are Id, Age, Sex, Dataset, Cp, Trestbps, Chol, Fbs, Restecg, Thalch, Exang, Oldpeak, Slope, Ca,Thal, Num.

B. *Handling Missing Data*

Missing data occur mainly due to incomplete extraction, corrupted files, or errors associated during the loading of data. Data missing is unavoidable, but it significantly degrades the performance and reliability of predictive models. Several methods to handle missing values have been discussed, which include deleting all the rows in which there exist missing entries, imputing the missing values by using any statistic such as mean, median, mode, using a separate category, estimating the missing values based on other features, and using algorithms which take care of missing data. In this case, we decided to delete rows with missing values since their frequency of occurrence is minimal, thus not affecting the integrity of the overall dataset and the analysis which followed. [21]

C. *Feature Engineering*

1) *One Hot Encoding:*

We use techniques in label encoding and one-hot encoding to convert the categorical columns, like ‘sex’, ‘cp’, and ‘thal’ into a numerical format so that it can be used properly with the machine learning models. The former assigns a unique integer to each category and is the base of the study. It may introduce bias in that process since it assumes an ordinal relationship between the categories that isn’t proper in this case since it pertains to nominal data. For that, one-hot encoding is used. Here, every categorical variable is represented through binary dummy variables. For instance, for a column such as sex with the values ”Male” and ”Female,” the encoding turns it into two columns of binary where one represents ”Male” and another represents ”Female.” These techniques strip away ordinal assumptions so that the model will treat these variables as categorical with no intrinsic ranking. Once both encoding techniques are in place, this makes it more flexible and ensures proper representation of data to the classification model.[22]

D. Feature Selection

The best 12 features for classification tasks are selected using SelectKBest with an ANOVA F-test in this work. ANOVA stands for Analysis of Variance and is used to assess whether means of several groups are significantly different from each other. It calculates the F-statistic-the ratio of variance between groups, and the variance within the groups. The ANOVA F-test is specially well-suited for classification problems when the input features are numerical in nature and the target feature is categorical. In Python, that function is implemented by using the f_classif() function from scikit-learn inside the SelectKBest class, which ranks features according to their F-test scores, selecting the k most statistically significant ones. This filter-based feature selection method filters out only the top-12 features that have the highest F-test scores, retaining a cut-down set of features as the most relevant, which best explains a strong relation to the target variable. The efficiency and effectiveness of the model are then enhanced thereby.[20]

E. Smote

SMOTE stands for Synthetic Minority Over-sampling Technique, or the Synthetic Minority Over-sample Technique. It is one of the popular methods to solve the class-imbalance problem with the aid of synthetic example generation instead of mere duplication of samples. The technique bases the new instances along the line segments joining a sample to its nearest neighbors in the feature space. This method works to perform its operations through calculation of the difference between a minority class sample and its nearest neighbor, multiplying the result by a random factor, and then adding it back to the original sample. New synthetic samples are added to help generalize the decision boundary of the minority class and, therefore, they work to enhance classifier performance. The SMOTE technique has become very widely applied in domains such as fraud detection, medical diagnosis, and many more based on its ability to reduce majority class bias and generally make the model able to generalize for all classes.[23]

F. Test And Train

For this study we have used 70% training and 30% testing.

G. Models And Model Evaluation

*1) **Logistic Regression**:* Logistic Regression is such a simple algorithm in the domain of machine learning, widely utilized for the purpose of binary classification. It computes the chances or probabilities for any event to occur considering the interaction between the input features of that event and the target variable. Considering the log-odds of the outcome and linear relationship of predictors, Logistic Regression can easily be interpreted while incurring minimal computational costs. It is found applicable in gigantic domains: spam detection, medical diagnosis, to name a few diseases prediction; risk assessment: credit scoring since it's clear and effective.[1]

$$z = \sum_{i=1}^{n} w_i x_i + b, \tag{1}$$

where w_i represents the weight for feature x_i, and b is the bias term.

Step 2: Next, we apply the sigmoid function to equation (12). Then, convert the score Z into a probabilty allowing the model to classify the output as zero or one based on a specified threshold.

Further, the sigmoid function is expressed as:

$$\hat{y} = \frac{1}{1 + e^{-z}}, \tag{2}$$

*2) **Gaussian Naive Bayes**:* Gaussian Naive Bayes is the Bayes' theorem-based probabilistic classifier, assuming features to be class-conditionally independent and models continuous features with a Gaussian (normal) distribution. Among all these classifiers, Gaussian Naive Bayes is very fast and simple, and it does well in applications such as text classification (for example, spam filtering), sentiment analysis, or recommendation systems, especially on small to medium-sized datasets.[5]

$$P(c|X) = \frac{P(X|c) \cdot P(c)}{P(X)}, \tag{3}$$

Where,
(P(c—X) = class c likelihood as a function of the feature vector X,
$P(X|c)$ = chance that the feature vector X is appropriate for class c,
$P(c)$ = Shows the previous likelihood linked to class c,
$P(X)$ = conditional likelihood of the feature vector X.

*3) **Random Forest Classifier**:* This is an ensemble learning algorithm whereby a large number of decision trees are built during training, with outputs combined for making predictions. Each of these trees has been trained on some random subset of data and features that are individually just as good for generalization by virtue of their diversity. The strength of a Random Forest is noise tolerance, and it finds wide application in image recognition, fraud detection, medical diagnostics, and customer segmentation.[2]

$$\hat{Y} = \frac{1}{T} \sum_{i=1}^{T} h_i(x), \tag{4}$$

Where,
$h_i(x)$ denotes the prediction made by the i-th tree for the input x, and T represents the total count of trees within the forest.

*4) **Gradient Boosting Classifier**:* Gradient Boosting is an ensemble technique in which the models are built sequentially, and each model is focused on correcting the mistakes of the preceding model. It adjusts its prediction by fitting an optimization function to the loss function at each iteration. Gradient Boosting is working really well for classification as well as regression purposes and handling imbalanced datasets much better as well. It has a wide application in terms of prediction analytics, fraud prevention, ranking systems, and

financial forecasting because the possibility of very accurate result generation exists.[4]

$$F_m(x) = F_{m-1}(x) + \gamma \cdot h_m(x) \tag{5}$$

Where:

- $F_m(x)$: The ensemble model after the m-th iteration.
- $F_{m-1}(x)$: The ensemble model after the $(m-1)$-th iteration.
- $h_m(x)$: The weak learner (e.g., decision tree) fitted to the gradient of the loss function at iteration m.
- γ: The learning rate, a hyperparameter that controls the contribution of each weak learner.

*5) **XGBoost Classifier**:* XGBoost is the optimized version of the gradient boosting algorithm, which maximizes performance besides scaling it up. XGBoost will employ regularization to handle overfitting and apply tree pruning for efficiency purposes, as well as it supports parallel processing that leads to faster computationally. XGBoost is fast and accurate and favored in the application in machine learning competition. It has been applied in several applications in several areas of industry, like predicting customer churn, building recommendation systems, predictive maintenance.[5]

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(y_i, \hat{y}i) + \sum k = 1^K \Omega(f_k), \tag{6}$$

For regression tasks,

$$l(y_i, \hat{y}_i) = \frac{1}{2}(y_i - \hat{y}_i)^2 \quad , \tag{7}$$

For classification tasks,

$$l(y_i, \hat{y}_i) = -y_i \log(\hat{y}_i) - (1 - y_i) \log(1 - \hat{y}_i) \quad , \tag{8}$$

Regularization is applied in XGBoost to minimize overfitting. The regularization term is defined as:

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2. \tag{9}$$

where:

$\mathcal{L}(\phi)$ is the overall objective function,

$l(y_i, \hat{y}_i)$ is the loss function,

$\Omega(f_k)$ is the regularization term to control model complexity,

γ is the penalty associated with the number of leaves, T,

λ is the L2 regularization parameter, and

w_j is the weight of the j-th leaf.

*6) **Support Vector Classifier(SVC)**:* Support Vector Classifier is another name of supervised learning algorithm, which identifies the optimum hyperplane that can separate the data points belonging to different classes. Maximizing the margin between classes produces maximal margin, which assures better generalization to new data. SVC can be used both in linear and non-linear classification problems through application of kernel tricks called polynomial and radial basis functions (RBF). It is very efficient in high-dimensional space and thus often used in applications such as facial recognition, bioinformatics, or text classification.[18]

$$w \cdot x + b = 0, \tag{10}$$

Let w be a weight vector. x represents an input vector. b denotes the bias term.

*7) **CNN**:* CNNs are deep models that have been specially implemented specifically for the processing of data in grid form, especially image-based. Through the automatic detection and extraction of hierarchical features such as edges, shapes, and textures, using convolutional layers, CNNs achieve incredible performance in tasks on images, such as object detection, image segmentation, facial recognition, and much more. Apart from image processing, CNNs can be used in medical imaging, autonomous vehicles, or in some augmented reality applications where the spatial patterns are to be captured.[9]

*8) **Baysian Networks**:* Bayesian Networks are probabilistic graphical models; a Directed Acyclic Graph captures the relationship between variables. They capture dependencies conditionally between variables and enable reasoning under uncertainty. Thus, they are very much needed for data that is incomplete and uncertain. Bayesian Networks are very applicable in decision-making systems, in medical diagnostics, fault detection, and risk analysis.[21]

*9) **Cross Validation**:* Cross-validation is utilized in determining the performance of a learning machine model. This has been done by splitting the same into several folds within the dataset. The model learns about some of the folds and, as such, tests on the other one. In this regard, this is done until each fold has served as the test set once. Most often employed is k-fold cross-validation whereby the data are divided into k subsets[22]. Cross-validation lends a probability of demonstrating how the model generalizes to new, unseen data, thus preventing overfitting. This is also a more robust measure of the performance of the model when only a very small or constrained dataset is available.[20]

*10) **Accuracy**:* Accuracy is probably one of the most commonly used metrics for ascertaining the success of a classification model. This can be thought of as how well the model predicts correct values for all classes, all against the total number of correct predictions compared with all the predictions performed by the model. While accuracy is useful in most situations, it is indeed misleading while dealing with imbalanced data. In imbalanced cases, one class dominates the other significantly. In such a case, high accuracy may not necessarily mean the model is actually right because it might be guessing the majority class most of the time.[20]

$$\text{Accuracy} = \frac{T_p + T_n}{T_n + T_p + F_n + F_p} \tag{11}$$

*11) **Precision**:* Precision is a measure that tells us how many instances the model really predicts are positive. In other words, precision measures the accuracy of positive predictions. Precision is particularly useful in such applications where

false positives could be costly. For example, testing for medic serious diseases, cancer for instance, high precision means that anytime the model provides a diagnosis, then it is probably correct when the result returned for the test is positive hence saving patients the costs of unnecessary treatment or testing for tests that are not infected.[23]

$$\text{Precision} = \frac{T_p}{T_p + F_p} \tag{12}$$

12) ***Recall****:* Recall also known as Sensitivity or True Positive Rate measures how many actual positive instances were rightly identified by the model. This measure is based on how well the model performs in accurately identifying the actual positive cases. If the cost of a false negative is high and such as fraud detection or medical diagnosis, then high recall is vital. There does exist a model that has high recall to ensure no critical case goes unreported, at the possible cost of raising false positives. [23]

$$\text{Recall} = \frac{T_p}{T_p + F_n} \tag{13}$$

13) ***F1-Score****:* The harmonic mean of precision and recall is the F1-score-the combination of both into a single value. A method especially helpful for achieving a trade-off between precision and recall is especially useful where class distributions are skewed. A high F1-score implies a model that has neither false positives nor false negatives. Thus, the F1-score becomes very effective when both precision and recall can become equally important for an application such as medical diagnosis, fraud detection or information retrieval.

$$\text{F1 score} = 2\left(\frac{P \cdot R}{P + R}\right) \tag{14}$$

where Tp = instances which are correctly identified as positive, Fp = instances which are wrongly identified as positive, Fn = instances which are wrongly marked as negative, Tn = instances which are correctly marked as negative.[4]

IV. Results And Discussion

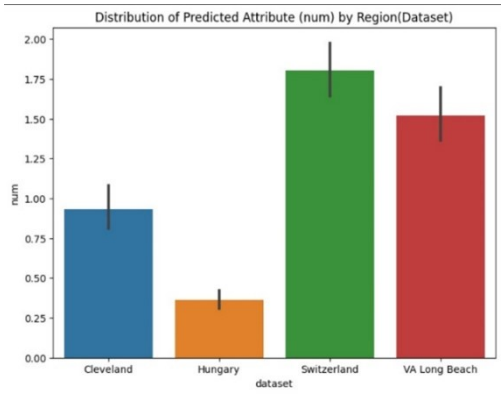

Fig. 1. Distribution of predicted attribute

The image is a bar chart that shows the distribution of a predicted attribute (num) across different regions (datasets).

This paper examines a number of machine learning, deep learning, and Bayesian network models developed for heart disease prediction purposes. Of all applied models, the XGB classifier reached the highest accuracy, 97.32%, with all others doing less good. A close second was taken by Random Forest with an accuracy of 95.89%. Then came Gradient Boosting at 94.64%. The rest of the models used, namely Logistic Regression and GaussianNB, showed fine performance with accuracies at 94.46% and 94.29%, respectively. SVC could produce an accuracy score of 93.04%, while CNN and Bayesian Network were 82.75% and 84.21%, respectively. Therefore, with these results, ensemble methods are proven to be very efficient, with special emphasis on the XGB classifier, in heart disease prediction.

TABLE I
Performance Metrics of Various Classification Model

Model Type	Accuracy	Precision	Recall	F1-Score
XGBoost	97.32%	98.33%	96.67%	97.09%
Random Forest	95.89%	98.33%	94.17%	95.66%
Gradient Boosting	94.64%	96.00%	94.17%	94.89%
Logistic Regression	94.46%	98.00%	92.17%	94.57%
GaussianNB	94.29%	90.00%	97.50%	92.33%
SVC	93.04%	94.67%	92.17%	92.57%
CNN	82.75%	100.00%	64.28%	78.26%
Bayesian Network	84.21%	80.00%	88.88%	84.21%

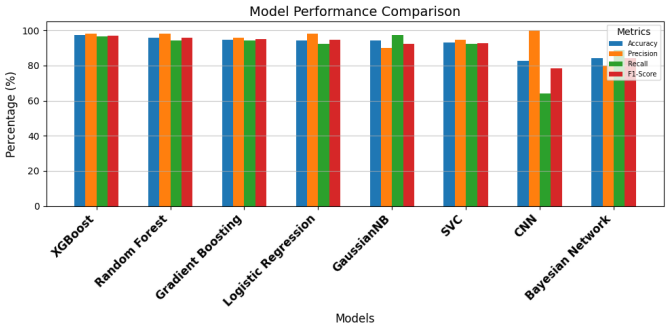

Fig. 2. Performance Metrics of Various Classification Model

V. Conclusion

Heart disease is still menacing the world, and the demand for precise predictive models will now ensure that patients receive diagnosis early for better treatment outcomes. For these purposes, the three approaches, namely ML, DL, and the Bayesian network approaches, are evaluated with UCI Cleveland and Kaggle Heart Disease datasets. Among all such models, ensemble techniques XGBoost and Random Forest achieved the highest accuracies of 97.32% and 95.89% respectively with high evaluation metrics that include precision and F1-score. Gradient Boosting, Logistic Regression, and Gaussian Naive Bayes were very good, but CNN and Bayesian Networks seemed to be very feasible for a couple of applications. Preprocessing techniques used in the paper are one-hot encoding and ANOVA F-test and SMOTE. It is at their best to perform with these and have successfully led

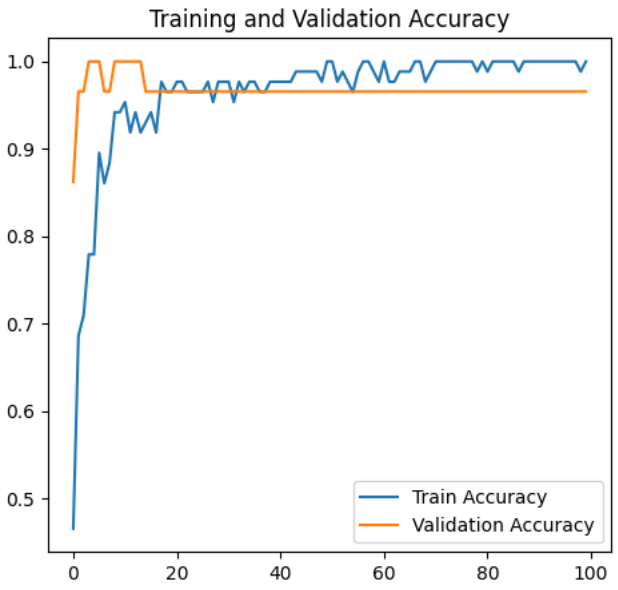

Fig. 3. Distribution of predicted attribute

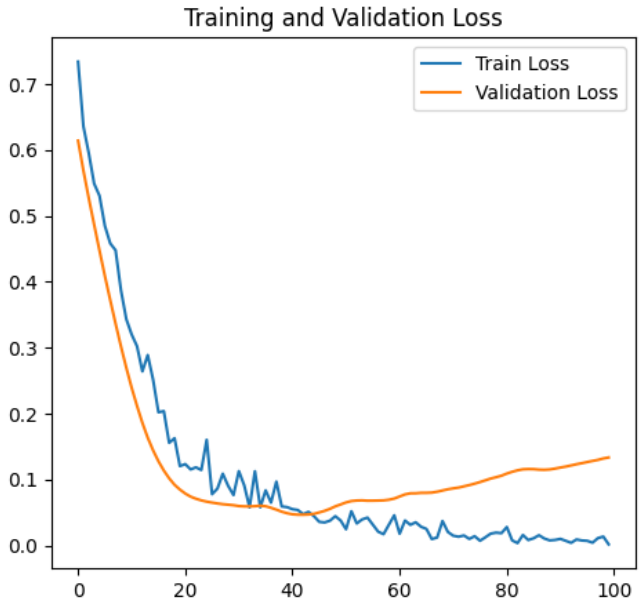

Fig. 4. Distribution of predicted attribute

to the best output for a model. Nevertheless, limitations still lie with data imbalanced, missing values, and limited diversity datasets. Thus, future work should integrate advances in DL architectures along with larger, diverse datasets that improve predictive accuracy and clinical applicability

REFERENCES

[1] A. K. Dwivedi, "Performance evaluation of different machine learning techniques for prediction of heart disease," *Neural Computing and Applications*, vol. 29, pp. 685–693, 2018.

[2] M. M. Ali, et al., "Heart disease prediction using supervised machine learning algorithms: Performance analysis and comparison," *Computers in Biology and Medicine*, vol. 136, p. 104672, 2021.

[3] J.-M. Kwon, et al., "Deep learning for predicting in-hospital mortality among heart disease patients based on echocardiography," *Echocardiography*, vol. 36, no. 2, pp. 213–218, 2019.

[4] R. Katarya and S. K. Meena, "Machine learning techniques for heart disease prediction: a comparative study and analysis," *Health and Technology*, vol. 11, no. 1, pp. 87–97, 2021.

[5] V. Sharma, S. Yadav, and M. Gupta, "Heart disease prediction using machine learning techniques," in *2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, IEEE, 2020.

[6] C. Zhou, et al., "A comprehensive review of deep learning-based models for heart disease prediction," *Artificial Intelligence Review*, vol. 57, no. 10, p. 263, 2024.

[7] M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.

[8] A. Singh and R. Kumar, "Heart disease prediction using machine learning algorithms," in *2020 International Conference on Electrical and Electronics Engineering (ICE3)*, IEEE, 2020.

[9] R. Bharti, et al., "Prediction of heart disease using a combination of machine learning and deep learning," *Computational Intelligence and Neuroscience*, vol. 2021, no. 1, p. 8387680, 2021.

[10] S. B. M. George, S. H. Bleiberg, N. Alawa, and D. Sanghavi, "Case study: Delivery and payment reform in congestive failure at two large academic centers," *Elsevier*, vol. 2, pp. 107–112, 2014. DOI: 10.1016/j.hjdsi.2014.04.001.

[11] WHO, "Cardiovascular diseases (CVDs)," [Online]. Available: https://www.who.int/en/newsroom/fact-sheets/detail/cardiovascular-diseases(cvds). [Accessed: 2016].

[12] P. D. Bijl, V. Delgado, and J. J. Bax, "Imaging for sudden cardiac death risk stratification: Current perspective and future directions," *Progress in Cardiovascular Diseases*, vol. 62, no. 3, pp. 205–211, 2019.

[13] S. Varun, G. Mounika, P. Sahoo, and K. Eswaran, "Efficient system for heart disease prediction by applying logistic regression," *International Journal of Computer Science and Technology*, vol. 10, no. 1, 2019.

[14] WHO, "Global hearts initiative," [Online]. Available: https://www.who.int/cardiovascular-diseases/globalhearts/Global-hearts-initiative/en/. [Accessed: July 03, 2017].

[15] Y. Wu, Z. He, H. Lin, et al., "A fast projection-based algorithm for clustering big data," *Interdisciplinary Sciences: Computational Life Sciences*, vol. 11, pp. 360–366, 2019. DOI: 10.1007/s12539-018-0294-3.

[16] H. USF, "Data mining in healthcare," [Online]. Available: https://www.usfhealthonline.com/resources/keyconcepts/data-mining-in-healthcare/.

[17] M. H. Tekieh and B. Raahemi, "Importance of data mining in healthcare," in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 2015. DOI: 10.1145/2808797.2809367.

[18] Wiharto, H. Kusnanto, and Herianto, "Performance analysis of multiclass support vector machine classification for diagnosis of coronary heart diseases," *International Journal on Computational Science and Applications (IJCSA)*, vol. 5, no. 5, pp. 27–37, 2015. DOI: 10.5121/ijcsa.2015.550.

[19] J. Patel, T. Upadhyay, and D. S. Patel, "Heart disease prediction using machine learning and data mining techniques," *International Journal of Computer Science and Communication*, vol. 7, no. 1, pp. 129–137, 2016. DOI: 10.090592/IJCSC.2016.018.

[20] M. S. Pathan, A. Nag, M. M. Pathan, and S. Dev, "Analyzing the impact of feature selection on the accuracy of heart disease prediction," *Healthcare Analytics*, vol. 2, p. 100060, 2022.

[21] M. Muibideen and R. Prasad, "A fast algorithm for heart disease prediction using Bayesian network model," *arXiv preprint*, arXiv:2012.09429, 2020.

[22] T. Al-Shehari and R. A. Alsowail, "An insider data leakage detection using one-hot encoding, synthetic minority oversampling and machine learning techniques," *Entropy*, vol. 23, no. 10, p. 1258, 2021.

[23] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.