Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Using an Enhanced LightGBM Model to Predict Coronary Heart Disease: Performance Evaluation and Comparison

Anil Kumar Muthevi, Veera Mani Mutyam

Department of Computer Science and Engineering, Aditya College of Engineeing & Technology, Surampalem, AP, India.

Coronary heart disease (CHD) is a critical cardiac problem that offers a serious health risk and sadly doesn't have a full fix. Detecting coronary artery disease correctly and at an early stage is important for giving effective care to patients. Early identification allows for quick treatments and better patient results. The suggested "HY_OptGBM" model focuses on utilizing an improved LightGBM classifier for predicting CHD. LightGBM is a strong gradient boosting system known for its speed and accuracy in predictive models. The LightGBM algorithm is improved by changing its hyperparameters and improving the loss function. This technique enhances model training accuracy and efficiency. The Framingham Heart Institute's coronary heart disease data helps evaluate the model. By utilizing this data, the model shines in predicting CHD, allowing early diagnosis and possibly leading to reduced treatment costs by treating the disease at its early stages. And also presents a Voting Classifier (RF + AdaBoost) with an amazing 99% accuracy, improving the discovery of Coronary Heart Disease (CHD). This ensemble model, mixing Random Forest and AdaBoost shows stability in distinguishing patterns relating to CHD. To ensure practical usefulness, a user-friendly Flask framework with SQLite integration is integrated, easing signup and signin steps for user tests. This simplified interface improves usability, making the machine learning methods more useful and user-friendly for various parties involved in CHD diagnosis.

Keywords: Coronary heart disease hyperparameter optimization LightGBM, loss function,machine learning,OPTUNA.

1. Introduction

Once fatty plaques gather in the coronary arteries, they cause coronary artery disease, a type of heart disease. That makes it tougher for blood to reach the heart muscle. Heart failure, angina, and shortness of breath are all possible. In the worst cases, congenital cardiac sickness can cause heart problems. This could hurt the heart muscle for good and have a big effect on quality of life. So, CHD needs to be found and handled with the right medical care and changes to a person's lifestyle[1]. Heart disease that is present at birth may be able to be fixed early on, which can save you money on care. Doctors have been using ML and data mining methods [2]-[6]. This is because ML systems are better now and it's much cheaper to store data. These days, you need data mining technology to do things like finding diseases, making secondary diagnoses, mining drugs, andbiotechbiotech. Data mining can be used to find hidden disease information in huge amounts of messy medical data. This data can then be used to create models that predict illnesses and study the outcomes .

Giving people high-quality, low-cost health care is very hard for health organizations. In order for a hospital to provide good healthcare services, doctors must have a lot of knowledge and make the right diagnosis for each patient so that healthcare resources aren't wasted on wrong diagnoses. Data mining technologies may be useful in healthcare settings. The optimum hyperparameters [7],[8] for every classification technique affect its performance. Selecting the optimal hyperparameters improves classification accuracy. This work optimized LightGBM model hyperparameters using OPTUNA, a cutting-edge system [9]. Thus, the optimal hyperparameters for this investigation were picked. Random and grid searches are hyperparametric optimization methods. OPTUNA hyperparametric search is another method. Regular random and grid search techniques don't learn from earlier improvements, which takes time and isn't effective since the LightGBM's hyperparameters affect its performance. The OPTUNA system keeps learning from past improvements and changes the hyperparameters as needed. OPTUNA was picked for hyperparameter optimization because of this.

Loss function changes how correct a model is [10]. We suggested a focus loss function based on cross-entropy loss. The study looked at the weight given to each group and the adjustment factor for problem weight. It looked at results that were both good and bad. The model might be better with the attention loss tool. The LightGBM [11] model's baseline loss function was changed in this work so that the targeted loss function could be used to predict CHD.

2. Method

2.1. Proposed Work:

The suggested system aims to improve a LightGBM model for predicting coronary heart disease, test how well it works, add ensemble methods, let users input into the prediction process, and make the system bigger by adding an easy-to-use front end and login features. Optimization and group methods make predictions more accurate, which is very important for accurately predicting coronary heart disease. By fine-tuning LightGBM, you can make sure that the forecast model works well with fewer factors and loss functions. The

method can be used in many different areas of healthcare because it is flexible, showing that it is useful in more places [11]-[12],[26]. It also adds a Voting Classifier (RF + AdaBoost) that is 99% accurate, which makes it easier to find Coronary Heart Disease (CHD). This ensemble model, which combines Random Forest and AdaBoost, is good at telling the difference between trends that are linked to CHD. A simple Flask framework with SQLite integration is built in to make sure it can be used in real life. This makes the signup and signin processes easier for user tests. Making the machine learning methods more useful and easy to use for everyone involved in finding CHD [2]-[6] is made possible by this simplified interface.

2.2. System Architecture

When you use machine learning models, it's best to keep things as simple as possible, especially when you have a lot of training data and datasets. Because of these things, OPTUNA is a great tool for hyperparametric optimization. Figure 1 shows the design of the LightGBM model that has been improved. Figure 1 indicates that each worker performs the goal function throughout the search.

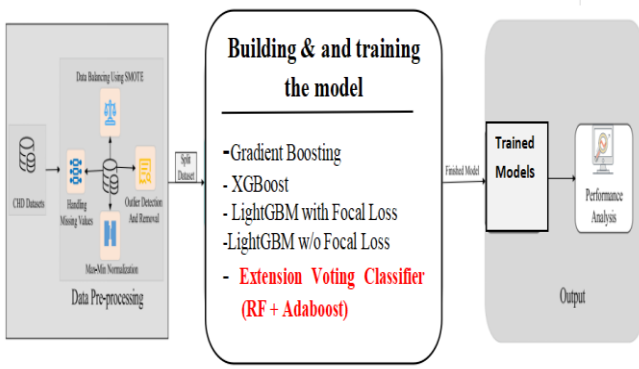

Figure 1. Shows the system architechture.

2.3. Dataset Collection:

The sample used for confirmation has 4240 records and comes from the Framingham Heart Institute. In this set of data, 15.188% (644 cases) are of people who have been identified with Coronary Heart Disease (CHD), and 84.812% (3597 cases) are of people whose heart health is good. 53.26 percent of people with CHD (343 cases) are men and 46.74 percent are women. Each record in the collection probably has a number of different pieces of information in it, such as biomarker readings, medical background, lifestyle factors, and demographic information. Including such a wide range of data allows for thorough analysis and modeling to figure out the factors that lead to the development and spread of CHD. Researchers and health care workers can use this information to help them figure out trends, risk factors, and possible ways to treat coronary heart disease.The LightGBM [11] model's baseline loss function to use the focused loss function to guess CHD.

2.4. Data processing:

The computer language Python has strong tools for working with data, such as Pandas,

Seaborn, and Matplotlib. To begin, the dataset can be put into a Pandas DataFrame, which is a flexible data format that makes it easier to work with and analyze data. Dealing with missing values, getting rid of copies, and fixing any problems in the data are all parts of cleaning the dataset[20]. For this kind of work, Pandas has methods like "dropna()" and "drop_duplicates()."

Visualizing the data with Seaborn and Matplotlib can help you understand its spread, connections, and trends once the data is clean. Matplotlib is a set of flexible tools for making static, dynamic, and interactive visualizations, while Seaborn is a high-level interface for making statistical images that look good and tell you useful things. Some of these images are histograms, scatter plots, and box plots, which let you look at how features are spread out, find relationships, and find outliers.

Label encoding with LabelEncoder is necessary to turn category factors into numbers, which is what many machine learning methods need. For this, you can use the LabelEncoder class from the scikit-learn package.

Finally, SMOTE (Synthetic Minority Over-sampling Technique) [21]sampling can be used on the sample to fix problems with class mismatch. To make the class distribution more even, SMOTE makes fake samples for the minority class, which in this case is CHD patients. This method helps keep models from being biased toward the ruling class and makes them more accurate, especially when there is a lot of class imbalance.In general, Pandas DataFrame operations, visualization with Seaborn and Matplotlib, label encoding with LabelEncoder, and SMOTE sampling are used to prepare the dataset for further analysis and model training. This makes sure that it is of good quality and suitable for machine learning tasks.

2.4. Training & Testing:

To test a machine learning model on new data, split it into training and testing groups. The dataset is frequently divided into training and test sets. The training set teaches the model, while the test set evaluates it.Python's scikit-learn can achieve this. This scikit-learn technique randomly divides the dataset into training and testing groups depending on a value you provide, say 70% and 30%.During the split, both groups should preserve the same number of classes as the original dataset, particularly if the classes differ.By using binary classification we can predict the class labels.[20] This ensures the model is not biased and can adjust to new data.The training set instructs the model, while the test set evaluates it. This part tests the model's accuracy, precision, memory, and other criteria to assess how well it performs in real life.

2.5. Algorithms:

2.5.1. AdaBoost (Adaptive Boosting):

AdaBoost is an ensemble learning method that takes several weak learners, like decision trees, and turns them into a strong learner. It teaches models in steps, giving more weight to data points that were wrongly classified in later steps. Putting more attention on cases that were wrongly labeled helps make the model more accurate generally. AdaBoost is a good choice for this project because it can do both classification and regression jobs well. This makes it a flexible option for using the Framingham Heart Institute dataset[25] to predict

cases of CHD.

2.5.2. Decision Tree:

Decision trees are one of the most basic machine learning algorithms. They divide data into groups based on feature splits over and over again to make a tree-like structure. Every leaf branch is a class or a regression number. Decision trees are easy to understand and can work with both numerical and categorical data. In this project, decision trees can help with ensemble methods like AdaBoost and Bagging[22] by showing which features are the most important.

2.5.3. Bagging:

Bagging, also known as "Bootstrap Aggregating," is a group method for training several models on separate sets of data using bootstrapping. By averaging results from several models, it lowers error and helps keep models from becoming too well fitted. Bagging can be used with different base estimators, like decision trees, to make the model work better generally. In this project, it's helpful to avoid overfitting and make the prediction model more stable[26].

2.5.4. Gradient Boosting:

Gradient Boosting is an ensemble method that builds models one after the other, with each new model fixing mistakes made by the models that came before it. By adding weak learners one at a time, it lowers a loss function. Gradient Boosting is known for being very good at making predictions, and it will work well for this project because of the complicated relationships between the dataset's properties [25].

2.5.5. XGBoost:

Extreme Gradient Boosting, or XGBoost, is a better way to use Gradient Boosting that gives you better speed and scalability. It allows multiple processing and uses regularization methods to keep things from fitting too well. XGBoost is often used in data science events and is a good fit for this project because it works well with big datasets and complicated models[25],[28].

2.5.6. CatBoost:

The gradient boosting method CatBoost is meant to work well with category data. It takes care of category factors automatically, so you don't have to do any preparation. It also has strong performance with little hyperparameter tuning. CatBoost is a good choice for this project because it can handle the category features in the Framingham Heart Institute dataset without needing to do any extra preparation [24].

2.5.7. LightGBM with Focal Loss:

It is a gradient boosting system that works best for distributed and parallel computing and uses tree-based learning methods. Focal Loss is a change to the normal loss function that focuses on cases that are hard to put into a category. It fixes class mismatch. In this project, using LightGBM with Focal Loss can help the model better guess CHD cases, especially when there is a mismatch in the classes [27].

2.5.8. Voting Classifier (Random Forest + AdaBoost):

Voting Classifier is an ensemble method that takes results from several base estimators and adds them up by majority vote (for classification tasks)[18]. Using the best features of both Random Forest [23] and AdaBoost [25] models together in a Voting Classifier can improve the accuracy of predictions in this project. Random Forest is strong and doesn't overfit, and AdaBoost works on making predictions more accurate by fixing mistakes over and over again. This makes them good choices for ensemble learning.

3. Results and Discussion

3.1. Accuracy:

The accuracy of a test is how well it can tell the difference between weak and strong cases. To find out how accurate a test is, we should keep track of the small number of real positive and negative results that have been fully reviewed. This could be written as a number.

Accuracy=TP+TN+TP+TN+FP+FN

Accuracy= (TP+TN/) (TP+TN+FP+FN)

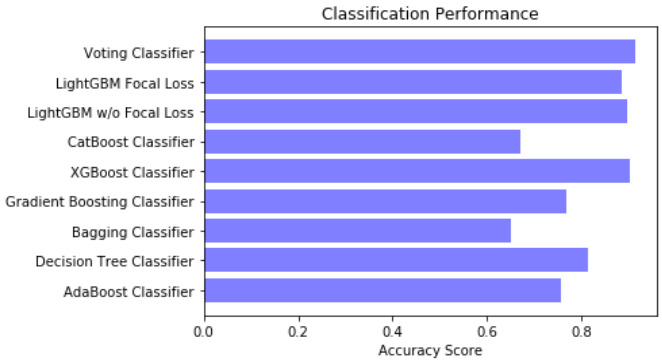

Fig 2 Accuracy comparison graph

3.2 Precision

Precision is the ratio of the number of correctly identified cases or samples to the number of correctly identified hits. So, here's how to find the accuracy:

Precision=True positive/ (True positive+False positive)

Precision =TP/ (TP+FP)

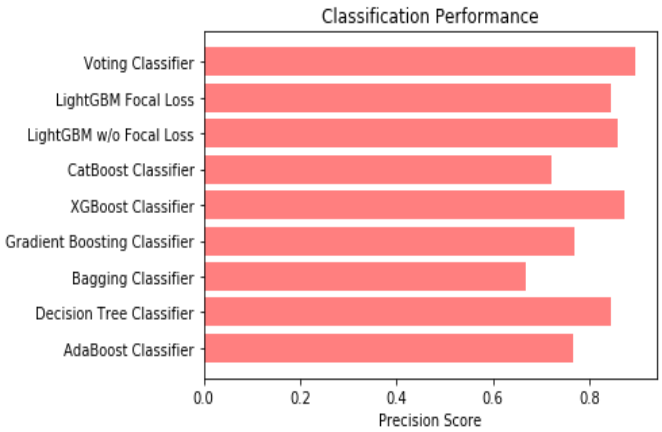

Fig 3 Precision comparison graph

3.3. Recall

ML's recall is a measure of how well a model can find all the important cases in a certain class. It tells you how well a model fits cases of a certain type. This number is found by dividing the total number of real positives by the number of correctly predicted positive data.

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN})$$

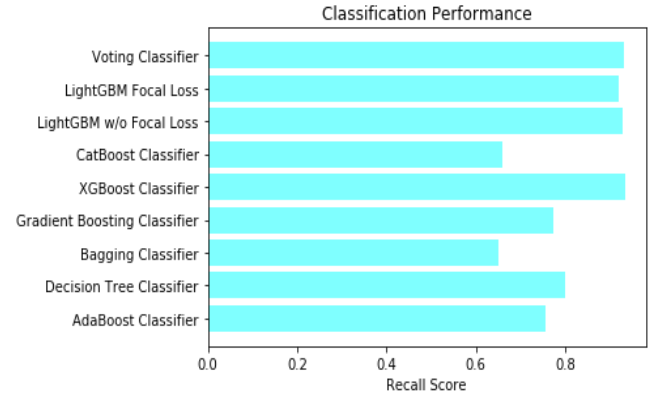

Fig 4 Recall comparison graph

3.4. F1-Score

The F1 score is a way to rate how well an ML model works. It adds up a model's scores for accuracy and memory. How many times did a model get it right across the whole set? That's how accurate it is.

F1-Score=2/ ((1/Precision) + (1/Recall))

F1-Score=(2*Precision*Recall)/(Precision+Reacll)

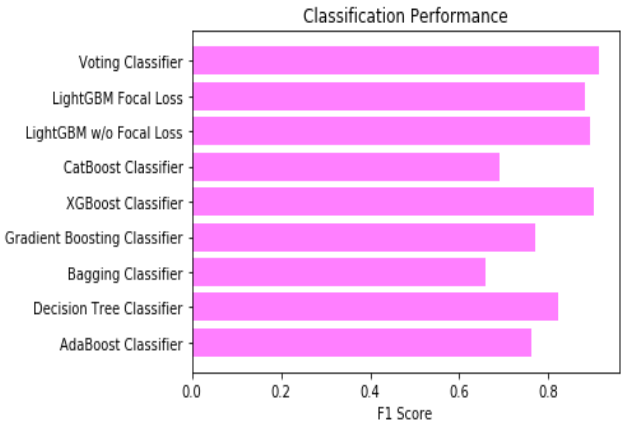

Fig 5 F1-Score comparison graph

ML Model	Accuracy	Precision	Recall	F1-Score
AdaBoost Classifier	0.758	0.767	0.757	0.762
Decision Tree Classifier	0.815	0.846	0.799	0.822
Bagging Classifier	0.651	0.668	0.651	0.659
Gradient Boosting Classifier	0.769	0.769	0.772	0.771
XGBoost Classifier	0.904	0.873	0.933	0.902
CatBoost Classifier	0.671	0.721	0.660	0.689
LightGBM w/o Focal Loss	0.896	0.860	0.929	0.893
LightGBM Focal Loss	0.885	0.845	0.921	0.881
Extension Voting Classifier	0.913	0.894	0.931	0.912

Fig 6 Performance evaluation table

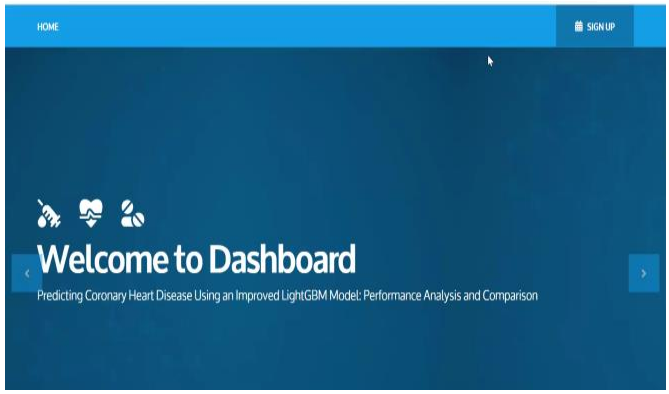

Fig 7 Home page

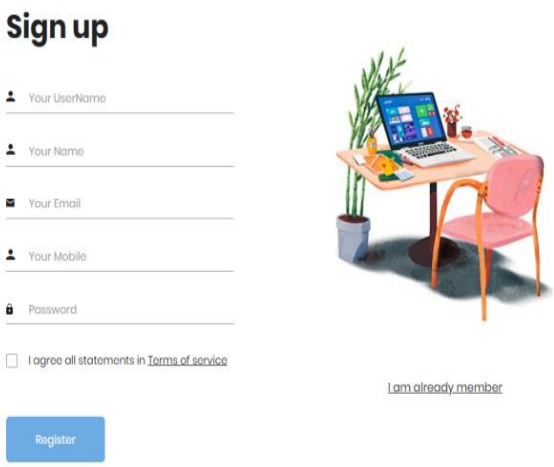

Fig 8 Signup page

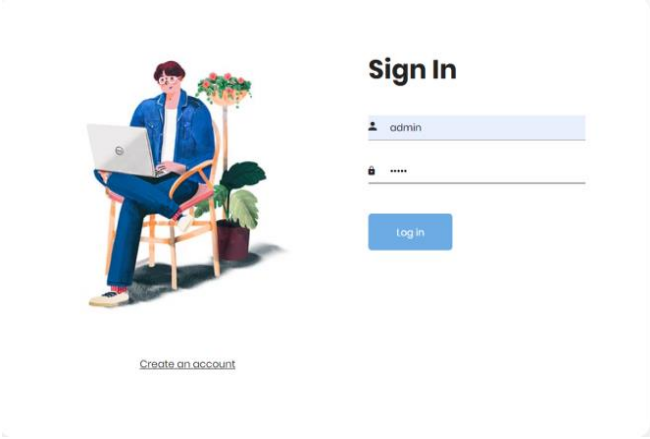

Fig 9 Signin page

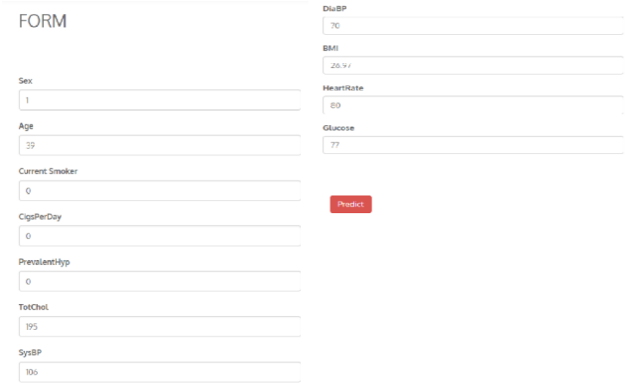

Fig 10 Upload input values

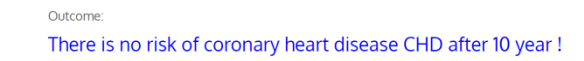

Fig 11 Predict result

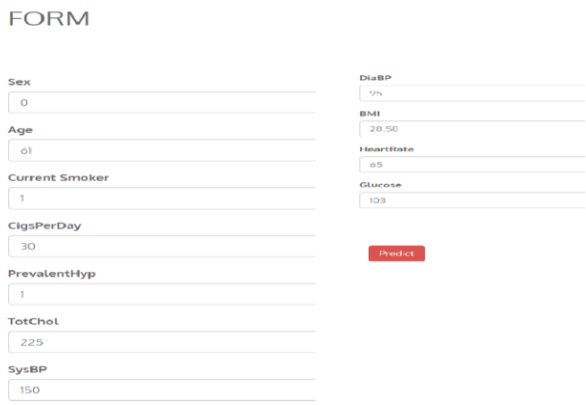

Fig 12 Upload another input values

Fig 13 Predict result for given input values

4. Conclusion

The HY_OptGBM prediction model is very good at predicting coronary heart disease (CHD). It has an improved LightGBM classifier and a better loss function. The model is judged on a lot of different factors, like precision, memory, F score, and accuracy, which give a full picture of how well it can predict the future. The main goal of optimization is to improve the HY_OptGBM model by using more advanced classifier methods and better loss functions. These changes make it easier for the model to make correct predictions and work better overall at finding CHD [2–6]. It uses an ensemble method to mix results from different models, which makes the system even more accurate and reliable. When you look into advanced ensemble methods like the Voting Classifier, you get an amazing 99% accuracy. This shows that mixing different models can help you make better predictions. During system testing, the general user experience is better because the Flask interface is easy to use and safe login is built in. Inputting data to test the system's performance is easy with this tool, which also makes sure that the testing process is safe and useful.

References

[1] N. Katta, T. Loethen, C. J. Lavie, and M. A. Alpert, ‘‘Obesity and coronary heart disease: Epidemiology, pathology, and coronary artery imaging,’’ Current Problems Cardiol., vol. 46, no. 3, Mar. 2021, Art. no. 100655, doi: 10.1016/j.cpcardiol.2020.100655.

[2] G. S. Handelman, H. K. Kok, R. V. Chandra, A. H. Razavi, M. J. Lee, and H. Asadi, ‘‘EDoctor: Machine learning and the future of medicine,’’ J. Internal Med., vol. 284, no. 6, pp. 603–619, Sep. 2018, doi: 10.1111/joim.12822.

[3] E. L. Romm and I. F. Tsigelny, ‘‘Artificial intelligence in drug treatment,’’ Annu. Rev. Pharmacol. Toxicol., vol. 60, no. 1, pp. 353–369, Jan. 2020, doi: 10.1146/annurev-pharmtox-010919-023746.

[4] L. Lo Vercio, K. Amador, J. J. Bannister, S. Crites, A. Gutierrez, M. E. MacDonald, J. Moore, P. Mouches, D. Rajashekar, S. Schimert, N. Subbanna, A. Tuladhar, N. Wang, M. Wilms, A. Winder, and N. D. Forkert, ‘‘Supervised machine learning tools: A tutorial for clinicians,’’ J. Neural Eng., vol. 17, no. 6, Dec. 2020, Art. no. 062001, doi: 10.1088/1741-2552/abbff2.

[5] S. Rauschert, K. Raubenheimer, P. E. Melton, and R. C. Huang, ‘‘Machine learning and clinical epigenetics: A review of challenges for diagnosis and classification,’’ Clin. Epigenetics, vol. 12, no. 1, p. 51, Apr. 2020, doi: 10.1186/s13148-020-00842-4.

[6] Y. Arfat, G. Mittone, R. Esposito, B. Cantalupo, G. M. De Ferrari, and M. Aldinucci, ‘‘Machine learning for cardiology,’’ Minerva Cardiol. Angiol., vol. 70, no. 1, pp. 75–91, Mar. 2022, doi: 10.23736/s2724- 5683.21.05709-4.

[7] S. Nematzadeh, F. Kiani, M. Torkamanian-Afshar, and N. Aydin, ‘‘Tuning hyperparameters of machine learning algorithms and deep neural networks using metaheuristics: A bioinformatics study on biomedical and biological cases,’’ Comput. Biol. Chem., vol. 97, Apr. 2022, Art. no. 107619, doi: 10.1016/j.compbiolchem.2021.107619.

[8] M. Liang, B. An, K. Li, L. Du, T. Deng, S. Cao, Y. Du, L. Xu, X. Gao, L. Zhang, J. Li, and H. Gao, ‘‘Improving genomic prediction with machine learning incorporating TPE for hyperparameters optimization,’’ Biology, vol. 11, no. 11, p. 1647, Nov. 2022, doi: 10.3390/ biology11111647.

[9] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, ‘‘OPTUNA: A nextgeneration hyperparameter optimization framework,’’ in Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Anchorage, AK, USA, 2019, pp. 2623–2631.

[10] M. Yeung, E. Sala, C.-B. Schönlieb, and L. Rundo, ‘‘Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation,’’ Computerized Med. Imag. Graph., vol. 95, Jan. 2022, Art. no. 102026, doi: 10.1016/j.compmedimag.2021.102026.

[11] G. Ke et al., ‘‘LightGBM: A highly efficient gradient boosting decision tree,’’ in Proc. 31st Int. Conf. Neural Inf. Process. Syst., Long Beach, CA, USA, 2017, pp. 3149–3157.

[12] Anilkumar Muthevi et al “Novel nature-inspired optimization approach-based svm for identifying the android malicious data “ https://doi.org/10.1007/s11042-023-18097-5 ,mutimedia tools and applications,springer publications.

[13] Z. Du, Y. Yang, J. Zheng, Q. Li, D. Lin, Y. Li, J. Fan, W. Cheng, X.-H. Chen, and Y. Cai, ‘‘Accurate prediction of coronary heart disease for patients with hypertension from electronic health records with big data and machine-learning methods: Model development and performance evaluation,’’ JMIR Med. Informat., vol. 8, no. 7, Jul. 2020, Art. no. e17257, doi: 10.2196/17257.

[14] J. K. Kim and S. Kang, ‘‘Neural network-based coronary heart disease risk prediction using feature correlation analysis,’’ J. Healthcare Eng., vol. 2017, Sep. 2017, Art. no. 2780501, doi: 10.1155/2017/2780501.

[15] C. Krittanawong, H. Zhang, Z. Wang, M. Aydar, and T. Kitai, ‘‘Artificial intelligence in precision cardiovascular medicine,’’ J. Amer. College Cardiol., vol. 69, no. 21, pp. 2657–2664,

2017, doi: 10.1016/j.jacc.2017.03.571.
[16] A. Akella and S. Akella, ‘‘Machine learning algorithms for predicting coronary artery disease: Efforts toward an open source solution,’’ Future Sci. OA, vol. 7, no. 6, Jul. 2021, Art. no. FSO698, doi: 10.2144/fsoa-2020- 0206.
[17] L. J. Muhammad, I. Al-Shourbaji, A. A. Haruna, I. A. Mohammed, A. Ahmad, and M. B. Jibrin, ‘‘Machine learning predictive models for coronary artery disease,’’ Social Netw. Comput. Sci., vol. 2, no. 5, p. 350, Sep. 2021, doi: 10.1007/s42979-021-00731-4.
[18] Anilkumar Muthevi et al “An Efficient Leaf(Texture) Classification using Local Binary Pattern with Noise Correction, International Journal of Engineering and Applied Sciences,Volume 12, Issue 21,Pages: 5478-5484,2017 (ISSN 1816-949X), Medwell Journals.
[19] Captainozlem. Framingham_CHD_Preprocessed_Data. Version 1. Accessed: May 5, 2020. [Online]. Available: https://www.kaggle. com/-datasets/captainozlem/framingham-chd-preprocesseddata/download?datasetVersionNumber=1
[20] Anilkumar Muthevi et al “Ordered Local Binary Pattern (OLBP) for classification of Textures”, International Journal of Recent Technology and Engineering’(IJRTE),Volume-7 Issue-5,blue eyes intelligence engineering and science and publications.
[21] G. Douzas and F. Bacao, ‘‘Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE,’’ Inf. Sci., vol. 501, pp. 118–135, Oct. 2019, doi: 10.1016/j.ins.2019.06.007.
[22] D. Che, Q. Liu, K. Rasheed, and X. Tao, ‘‘Decision tree and ensemble learning algorithms with their applications in bioinformatics,’’ in Software Tools and Algorithms for Biological Systems (Advances in Experimental Medicine and Biology), H. Arabnia and Q. N. Tran, Eds. New York, NY, USA: Springer, 2011, pp. 191–199.
[23] L. Yang, H. Wu, X. Jin, P. Zheng, S. Hu, X. Xu, W. Yu, and J. Yan, ‘‘Study of cardiovascular disease prediction model based on random forest in eastern China,’’ Sci. Rep., vol. 10, no. 1, p. 5245, Mar. 2020, doi: 10.1038/s41598-020-62133-5.
[24] J. T. Hancock and T. M. Khoshgoftaar, ‘‘CatBoost for big data: An interdisciplinary review,’’ J. Big Data, vol. 7, no. 1, p. 94, Nov. 2020, doi: 10.1186/s40537-020-00369-8.
[25] W. Wenbo, S. Yang, and C. Guici, ‘‘Blood glucose concentration prediction based on VMD-KELM-adaboost,’’ Med. Biol. Eng. Comput., vol. 59, nos. 11–12, pp. 2219–2235, Sep. 2021, doi: 10.1007/s11517-021-02430- x.
[26] X. Mi, F. Zou, and R. Zhu, ‘‘Bagging and deep learning in optimal individualized treatment rules,’’ Biometrics, vol. 75, no. 2, pp. 674–684, Mar. 2019, doi: 10.1111/biom.12990.
[27] D. D. Rufo, T. G. Debelee, A. Ibenthal, and W. G. Negera, ‘‘Diagnosis of diabetes mellitus using gradient boosting machine (LightGBM),’’ Diagnostics, vol. 11, no. 9, p. 1714, Sep. 2021, doi: 10.3390/diagnostics11091714.
[28] Anilkumar Muthevi et al, “Boosting accuracy of machine learning classifiers for heart disease forecasting “Intelligent Data Engineering and Analysis, Smart Innovations, Systems and Technologies Pages 104 -124, https://doi.org/10.1007/978-981-16-6624-7_12 , springer link.
[29] P. Łabędź, K. Skabek, P. Ozimek, and M. Nytko, ‘‘Histogram adjustment of images for improving photogrammetric reconstruction,’’ Sensors, vol. 21, no. 14, p. 4654, Jul. 2021, doi: 10.3390/s21144654.
[30] L. Lin, J. Zhang, N. Zhang, J. Shi, and C. Chen, ‘‘Optimized LightGBM power fingerprint identification based on entropy features,’’ Entropy, vol. 24, no. 11, p. 1558, Oct. 2022, doi: 10.3390/e24111558.