Scientific Journal of Informatics
Vol. 12, No. 2, May 2025

p-ISSN 2407-7658 https://journal.unnes.ac.id/journals/sji/index e-ISSN 2460-0040

Optimization of Coronary Heart Disease Risk Prediction Using Extreme Learning Machine Algorithm (Case Study: Patients of Dr. Soeselo Hospital)

Arie Iswanti[1*], R. Rizal Isnanto[2], Catur Edi Widodo[3]
[1]Department of Information Systems, Universitas Diponegoro, Indonesia
[2]Department of Computer Engineering, Universitas Diponegoro, Indonesia
[3]Department of Physics, Universitas Diponegoro, Indonesia

Abstract.
Purpose: Coronary heart disease (CHD) is the leading cause of death globally, with 17.8 million deaths reported by the WHO in 2021. Early detection remains a major challenge due to low public awareness and dependence on manual diagnostic procedures. These limitations necessitate the development of automated and accurate predictive models. This study aims to construct a CHD risk prediction model using the Extreme Learning Machine (ELM) algorithm. The research addresses a critical limitation in existing models, namely, poor performance on minority classes (CHD stages 2–4), caused by data imbalance. To overcome this, oversampling techniques such as Synthetic Minority Oversampling Technique (SMOTE) and Adaptive Synthetic Sampling (ADASYN) are applied. The objective is to improve classification performance, particularly in high-risk categories, and to enhance the model's generalisation capability for real-world implementation.
Methods: This research implements the Extreme Learning Machine (ELM) algorithm to achieve optimal prediction results. The data used in this study as the initial database of patients consists of gender, age, height, weight, whether they have diabetes or not, the number of cigarettes consumed daily, and blood pressure. The data will be the main component in building the heart disease prediction system. The prediction classes are: no heart disease, stage 1 heart disease, stage 2 heart disease, stage 3 heart disease, and stage 4 heart disease. The total number of dataset are 521 data points, with 70% of the training data amounting to 364 patients, and 30% of the test data amounting to 157 patients. The data collection process uses patient data from RSUD Dr. Soeselo, Tegal Regency, Central Java, for the years 2023 and 2024.
Result: The research successfully developed and evaluated an Extreme Learning Machine (ELM) algorithm for Coronary Heart Disease (CHD) risk prediction using patient data from Dr. Soeselo Hospital. The model achieved an overall accuracy of 82% on the dataset of 157 patients, demonstrating a promising capability for automated risk assessment.
Novelty: This predictive model can be utilised in the medical field to facilitate the early detection of heart disease or other risks. This model will soon be introduced in hospitals in the Tegal Regency and City area, Central Java.

Keywords: Coronary heart disease, Optimization, ANN algorithm, Extreme learning machine, Oversampling techniques
Received May 2025 / **Revised** June 2025 / **Accepted** July 2025

INTRODUCTION

The heart is a vital organ responsible for circulating oxygen and nutrients, and its dysfunction, often caused by atherosclerosis, can disrupt the entire body, commonly resulting in coronary heart disease [1]. Several recent studies [2]–[6] have suggested that coronary heart disease (CHD), mainly caused by atherosclerosis, is a leading global cause of death. Risk factors like hypertension, high cholesterol, and unhealthy lifestyles worsen the condition. CHD also affects patients with other illnesses, such as coarctation of the aorta. However, limited public awareness and complicated, costly diagnostic processes hinder early detection and treatment. Unhealthy lifestyle habits, such as smoking, poor diet, and lack of exercise, along with social and environmental factors, contribute to increased cardiovascular disease (CVD) risk. Adopting a healthy lifestyle is key to prevention. Early symptom recognition and addressing work-related stress and unhealthy habits are also essential for effective CHD prevention [7-10].

[*]Corresponding author.
Email addresses: aismaulana81@gmail.com (Iswanti)
DOI: 10.15294/sji.v12i2.24746

Artificial neural networks are a machine learning method with a layered structure consisting of neurons, connections, and weights. Neurons process input signals through weighted connections and activation functions to produce outputs. The network's layers—from input to output enable the processing of complex data such as classification or prediction [11-14]. The Extreme Learning Machine (ELM) is a fast and efficient algorithm for training Single Layer Feedforward Networks (SLFN). ELM stands out for its non-iterative learning process, enabling faster training without backpropagation. Neural networks, inspired by the human brain, consist of interconnected layers of neurons and weighted connections to process data. Machine learning, as a broader field of artificial intelligence, allows systems to learn from data and is widely applied across domains like healthcare, including tasks such as tumor detection via MRI classification [15-18].

Several recent studies [19-21] have demonstrated ELM's strength in fast, accurate predictions across domains. ELM is used to classify heart disease [19], detect diabetes [18], and forecast system errors [20], often outperforming traditional models. Its efficiency, low cost, and adaptability make it suitable for both medical and industrial applications. Vashist et al in 2022 compared AI models for crop monitoring and yield prediction. ELM achieved the highest accuracy (98.83%), outperforming CNN, LeNet, and CNN & PSO [22]. Artificial neural networks are a machine learning method with a layered structure consisting of neurons, connections, and weights [26]. Neurons process input signals through weighted connections and activation functions to produce outputs [27]. The network's layers from input to output enable the processing of complex data such as classification or prediction [28]. The Extreme Learning Machine (ELM) is a fast and efficient algorithm for training Single Layer Feedforward Networks (SLFN) [29]. ELM stands out for its non-iterative learning process, enabling faster training without backpropagation [30]. Neural networks, inspired by the human brain, consist of interconnected layers of neurons and weighted connections to process data [31]. Machine learning, as a broader field of artificial intelligence, allows systems to learn from data and is widely applied across domains like healthcare, including tasks such as tumor detection via MRI classification [32]. Given the need for efficient and accurate models in fields such as medical imaging, where large and complex datasets are common, ELM offers distinct advantages that complement general machine learning frameworks like scikit-learn**.** While scikit-learn provides a consistent interface for various machine learning algorithms, making method comparison easy and integrating well into diverse applications [23], [24], ELM's unique characteristics provide a powerful alternative, especially when computational speed and simplified training are critical. ELM has gained popularity for its speed and effectiveness, but its use in outlier detection is underexplored. This study highlights key research gaps, including limited evaluation for outlier detection, lack of analysis on imbalanced data handling, absence of ELM taxonomy, insufficient standard metrics, and the need for benchmark datasets [25].

Based on previous research, ELM has demonstrated its potential in various predictive applications, including health. With high computational speed and the ability to handle large data, ELM is applied to predict the risk of coronary heart disease (CHD) with high accuracy. However, a critical limitation in existing models for CHD risk prediction, particularly for earlier or less common stages (minority classes like CHD stages 2–4), is poor performance, often caused by data imbalance. To overcome this challenge and improve classification performance, especially in these high-risk categories, oversampling techniques such as Synthetic Minority Oversampling Technique (SMOTE) and Adaptive Synthetic Sampling (ADASYN) are applied. The main goal is to enable early identification and timely intervention for patients, thereby improving diagnosis, disease management, and reducing incidence and mortality rates. With high computational speed and the ability to handle large data, ELM is applied to predict the risk of coronary heart disease with high accuracy. The main goal is to enable early identification and timely intervention for patients, thereby improving diagnosis, disease management, and reducing incidence and mortality rates.

METHODS

This study implements an Extreme Learning Machine (ELM)-based heart disease prediction system using Python 3.10.0/3.12.0, MySQL, and key libraries (e.g., scikit-learn, pandas). The system classifies five stages: no heart disease (Stage 0) to Stage 4. Below is the structured methodology.

The provided flowchart outlines the general steps of data processing and model implementation. To address the critical issue of data imbalance, particularly for the minority classes (Coronary Heart Disease stages 2–4), this research incorporates oversampling techniques. This approach is essential to improve the classification performance and enhance the model's generalization capability for real-world implementation, especially for high-risk categories. The workflow of ELM-based CHD risk prediction can be shown in Figure 1.

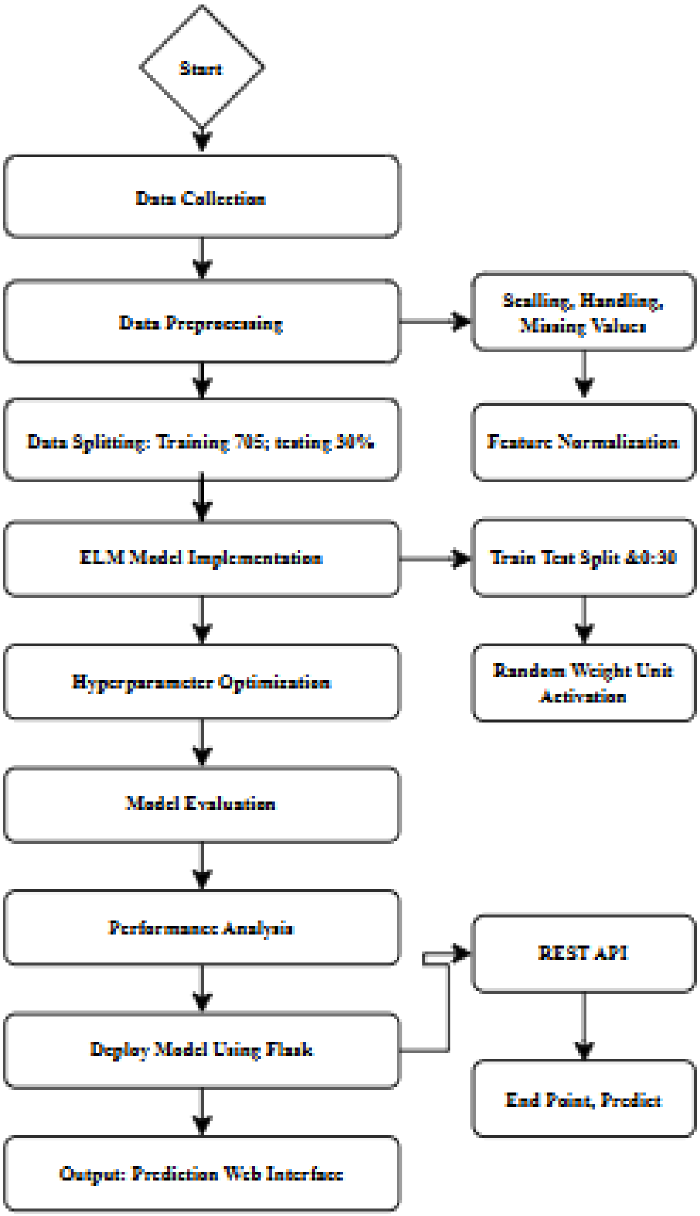

Figure 1. ELM-based CHD risk prediction workflow

Data Collection

This study uses primary data obtained from medical records of patients diagnosed with coronary heart disease at Dr. Soeselo Regional Hospital, Tegal Regency, Central Java, during 2023–2024. The dataset includes 443 patient records and features such as gender, age, height, weight, blood pressure, and diabetes status. The risk classification output consists of five categories: no heart disease, stage 1, stage 2, stage 3, and stage 4 were used and stored in .csv format. Here is Table 1 that describes the variables, their possible values, and data types based on the provided dataset.

Table 1. Data Description

Variable Name	Description	Data Type
Age	Age of the individual (in years)	Integer
Gender	Gender of the individual (encoded) 1 = Male, 2 = Female	Integer
Level of Chart Plan, Dataset Activity	Activity level related to a chart plan	Integer
Blood pressure	Systolic blood pressure (mmHg)	Integer
Cholesterol	Total cholesterol level (mg/dL)	Integer
Glucose	Fasting glucose status (binary)	Integer
ECG Test	Result of an exercise ECG test (ordinal)	Integer
Heart Rate	Resting heart rate (beats per minute)	Integer
Chart Plan Damage Activity	Indicates presence of damage (binary). 0 = No, 1 = Yes	Integer
Attributes of Heart Disease Problems	Severity/classification of heart disease (ordinal). 1 (Low) to 4 (High)	Integer

Data Preprocessing

Initial preprocessing steps included handling missing values, converting data into a matrix format, and feature scaling. Feature normalization was performed using the Standard Scaler method to ensure that each feature has a distribution with a mean of 0 and a variance of 1 [12], [23], [24].

Data Splitting

The dataset was divided into training and testing subsets using a 70%:30% split, resulting in 310 training samples and 133 testing samples. This ratio was chosen based on common practices in machine learning research, which recommend allocating a larger portion of the data for training to ensure sufficient model learning, while preserving an adequate portion for evaluating generalization performance. The train_test_split function from Scikit-learn was used, with a random_state set to 42 to maintain reproducibility. The 70:30 split offers a balanced trade-off between training and testing sizes, especially for datasets of moderate size, such as the one used in this study (443 samples). A smaller training size could lead to underfitting, while a smaller test size might not provide a reliable assessment of model performance.

Modelling

This study employs the Extreme Learning Machine (ELM) algorithm as the main classifier. A key characteristic of ELM that sets it apart from traditional neural networks is its non-iterative learning process. Unlike backpropagation, which iteratively adjusts weights, ELM randomly assigns the hidden layer weights and biases, and then the output weights (connecting the hidden layer to the output layer) are analytically determined. This analytical determination significantly contributes to its rapid training capability.
The ELM architecture consists of:

- Input Layer: This layer receives the patient feature inputs, which are the preprocessed data attributes like age, gender, blood pressure, cholesterol, etc.
- Hidden Layer: This layer contains 100 neurons. A key characteristic of ELM is that the weights connecting the input layer to the hidden layer, as well as the biases for the hidden layer neurons, are randomly assigned. The activation function used for these hidden neurons is the sigmoid function. While the text initially mentions a "sign activation function", it immediately clarifies that "sigmoid or ReLU are also commonly used", and the code snippet in Figure 5 explicitly shows 'sigm' (sigmoid) being used.
- Output Layer: Five class outputs (stage 0–4).
  This layer produces five class outputs, corresponding to the five stages of heart disease (Stage 0–4). The model was implemented using Python 3.8.5 and the Python libraries Flask and Visual Studio Code version 1.99.3 (user setup).

The model is trained using preprocessed data, which includes normalized features and one-hot encoded labels. There is the flow of the ELM Algorithm Process [26]: (1) Input Data Preparation: Standardize or normalize the input features and convert target labels into a suitable format, such as one-hot encoding for multi-class classification; (2) Random Initialization of Hidden Layer Parameters: Randomly assign the input weights (connecting the input layer to the hidden layer) and the biases for each neuron in the hidden layer; (3) Hidden Layer Output Calculation: Compute the output of the hidden layer (often denoted as H) for the given input data. This involves applying the chosen activation function (sigmoid in your case) to the weighted sum of inputs plus biases for each hidden neuron; (4) Output Weight Calculation: Analytically determine the output weights (often denoted as β) that link the hidden layer to the output layer. This is typically achieved using the Moore-Penrose generalized inverse of the hidden layer output matrix (H) and the target output matrix (T). The common formula is $\beta = H^{\dagger}T$, where $H^{\dagger}$ is the Moore-Penrose generalized inverse of H. (5) Prediction: For new, unseen input data, the trained ELM model uses the previously randomly assigned input weights and biases, the activation function, and the analytically calculated output weights to generate predictions.

Evaluation Model

Model performance was evaluated using the test dataset, measuring accuracy, precision, recall, and analyzing the confusion matrix. Accuracy shows the overall correctness of the model's predictions, while precision shows how many of the predicted positive cases were correct. In addition, recall shows the model's ability to identify all actual positive cases. The confusion matrix below presents a summary of a classification model's performance on a given dataset. This matrix illustrates the comparison between the actual values (true classes) and the predicted values (classes predicted by the model) for each existing class.

Table 2. Confusion matrix

Actual Class	**Predicted Class**	
	Positive	**Negative**
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)

Implementation

This research implements the Extreme Learning Machine (ELM) method to classify the risk of heart disease using a dataset with 521 samples. The process was carried out in the following stages: input data. The data entered for testing includes age, gender, chest pain experienced, blood pressure, cholesterol level, blood sugar, ECG results, and heart rate. After entering the data such as age, gender, chest pain, blood pressure, cholesterol level, blood sugar level, ECG results, heart rate, chest pain during activity, and the heart disease prediction is processed, the results will show a classification according to the prediction.

The results of this study are in the form of aspect-based Extreme Learning Machine (ELM) predictions, so that the category aspects of each heart disease risk prediction are obtained using the Python programming language. The Python software that will be used is Python version 3.8.5, and the Python, Flask, and Visual Studio Code libraries version 1.99.3 (user setup). ELM was chosen because it has several advantages in prediction, namely high training speed (the training process is based on an analytical algorithm that directly calculates the output weight without requiring long iterations such as in neural networks), a simple and efficient ELM structure (only has one hidden layer and the use of random weights makes it simpler and easier to implement, and minimizes the need for complex parameter tuning).

RESULTS AND DISCUSSIONS

The patient data consists of 521 individuals, with a distribution percentage of 70% for the training data, which amounts to 365 patients, and 30% for the test data, which amounts to 156 patients. There is a description of the characteristics of chest pain during activity. It's shown in Table 3.

Table 3. Characteristics of chest pain during activity

Level	Description of Characteristics of Chest Pain During Activity
1	Chest pain due to emotional or physical stress
2	Feeling uncomfortable in the chest, or not severe
3	Feels uncomfortable near the breastbone
4	None like 1, 2, and 3

Description of Blood Pressure is shown in Table 4.

Table 4. Description of Blood Pressure

Variable	Description	Unit
Blood pressure	Blood Pressure	mmHg
Cholesterol level	Cholesterol Level	mm/dl
Blood sugar	Blood sugar >120, (0 = no, and 1 = yes)	mg/dl
ECG	ECG test results	mg/dl

Description of ECG test results is shown in Table 5.

Table 5. Description of ECG test results

ECG Test	Description
0	Normal
1	Not Normal
2	Left ventricle enlargement occurred.

Description of ECG test results is shown in Table 6.

Table 6. Description of chest pain

Variable	Description
Heart rate	Maximum heart rate achieved during the ECG test
Chest pain during activity	Chest pain during activities, 0 = no; 1=Yes

The attributes of heart disease prediction results are shown in Table 7.

Table 7. Attributes of heart disease prediction results

Level	Description of Characteristics of Chest Pain During Activity.
0	Does not suffer from heart disease
1	Suffering from stage 1 heart disease
2	Suffering from stage 2 heart disease
3	Suffering from stage 3 heart disease
4	Suffering from stage 4 heart disease

Table 8 summarizes the diagnostic criteria for each heart disease stage (Stage 1–4), including clinical thresholds that distinguish them from the no heart disease category (Stage 0).

Table 8. Heart disease criteria

Heart Disease Class	Criteria
Stadium I	Patients with heart disease but no restrictions on physical activity. Ordinary physical activity does not cause excessive fatigue, palpitations, dyspnoea, or angina pain.
Stadium II	Patients with heart disease have minimal physical activity restrictions. Feels comfortable at rest. The result of normal activities causes physical fatigue, palpitations, dyspnoea, or angina.
Stadium III	Patients with heart disease who have physical activity restrictions. Feels comfortable at rest. Light physical activity causes fatigue, palpitations, dyspnoea, or angina.
Stadium IV	Patients with heart disease are unable to perform any physical activity without discomfort. Heart failure symptoms can appear even at rest. The complaint worsens during activities.

The features are normalized using Standard Scaler to ensure a mean distribution of 0 and a variance of 1.

```
scaler = StandardScaler().fit(X_train)
X_train_scaled = scaler.transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

Figure 2. Standard scaler

The features are normalized using Standard Scaler to ensure a mean distribution of 0 and a variance of 1 (Figure 2). This standardization method was chosen to mitigate the impact of varying feature scales on the ELM model's performance, particularly given the heterogeneous nature of clinical data (e.g., blood pressure in mmHg vs. cholesterol in mg/dL). By centering the distribution around zero with unit variance, Standard Scaler ensures that no single feature dominates the model's learning process due to its inherent scale [12]. Categorical data in the target variable was converted into a numerical format using one-hot encoding via the One Hot Encoder from scikit-learn. Each category is represented as a binary vector, with the parameter sparse, output=False, ensuring the output is a dense array. This transformation ensures compatibility with the machine learning algorithms employed in the experiment. Preprocessing (Changing the target labels into a binary numerical form (one-hot) so that the ELM model can use them for classification). Then, class labels are converted to one-hot encoding using One Hot Encoder.

ELM Model:

```
elm = ELM(X_train_scaled.shape[1], y_train_encoded.shape[1], classification="c")
elm.add_neurons(100, "sigm")  # Jumlah neuron tersembunyi & fungsi aktivasi
elm.train(X_train_scaled, y_train_encoded)
```

Figure 3. Model ELM

Figure 3 illustrates the implementation of an Extreme Learning Machine (ELM) model for classification tasks. The model is initialized with input dimensions matching the number of features in the normalized training data (X train, scaled, shape 1) and output dimensions corresponding to the number of classes after one-hot encoding (y train, encoded, shape 1). The parameter classification='c' indicates that the model is designed for classification problems.

The ELM architecture consists of a single hidden layer with 100 neurons and a *sigmoid* activation function, which produces binary outputs (−1 or +1). While the *sign* activation function introduces non-linearity, other alternatives such as *sigmoid* or *ReLU* are also commonly used. The model is trained on preprocessing data, including normalized features (X train, scaled) and encoded labels (y train encoded). A key characteristic of ELM is its use of randomly assigned hidden layer weights and analytically determined output weights, enabling rapid training.

Evaluation Model

Figure 6 presents the evaluation results of the Extreme Learning Machine (ELM) model's performance on test data. The model achieved an accuracy score of 82,17% (0.82165) when predicting classes from the scaled test features (X test scaled). Predictions were obtained by taking the argmax of the output to convert the one-hot encoded results back to class labels. While the accuracy provides an overall measure of correctness, the classification report (though misspelled as "Classification report" in the code) would typically reveal more detailed metrics, including precision, recall, and F1-score for each class (presumably stages 0-4 based on the context).

```
rasi: 0.821656050955414

ssification Report:
              precision    recall  f1-score   support

           0       0.87      1.00      0.93        27
           1       0.83      0.74      0.78        27
           2       0.76      0.84      0.80        31
           3       0.80      0.74      0.77        38
           4       0.85      0.82      0.84        34

    accuracy                           0.82       157
   nacro avg       0.82      0.83      0.82       157
  ghted avg       0.82      0.82      0.82       157
```

Figure 4. Model performance for each class

The moderate accuracy of 82% suggests the model has learned some discriminative patterns but still exhibits significant room for improvement in generalization capability. The mention of stages 0-4 in the results implies this is likely a multi-class medical classification task, where understanding the model's performance per class becomes particularly important to identify potential biases or specific classes that are more challenging to predict.

Dataset Analysis

- **Class Distribution:**

  Accuracy: 0.82 (82%)

  This indicates that the model correctly classified 82% of all instances in the dataset. Total number of samples (support) in the dataset is 157.

  The performance for each class (0, 1, 2, 3, 4):

  - **Class 0**

    Precision: 0.87; Recall: 1.00(The model successfully identified 100% of all actual Class 0 instances. This is perfect recall. F1-score: 0.93 - A high F1-score indicates excellent balance between precision and recall for this class. Support: 27 - There are 27 actual instances of Class 0 in the dataset. The model performs exceptionally well for Class 0. It rarely makes false positive predictions for Class 0, and it correctly identifies every single instance of Class 0.
  - **Class 1**

    Precision: 0.83; Recall: 0.74 - The model only captured 74% of all actual Class 1 instances. This means 26% of actual Class 1 instances were missed (false negatives). F1-score: 0.78 - A decent F1-score, but lower than Class 0, indicating room for improvement, particularly due to the lower recall. Support: 27 - There are 27 actual instances of Class 1. The model has good precision for Class 1, but its recall is relatively lower. This suggests that the model is somewhat conservative in predicting Class 1, and when it does predict, it's usually right. However, it fails to identify a significant portion of true Class 1 instances.
  - **Class 2**

    Precision: 0.76; Recall: 0.84 - The model identified 84% of all actual Class 2 instances; F1-score: 0.80 - A good F1-score; Support: 31 - There are 31 actual instances of Class 2. The model performs reasonably well for Class 2. It has higher recall than precision, meaning it's good at finding true Class 2 instances, but it occasionally mislabels other classes as Class 2.
  - **Class 3**

    Precision: 0.80; Recall: 0.74 - Similar to Class 1, the model only identified 74% of all actual Class 3 instances, missing 26%.; F1-score: 0.77 - This is the lowest F1-score among all classes, indicating this is the most challenging class for the model; Support: 38 - This is the largest class with 38 actual instances. Performance for Class 3 is a weakness. While its precision is acceptable, its low recall means a significant number of actual Class 3 instances are being missed or misclassified into other categories (as seen in the confusion matrix analysis, where it was confused with Class 2 and Class 4).
  - **Class 4**

    Precision: 0.85; Recall: 0.82 - The model identified 82% of all actual Class 4 instances; F1-score: 0.84 - A strong F1-score, indicating good overall performance for this class; Support: 34 - There are 34 actual instances of Class 4. The model performs well for Class 4, demonstrating a good balance of precision and recall.

Confusion Matrix Analysis

The prediction results are shown in Table 8.

Table 8. Confusion matrix of the classification model

Actual	Prediction					
	Class	0	1	2	3	4
	0	27	0	0	0	0
	1	4	20	3	0	0
	2	0	4	26	1	0
	3	0	0	5	28	5
	4	0	0	0	6	28

Based on the test results, we can now compute the primary performance indicators for our classification model.

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \tag{1}$$

Calculation:
1. Total Correct Predictions (sum of TPs): 27+20+26+28+28=129.
2. Total Predictions (sum of all values in the matrix): (27) + (4+20+3)+(4+26+1)+(5+28+5)+(6+28)=157
3. Accuracy: Accuracy=157129≈0.8217

The result of the calculation for the model's overall accuracy is 82.17%

Per-Class Performance Metrics

There are calculations to calculate the precision, recall, and F1-score for each class. Formula:

$$Precission = \frac{(TP)}{(TP + FP)} \tag{2}$$

$$Recall = \frac{(TP)}{(TP+FN)} \tag{3}$$

$$F - 1\,Score = \frac{(2*Recall*Precission)}{(Recall+Precission)} \tag{4}$$

Table 9. Cass performance metrics

Class	Precision	Recall	F1-Score
0	0.8710	1	0.9310
1	0.8333	0.7407	0.7843
2	0.7647	0.8387	0.8000
3	0.8000	0.7368	0.7671
4	0.8485	0.8235	0.8358

Based on the confusion matrix analysis, the model was evaluated using a total of 157 data samples whose distribution was relatively balanced across the five classes. This balance is a good condition for model training, although Class 3 has slightly more samples than the other classes. Overall, the model's performance shows very promising results, especially in Class 0, which was perfectly identified without a single error (True Positive = 27, False Positive = 0, False Negative = 0). This indicates that the features for Class 0 are very distinctive and easily recognized by the model.

Further analysis shows that there is a certain level of error between classes. Class 3 is the most challenging class for the model, with 10 samples misclassified as Class 2 or Class 4. This indicates overlapping features or similar characteristics between the three classes. Likewise, the model still has a little difficulty in distinguishing between Class 1 and Class 2, which are interchanged. Meanwhile, Class 4 performs quite well, with the majority of errors occurring due to misprediction, as in Class 3. These findings provide important insights for future model improvements, especially in improving the model's ability to distinguish between adjacent classes.

IMPLEMENTATION of ELM

This research implements the Extreme Learning Machine (ELM) method to classify the risk of heart disease using a dataset with 521 samples. The process was carried out in the following stages. This is shown in Figure 5.

Figure 5. The first view of the input heart disease risk prediction

Prediction input page. This input is used to classify the level of risk for heart disease. The data entered for testing includes age, gender, chest pain experienced, blood pressure, cholesterol level, blood sugar, ECG results, and heart rate.

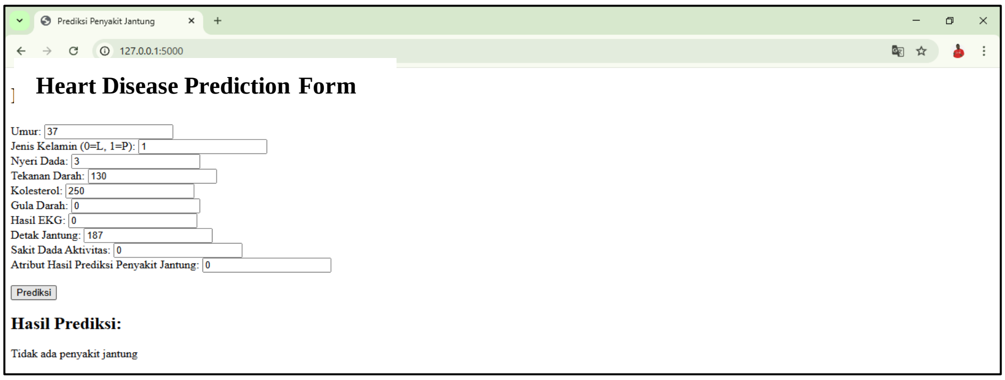

Figure 6. The result display shows no heart disease

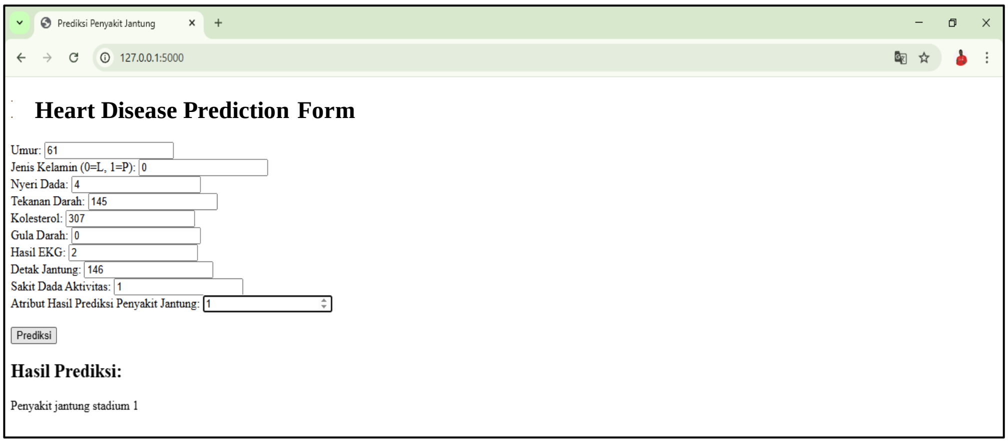

Figure 7. The result display shows heart disease stage 1

On the test result page, after the data is filled in by the patient, the results will immediately appear, indicating whether or not the patient has heart disease.

Table 10. The Comparison of this study with existing research

Study	Algorithm	Result
Diagnosis of heart disease (2023)	ELM & PSO	Accuracy = 57,2%
Forecast soil fertility rank (2024)	ELM	Accuracy = 84%
Diabetes Prediction (2022)	ELM	Accuracy = 97%
This study	ELM	Accuracy = 82%

This study achieved an accuracy of 82% using the ELM algorithm. When compared to existing research, this result demonstrates a competitive performance, surpassing the 57.2% accuracy reported for heart disease diagnosis (2023) and positioned reasonably close to the 84% accuracy for soil fertility rank forecasting (2024), both also utilizing ELM or ELM-based methods. While not reaching the high accuracy of 97% observed in the 2022 Diabetes Prediction study, the 82% accuracy indicates the effectiveness of the proposed approach within its specific domain and dataset. This comparison underscores the model's capability and provides context for its contribution to the field.

CONCLUSION

The research successfully developed and evaluated an Extreme Learning Machine (ELM) algorithm for Coronary Heart Disease (CHD) risk prediction using patient data from Dr. Soeselo Hospital. The model achieved an overall accuracy of 82% on the dataset of 157 patients, demonstrating a promising capability for automated risk assessment. Detailed analysis of the classification report reveals varying performance across the five identified risk classes (0-4): (1) The model exhibits exceptional performance for Class 0 (likely representing "no risk" or "very low risk"), achieving perfect recall (1.00) and high precision (0.87), indicating that it reliably identifies healthy or low-risk individuals without significant misclassifications. (2) For Class 4, the model also shows strong performance with a high F1-score of 0.84, suggesting good reliability in identifying the highest risk group. (3) However, the model demonstrates notable weaknesses in predicting Class 1 and Class 3, both having the lowest recall rates (0.74). This implies that a significant portion (26%) of actual patients belonging to these risk categories are being missed or misclassified into other groups. Specifically, Class 3 has the lowest F1-score (0.77), indicating it is the most challenging class for the model to predict accurately. The confusion matrix further highlights this, showing that Class 3 instances are frequently misclassified as Class 2 or Class 4, and Class 1 instances are confused with Class 0 and Class 2. For future research, suggested to given the critical implications of missing high-risk patients in medical diagnosis (false negatives), efforts should specifically target improving the recall for these underperforming classes.

REFERENCES

[1] Junaid, M.J.A, & Kumar, R., 2020., Data Science and Its Application in Heart Disease Prediction, Proceedings of International Conference on Intelligent Engineering and Management, ICIEM 2020, 396-400.[2] Xie et al., 2025., Relationship between older coronary heart disease patients' phase II cardiac rehabilitation intentions, illness perceptions, and family caregivers' illness perceptions. Heart and Lung, 70, 183–190.

[3] Xieet al.,, 2025., Relationship between older coronary heart disease patients' phase II cardiac rehabilitation intentions, illness perceptions, and family caregivers'illness perceptions. Heart & Lung. Vol 70. Hal 183-190.

[4] Mai et al., 2018., Non Laboratory-Based Risk Factors for Automated Heart Disease Detection. International Symposium on Medical Information and Communication Technology, ISMICT, 2018-March, 1-6.

[5] Lee et al., 2025., Coarctation of the aorta and accelerated atherosclerosis: A contemporary review on the burden of atherosclerotic cardiovascular disease, International Journal of Cardiology Congenital Heart Disease, Vol 19-100561.

[6] Mori et al., 2020., Impact of Tai ChiYuttari-exercise on arteriosclerosis and physical function in order people. Archives of Gerontology and Geriatrics, 87(January), 104011.

[7] Abufaraj et al., 2020., The impact of hormones and Reproductive factors on the risk of bladder cancer in women: Results from the Nurses' Health Study and Nurses' Health Study II, International Journal of Epidemiology, 49(2), 599-607.

[8] Jabri et al., 2021., Meta-analysis of effect of vegetarian diet on ishemic heart disease and all-cause mortality . American Journal of Preventive Cardiology, 7, 100182.
[9] Naomi et al., 2021., Faktor Risiko Kejadian Penyakit Jantung Koroner (Studi Kasus di RSUD Prof. Dr. W. Z. Johannes Kupang). Media Kesehatan Masyarakat. Vol 3, No 1, 2021, 99-107.
[10] Roth et al., 2020., Global Burden of Cardiovascular Diseases and Risk Factors, 1990–2019. Journal Of The American College Of Cardiology. VOL. 76, NO. 25
[11] J. Wang et al., 2022., A review on extreme learning machine, Multimedia Tools and Applications, 81:41611–41660.
[12] Sathyanarayanan, 2024., Confusion Matrix-Based Performance Evaluation Metrics. Journal of Biomedical Research African. Res. Vol. 27, No.4s
[13] Choi et al., 2020., Introduction to Machine Learning, Neural Networks, and Deep Learning. Trans Vis Sci Tech. 2020;9(2):14.
[14] Khan et al., 2021., Prediction of COVID-19 - Pneumonia based on Selected Deep Features and One Class Kernel Extreme Learning Machine, Computers and Electrical Engineering, Vol. 90-106960.
[15] Praiss et al., 2020., Using Machine Learning to Create Prognostic Systems for Endometrial Cancer. Gynecologic Oncology, Vol 159 (3), 744 – 750.
[16] Saberi-Movahed et al., 2020., Receiving More Accurate Predictions for Longitudinal Dispersion Coefficients in Water Pipelines: Training Group Method of Data Handling Using Extreme Learning Machine Conceptions. Water Resources Management. 34:529–561.
[17] Elsayed et al., 2022., Early Stage Diabetes Prediction via Extreme Learning Machine. University of Cincinnati, Cincinnati, Ohio. arXiv:2202.11216v1
[18] Ariyanti et al., 2023., Application of Extreme Learning Machine Method With Particle Swarm Optimization to Classify of Heart Disease. IJCCS (Indonesian Journal of Computing and Cybernetics Systems). Vol.. 17, No.3.
[19] Qi & Zhang, 2020., Data-driven fault diagnosis and prognosis for process faults using principal component analysis and extreme learning machine. Conference on Industrial Informatics (INDIN) | 978-1-7281-4964-6/20.
[20] Kiani et al., 2024., Survey on extreme learning machines for outlier detection. Machine Learning. 113:5495–5531.
[21] Vashisht et al., 2022., Improvised Extreme Learning Machine for Crop Yield Prediction, International Conference on Intelligent Engineering and Management (ICIEM). 978-1 6654-6756.
[22] Pedregosa et al., 2011., Scikit-learn: Machine Learning in Python. JMLR 12, pp. 2825–2830.
[23] Raschka et al., 2020., Machine learning in Python: Main developments and technology trends in data science, machine learning, and artificial intelligence. MDPI Journal. 11.193.
[24] Pant et al., 2024., A Data-Driven Approach to Soil Fertility Assessment Using Extreme Learning Machine, International Conference on Innovations in Communications, Electrical and Computer Engineering (ICICEC) | 979-8-3503-7651-7/24/$31.00 ©2024 IEEE.
[25] Huang et al., 2006; Extreme learning machine: Theory and applications. Neurocomputing, 70(1-3), 489-501.