Date of publication DEC-31, 2023, date of current version MAY-24, 2023.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.22.4.3354

A Weighted Majority Voting Ensemble Model for Disease Prediction Boosted by PSO: The Case of Type 2 Diabetes

MULJONO[1], NOVIAN ADHIPURNA[2]

[1] Department of Informatics Engineering, Universitas Dian Nuswantoro, Semarang, Indonesia
[2] Department of Pharmacy, Universitas Borneo Lestari, Kalimantan, Indonesia

Corresponding author: Muljono (e-mail: muljono@dsn.dinus.ac.id).

"Authors would like to thank Universitas Dian Nuswantoro for the continuous support in completing this research."

ABSTRACT Early detection of diabetes is critical to reducing the number of cases, which continues to rise year after year. Many approaches to diagnosis have been used, but they still have flaws in making clinical decisions that are more effective and efficient. The use of intelligent systems is very effective in assisting in data analysis and clinical decision support. The purpose of this article is to develop a model to predict diabetes mellitus using the Pima Indian Diabetes Dataset (PIDD). The ensemble method has shown to be quite effective at increasing accuracy, but it has the issue of determining the optimal weight. As a result, to improve prediction accuracy, this study employs PSO optimization in the selection of ensemble majority voting weights. The test results show that predictions for ensemble majoritarian voting using PSO perform well, with an accuracy rate of 0.9322, precision of 0.9412, recall of 0.8421, and F1-score of 0.8889. PSO accuracy is improved by 4% and 7%, respectively. This demonstrates that applying PSO to the ensemble model can improve accuracy.

KEYWORDS Diabetes; Machine learning; Ensemble majority voting; PSO.

I. INTRODUCTION

A non-communicable chronic disease with a significant death rate is diabetes mellitus (DM) [1]. Both low- and middle-income countries as well as numerous high-income countries frequently have this disease. High blood sugar levels cause symptoms that are unique to DM. The pancreas organ produces the hormone insulin, which controls the amount of sugar in the blood [2]. Patients with diabetes cannot create insulin, which is necessary for the body to convert glucose into energy [3].

Type 1 diabetes is characterized by the loss of beta cells in the pancreas, which create insulin, whereas type 2 diabetes is characterized by insulin resistance [4]. Although the body is still able to manufacture insulin, it cannot be utilized properly [5]. The severity of this illness increases the possibility of major side effects like heart attack, stroke, renal failure, vision loss, leg amputation, nerve damage, and others [6]. From year to year, there are more and more cases of diabetes. Type 2 diabetes accounts for 90% of all diabetes cases worldwide. In 2019, there were 463 million cases of diabetes worldwide. By 2045, that number will have increased by 51% to 700 million cases [7].

As a result, early detection of diabetes is critical to reducing the number of cases, which continues to rise year after year. Many approaches to diagnosis have been used, but they still have flaws in making clinical decisions that are more effective and efficient. Many factors can make it difficult for medical staff to make a diagnosis in the health sector. Diabetes diagnosis relies heavily on medical data and decision-support systems. The use of intelligent systems, such as the machine learning approach, is very effective in assisting in data analysis and clinical decision support.

Machine learning-based diabetes detection methods have been proposed in previous literature publications in recent years. An intelligent e-healthcare system for diabetes prediction uses the Decision Tree algorithm [8]. The main determinants of diabetic complications in patients were identified based on selected features using prediction and comparative testing of models such as J48, NB, RF, and LR [9] and [10]. Due to its low computing complexity, LGBM is applied to the early identification of diabetes [11]. Faster computations are possible thanks to the parallel processing capabilities of the XG-Boost algorithm [12]. In [13], where the authors implemented the LR, SVM, KNN, RF, NB, and GB models, supervised machine learning performance measurement employed six prediction models. According to the experimental findings, RF has a higher accuracy and ROC

than other models (86.76% and 86.28%, respectively). The attribute coefficient of glucose has a significant impact on the diagnosis of type 2 diabetes, according to the findings of the correlation analysis. The authors of [9] examined the application of several K-fold algorithms to enhance classifier performance. The implemented classifier models are J48, NB, RF, and LR. The measurement findings employed K-fold 10 CV to prevent overfitting, and the LR of 0.77 has the maximum accuracy. In [10], the idea of anticipating diabetic problems was put out. Through feature selection, the author determined the key traits of diabetes complications. The implementations of the comparison models include LR, SVM, DT (CART), RF, ADB, and XGB. Grid search with 10-fold CV validation and hyperparameter tuning according to the experimental findings, XG-Boost had the greatest accuracy rate for diabetic foot problems, at 97.8%, out of the eight diabetic issues. The use of ensemble stacking techniques to increase accuracy has been suggested in [14]. Preprocessing like feature selection, normalization, and the removal of outliers and imputations was the subject of experiments. ADB, SVM, and MLP are the components of the ensemble stacking model. According to the testing findings, the stacking ratio outperforms ADB by 1.66%. In [15], we used ensemble soft voting to apply the meta-classifier model to the prediction model. The suggested ensemble approach integrates the RF, LR, and NB algorithms. AdaBoost, LR, SVM, RF, NB, Bagging, GradientBoost, XGBoost, and CatBoost are compared to the ensemble model. The results show that Ensemble's soft voting performs best, with accuracy, precision, recall, and F1 scores of 79.04%, 73.48%, 71.45%, and 80.6%, respectively.

Although many prediction models have been used in recent years, low accuracy is still a common problem [8]. When compared to using a single classifier, the ensemble technique improves accuracy; however, finding the optimal ensemble weight in the ensemble model is difficult, so the prediction results are less accurate [16]. Particle Swarm Optimization (PSO) is an effective optimization method for multidimensional and multiparameter problems. When it comes to determining classifier weights, PSO has the best fitness [17]. This study proposes using PSO to calculate the weight of the ensemble majority vote from a combination of models such as LR, GBM, XGB, and LGBM.

The study's remaining sections are structured as follows: The dataset, the detailed descriptions of the suggested approaches, and the evaluation metrics are provided in Section 2. Section 3 includes an explanation of the experiment's findings and discussion. A summary and recommendations based on the experimental findings are provided in Section 4.

II. MATERIALS AND METHODS

This section describes several subsections, including dataset information, proposed pre-processing methods, predictive models, and optimization.

A. DATASET

The National Institute of Diabetes and Digestive and Kidney Diseases dataset, which is part of the Pima Indian Diabetes Database (PIDD) and is openly accessible, was used as a secondary dataset in this investigation. This information revealed that samples were drawn from the female population of the Phoenix area, namely the state capital of Arizona in the southwest of the United States, which comprised up to 768 individuals. It was confirmed that 258 persons out of the whole sample had been tested as positive for diabetes mellitus and 500 had been tested as negative. As a result of the diagnosis, eight more traits in addition to the target class were also discovered from the examination findings.

Table 1. List of Pima Indian dataset attributes used to predict diabetes

Attribute	Description
Pregnancies	Indicator of the pregnancy rate
OGTT (Oral Glucose Tolerance Test).	An oral glucose tolerance test's two-hour plasma glucose concentration (mg/dl)
Blood Pressure.	diastolic blood pressure (mmHg)
Skin Thickness	Triceps skinfold thickness (mm)
Insulin	Two-hour serum insulin (mu U/ml)
BMI (Body Mass Index)	Body mass index (kg/m^2)
Age	Patient age
Pedigree Diabetes Function	a function that analyzes family history to determine the history of diabetes

B. PROPOSED METHODS

This study suggests an ensemble technique for classifying data by maximizing the choice of the appropriate weight value in the ensemble majority vote, where the PSO algorithm is used in the process to get a more optimal value. Fig. 1 depicts the whole flowchart. Datasets for training and datasets for testing are created from the used dataset. The training data is used to develop several n classifier models $\{C_1, C_2, \ldots, C_n\}$, each of which weights W relative to the other classifiers $\{W_1, W_2, \ldots, W_n\}$, where n = 1, 2,... n. Aggregation is used to combine the classifiers into a single entity, which is then given a weight for each classification to create predictions based on the probability of voting. The PSO approach is used to optimize the ensemble weights, producing the ideal weight values in the process. The stages of the PSO workflow can be seen in Fig. 2.

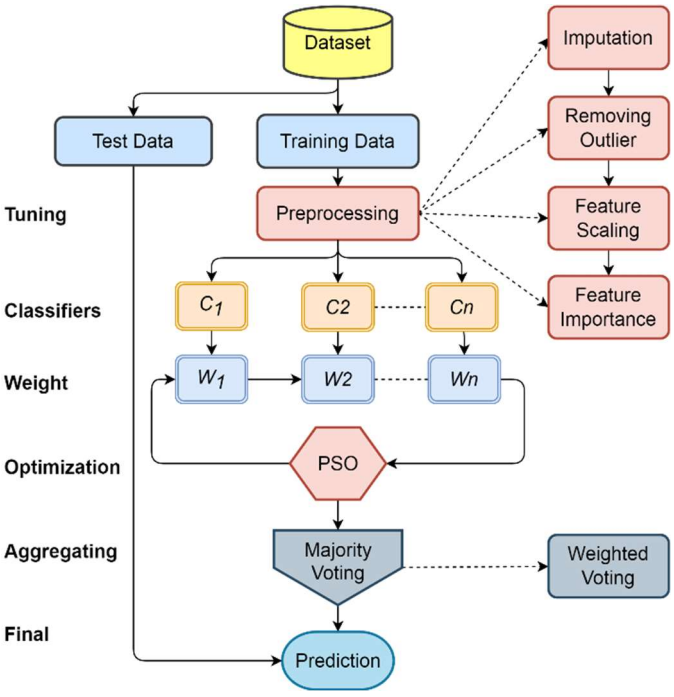

Figure 1. EMVPSO architecture

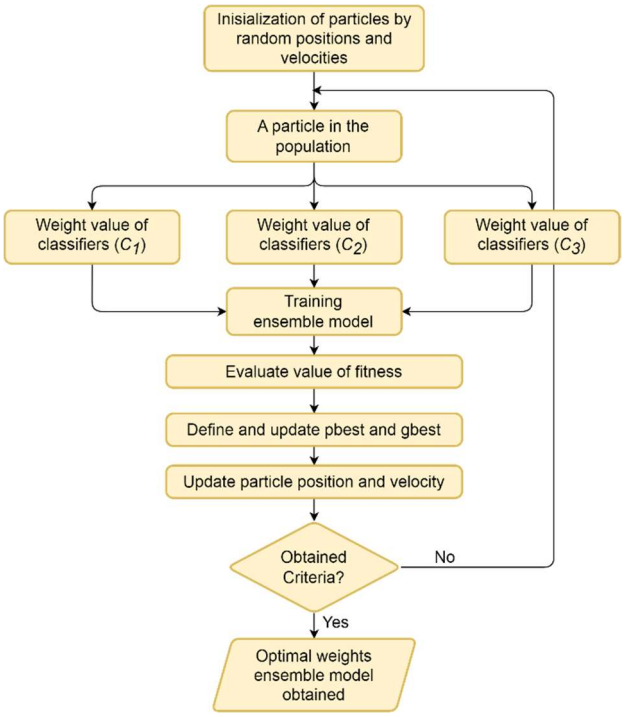

Figure 2. PSO workflow

The following are the steps for the PSO flow process:

1. Initialize the random particle population's position and velocity in the search dimension.
2. Evaluate the fitness of the classifier model $\{C_1, C_2, \ldots, C_n\}$ based on the position of each particle.
3. Determine the particle's best fitness and set it as G-best. For each particle, the initial position will be equal to the initial P-best.
4. Continue until the stopping criteria are satisfied.

If P-best and G-best already exist or are not in the initial position:

a. Update each particle's velocity using Eq. 10. Using Eq. 11, update each particle's position based on the new velocity value.
b. Determine each particle's fitness.
c. Determine the particle's best fitness and set it as G-best. Compare the current position of each particle to the previous P-best position.
d. Examine the stopping criteria or maximum iteration; if they are met, stop; otherwise, return to point (a).

Show the optimal weight based on the best ensemble-weighted majority voting accuracy

C. PREPROCESSING

C.1 IMPUTATION

By substituting replacement values for missing values, imputation is a strategy for maintaining the majority of the data or information from a data collection. This method is employed because it can result in a significant reduction in the size of the data collection and is not always preferable to remove data from a data set. It has the potential to bias the data set and produce inaccurate analytical results. The median property was used in this study to implement the imputation procedure [18].

$$I(x) = \begin{cases} median(x), & if\ x = null/missed \\ x, & otherwise \end{cases}, \quad (1)$$

where x is a dimensional space-based instance of a feature vector. This technique works well with numerical variables. The median of the variable is used to replace every instance of missing or unrepresented values (NaN) in this technique.

C.2 REMOVING OUTLIER

Outliers are numerical observation distances from the remaining data or values that fall outside the data or sample pattern range. Outlier observations are required because they can lead to inaccurate estimations. The interquartile range (IQR) method was used as an outlier removal technique in this study. The IQR method determines the limits of the data by using the upper and lower limit values on the data. The range of values between 1.5 x IQR and 1.5 x IQR is designated as an outlier and deleted.

$$O(x) = \begin{cases} x, & if\ Q_{low} \leq x \leq Q_{up} \\ drop, & otherwise \end{cases}, \quad (2)$$

where x is an instance of a feature vector in a dimensional space.

1. The first quartile, or the 25th percentile, of the total data is represented by Q1.
2. The second quartile, median, or 50th percentile of the amount of data is represented by Q2.
3. The third quartile, or 75th percentile, of the total data is represented by Q3.

(Q1 – 1.5 x IQR) represents the data set lower limit Q_{low}, and (Q3 + 1.5 x IQR) represents the upper limit Q_{up}.

C.3 FEATURE SCALING

Making numerical data in datasets that have the same range of values is possible with feature scaling (scale). Data variables that are thought to predominate over other factors should be adjusted. In general, scaling is independent of tree-based models, and vice versa, scaling is independent of non-tree models. When there are negative values, the feature range is normalized to [-1, 1] or [0, 1] using the MinMax scaling technique. When the data is not regularly distributed or the standard deviation is very small, the MinMax Scaler is helpful. An equation is as follows [19]:

$$S(x) = \frac{\chi^{(t)} - \chi_{min}}{\chi_{max} - \chi_{min}}. \quad (3)$$

C.4 FEATURE IMPORTANCE

It is essential to evaluate the significance of each feature in the dataset before selecting the optimal one because certain characteristics can contain extraneous data. The node impurity weight is decreased to indicate the significance of the feature based on the likelihood that the node will be reached. By dividing the total number of samples by the number of samples that reach the node, the node probability may be calculated. The highest value in the feature is thought to be more important. In this study, features that are significant and features that are not

required for the prediction of the target variable have been found using the additional tree model.

D. ENSEMBLE METHOD

A meta-classifier called the ensemble incorporates multiple different classifier algorithms [20]. When compared to individual classifiers, the performance of meta-classifiers is superior. Majority voting is a style of ensemble method that is highly common. Majority voting typically refers to binary classes, while pluralistic voting is the extension of the majority voting principle to numerous classes.

D.1 HYPERPARAMETER TUNING

In this study, the best parameters are found using a random search method, which has been validated using 10-fold correlated CVs. Random search works by performing a domain boundary search in the search space parameter using random sample points. This method has been chosen because it is quick and produces accurate results.

D.2 ENSEMBLE MAJORITY VOTING

Majority voting is one of the most widely used algorithms in the ensemble method. The majority voting principle is used to determine the prediction results, and it is based on the number of votes from each label that has the most votes [21]. The training data is used to train each classifier in Fig. 2 $(C_1, C_2, \dots, C_m)$. The ensemble technique can employ many classifiers, such as DT, LR, NB, etc., or it can employ the same classifier with various subsets, like the Random Forest algorithm, which gives each Decision Tree in it a majority vote. For either simple majority vote or plurality voting, the classifications can be predicted using the following equation:

$$\hat{y} = mode\{C_1(x), C_2(x), \dots, C_m(x)\}. \quad (4)$$

It is assumed that the ensemble has three classifiers $C_i (i \in \{0,1\})$ with class labels 0 and 1 based on sample x.

$$C_1(x) = 0, C_2(x) = 0, C_3(x) = 1. \quad (5)$$

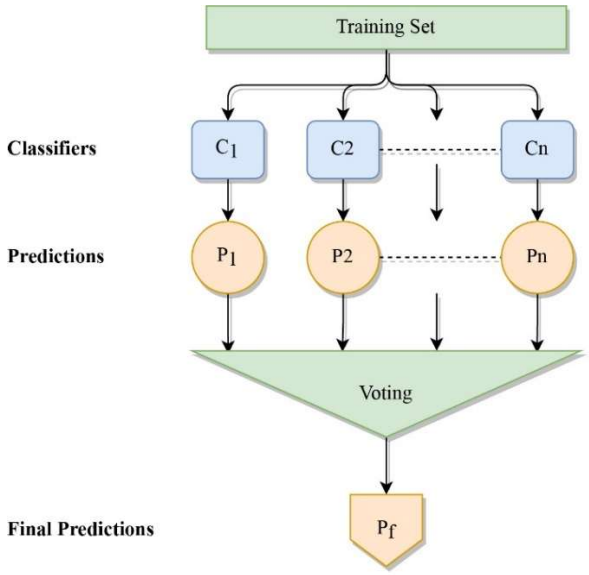

Figure 3. Majority voting architecture

For instance, the predicted results indicate that class 0 has 2x votes and the remaining 1x. When all classifier weights are equal, the outcome of a majority vote is anticipated to be 0.

$$\hat{y} = mode\{0,0,1\} = 0. \quad (6)$$

When using ensembles, the capacity to categorize fundamental people typically varies, and some may not perform better than others, leading to aggregation that may be subpar. The correct weight value for classifiers based on their individual performances is required to solve this issue. The selection of weight is crucial since it has the potential to change how the ensemble performs.

D.3 ENSEMBLE WEIGHTED VOTING

Not all classifiers in an ensemble-based classification system are equally accurate or powerful. It is feasible to give each classifier authority in choosing by assigning each classifier a certain amount of weight [22]. In Fig. 4, $p_{i,j}$ is denoted as the predicted member class of the probability of iteration i = th on the classifier for class label j. The w_i value is the optional parameter weight, by default $w_i = 1/n, \forall w_i \in \{w_1, w_2, \dots, w_n\}$, written in the following equation:

$$\hat{y} = \arg\max_j \sum_{i=1}^{n} w_i p_{i,j}. \quad (7)$$

By giving each classifier C_ta weight w_i proportionally based on performance, it is important to estimate the strength of the classifier performance in the future.

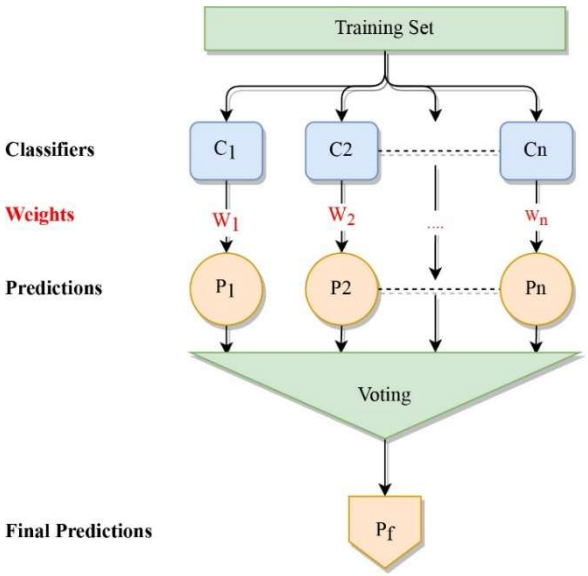

Figure 4. Weight voting architecture

E. OPTIMIZATION PSO

Kennedy and Eberhart introduced the PSO algorithm, which was first influenced by flocks of birds searching for food [23]. PSO is a useful optimization technique because of its quick time convergence, stability, few parameters, and simplicity of use. Potential solutions are represented in the PSO algorithm by particles, which come together to form a swarm. Position and velocity are the two qualities that each particle possesses. The search space vector dimension is denoted by the letter D.

The positions and velocities of the ith particle on the twelfth sequence are given as [24] :

$$x_i^t = (x_{i1}, x_{i2}, \dots, x_{iD}), \quad (8)$$
$$v_i^t = (v_{i1}, v_{i2}, \dots, v_{iD}). \quad (9)$$

The particle's velocity and position are repeatedly updated in the $(t+1)$th iteration using the following equation to calculate the speed and separation between the new and old particles:

$$v_{ij}^t = wv_{ij}^{t-1} + c1r1\left(P_{best}^t - x_{ij}^{t-1}\right) + c2r2\left(G_{best}^t - x_{ij}^{t-1}\right). \quad (10)$$

New particle position update:

$$x_{ij}^t = x_{ij}^{t-1} + v_{ij}^t, \quad (11)$$

where w is the inertial weight, $c1$ and $c2$ are the learning factors that determine positive constants. $r1$ and $r2$ are two random numbers from 0 to 1. P_{best}^t is the best position of the personal particle and G_{best}^t is the best position of the swarm.

F. PEFORMANCE EVALUATION

The PSO algorithm is applied to ensemble majority voting in this study by evaluating performance with the confusion matrix measurement in Table 2 and then measuring performance with several evaluation metrics that are commonly used to validate classifier models.

Table 1. Confusion Matrix

	Predicted class P.	Predicted class N.
Actual Class P.	True Positives (TP)	False Negatives (FN)
Actual Class N.	False Positives (FP)	True Negatives (TN)

It is crucial in assessing the confusion matrix classifier model. Positive data is anticipated to be true for the four matrix measurement reports that reflect true positives (TP). Data that is projected to be negative and so true (TN). Negative data that is expected to be positive is known as a false positive (FP). False negatives (FN) are positive results that are actually projected to be negative. It is therefore possible to utilize an assessment metric to assess the performance of the model based on the potential value revealed by the confusion matrix results. The performance measurement for this study, namely

$$AC = \frac{TP+TN}{TP+TN+FP+FN}. \quad (12)$$

The model accuracy in correctly identifying data is measured by accuracy (AC). The equation for the outcome of the accurate prediction ratio for all data is provided in Eq. (12).

$$PR = \frac{TP}{TP+FP}. \quad (13)$$

Precision (PR) measures how accurately the requested data compares to the model forecast outcomes. The ratio of correct positive predictions to all positive prediction outcomes is displayed in Eq. (13).

$$RE = \frac{TP}{TP+FN}. \quad (14)$$

Recall or sensitivity (RE) measures how well a model can retrieve data. To compare all of the correctly positive data, the ratio of correctly positive predictions is used; the equation is provided in Eq. (14).

$$SP = \frac{TN}{TN+FP}. \quad (15)$$

The model specificity (SP), which measures its accuracy in identifying data with negative labels, is how reliable it is. The outcome of the ratio of the accuracy of the negative predictions to the total negative data is displayed in Eq. (15).

$$F1 = 2 * (RE * PR) / (RE + PR). \quad (16)$$

By averaging precision and sensitivity harmonically, the F1 score is determined. The outcome of the weighted average precision and recall comparison ratio is displayed in the Eq. (16).

III. RESULTS AND DISSCUSSION

The experimental findings of the measurement reports from the preprocessing subsection are presented in this section, together with baseline classifiers, model parameters, a comparison of ensemble approaches, and an assessment of model performance.

A. RESULT OF PREPROCESSING

An empty value in the data is identified using the imputation approach. The results of the analysis indicate that several characteristics, including blood pressure, glucose, insulin, skin thickness, and BMI, have a value of 0 [25]. As a result, in this experiment, the value 0 becomes NaN. Table 3 displays how many values are empty. The MinMax scaling approach is used to normalize the data throughout the data transformation process. Although the scaling strategy is useful for some models, it does not influence the accuracy level of some classifiers, such as the model tree. A feature dataset with outliers was subjected to the IQR approach, which led to the removal of 186 rows of data. As can be seen in Fig. 5, some of the prior features before the implementation of IQR still have some outliers. When the data are considered free of outliers, it is then shown once more in Fig. 6. In Fig. 7, the outcomes of applying feature importance are displayed, with the features that are thought to be the most crucial in determining predictions on the target arranged according to their level of significance. The four most crucial factors insulin, glucose, skin thickness, and age are used in this experiment. Features that are considered unnecessary will be removed since they have no discernible impact on the outcomes of the predictions.

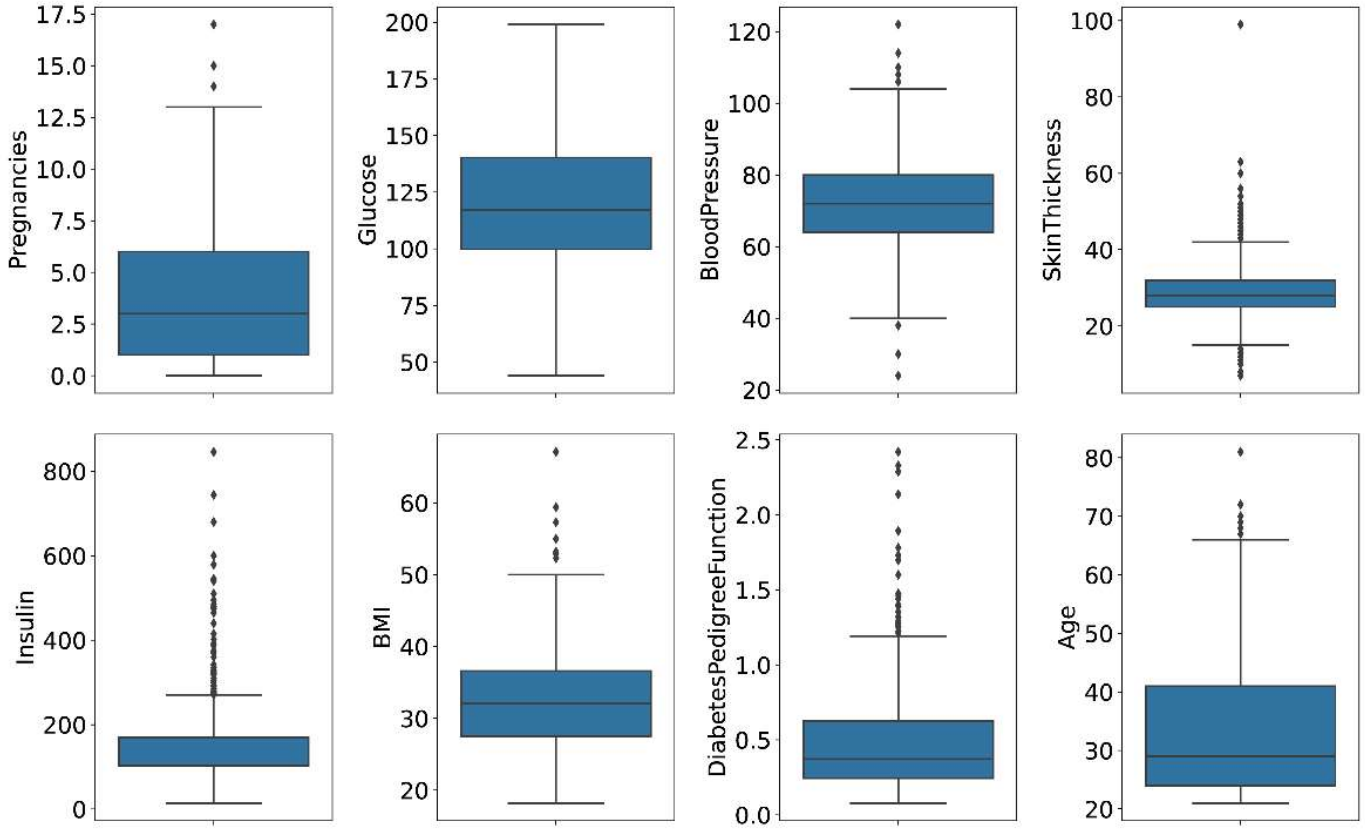

Figure 5. Dataset Outlier

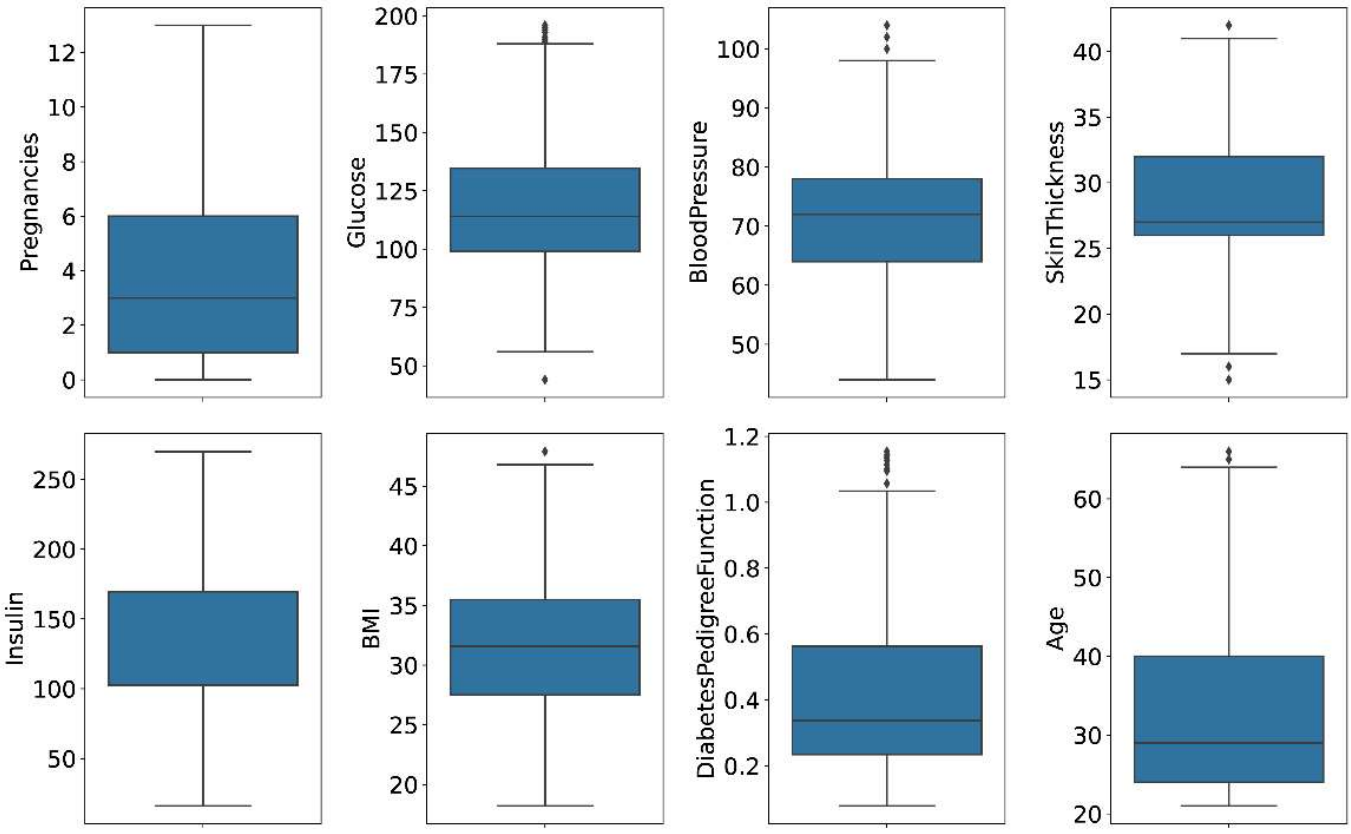

Figure 6. After Removing Outlier

Table 2. Comparism of preprocessing score

Preprocessing	XGB	LGBM	GBM	LR
[N]	0.7570	0.7222	0.7570	0.7552
[I]	0.8802	0.8854	0.8890	0.7900
[I, S]	0.8802	0.8820	0.8890	0.7813
[I, S, O]	0.8898	0.8921	0.9013	0.8621
[I, O, S, P]	0.9006	0.9121	0.9121	0.8528

Measurements were taken for each technique used to test the preprocessing stage to obtain the optimum tuning outcomes. XGB, LGBM, GBM, and LR are the four models with standard parameters (N) that were used to measure the prediction results using the PIDD dataset. The reported score does not determine the ultimate accuracy level because this step is only intended to evaluate the tuning outcomes. Rather, it serves as a measurement limit to identify the accuracy score level of the selection of preprocessing methods and is validated solely using the training data set. Using (I) as the median value to determine the blank value, the imputation stages are shown. Additionally, each feature's data is cleaned using the IQR approach, which is represented as (O). To determine which features contribute most to the selection of prediction targets, feature scaling (S) is performed by transforming feature values and applying feature importance (P). Table 4 shows a comparison of accuracy score measurements.

Table 3. Missing Values Data

No	Attributes	Missing Values
1	Pregnancies	0
2	Glucose	5
3	BloodPressure	35
4	SkinThickness	227
5	Insulin	374
6	BMI	11
7	DiabetesPedigreeFunction	0
8	Age	0
9	Outcome	0

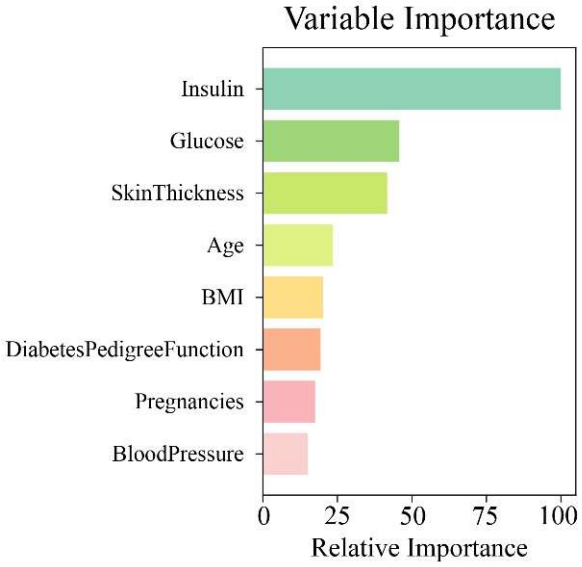

Figure 7. Feature Importance

B. RESULTS OF HYPERPARAMETER TUNING

The use of random search to find the best parameters yielded the best results, as shown in Table 5. The best score results and accuracy in the test data set. LR has a value of 0.8661 and a test value of 0.8644. GBM receives 0.9006 and 0.9152, XGB receives 0.9083 and 0.8983, and LGBM receives 0.9062 and 0.8814.

Table 5. The suitable model parameter

Model	Parameter	Best Parameter
LR	solver	liblinear
	penalty	l1
	C	100
GBM	n_estimators	250
	max_depth	3
	learning_rate	1
XGB	max_depth	4
	learning_rate	0.0342
	n_estimators	200
	subsample	0.5310
	gamma	5
	colsample_bytree	0.6357
	reg_alpha	0.7250
	reg_lambda	0.0013
LGBM	learning_rate	1.0
	num_leaves	24
	feature_fraction	0.1
	bagging_fraction	0.8
	max_depth	5
	max_bin	20
	min_data_in_leaf	80
	min_sum_hessian_in_leaf	0
	subsample	1.0

C. EMV RESULTS

A mixture of the four classifier models, LR, XGB, GBM, and LGBM, was used for voting in deciding the outcome prediction choice based on the findings of the tuning process in Table 5. The model parameters and weights are given below:

Table 6. Majority voting parameter

Model	Weight	Voting
XGB	default	soft
LGBM	default	soft
LR	default	soft
GB	default	soft

Table 6 shows the accuracy of the prediction using ensemble majority voting with default weights. The accuracy rate was 0.8983, the precision was 0.8824, the recall was 0.789, and the F1 was 0.8333.

D. PSO RESULT

The following PSO parameters are used during the optimization of the PSO on the ensemble majority voting weight in this study:

Table 7. PSO parameter

Parameter	Values
Learning Factor 1	0.5
Learning Factor 2	0.3
Inertia weight	0.9
Lower and upper bound	[0,1]
Dimension space	4
Number of particle	45
Number of iteration	100

The best cost measurement findings were produced based on Table 7, and they were 1.07272 with the best particle position being [0.2951 0.9060 0.0279 0.3804]. The ensemble majority voting model employed uses the particle location values as weights. The following is a diagram of the progression of particle measurement findings:

In terms of PSO's cost history Fig. 8 depicts a particle graph based on the fitness function in the search for space dimensions. The particles are known to converge at the 30th iteration of the 100 used. The best weight for each particle is determined and applied to the ensemble prediction model.

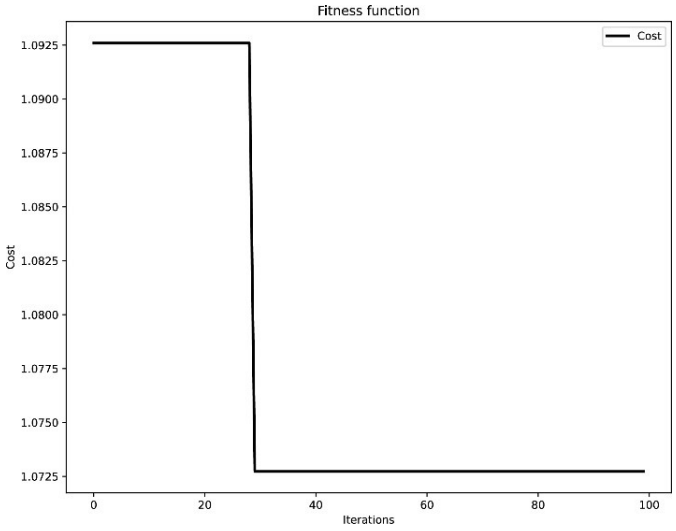

Figure 8. PSO cost history

E. EMVPSO RESULTS

The weights obtained through PSO optimization for each classifier in the optimized ensemble prediction model are shown in the EMVPSO parameters as follows:

Table 8. Weighted voting is a voting parameter

Model	Weight	Voting
XGB	0.2951	weight
LGBM	0.9060	weight
LR	0.0279	weight
GB	0.3804	weight

According to Fig. 9, the ensemble majority voting model with PSO-optimized weights produces the best prediction accuracy; the accuracy level is 0.9322, the precision is 0.9412, the recall is 0.8421, and the F1 is 0.8889.

F. EVALUATION PEFORMANCE

Examining the outcomes of the prediction model comes next after the training and testing process has been completed on the ensemble majority vote that has been optimized using PSO. The following curve illustrates the evaluation of model performance based on the false positive rate and false negative rate using the Receiver Operator Characteristic (ROC) and Area Under the Curve (AUC) score:

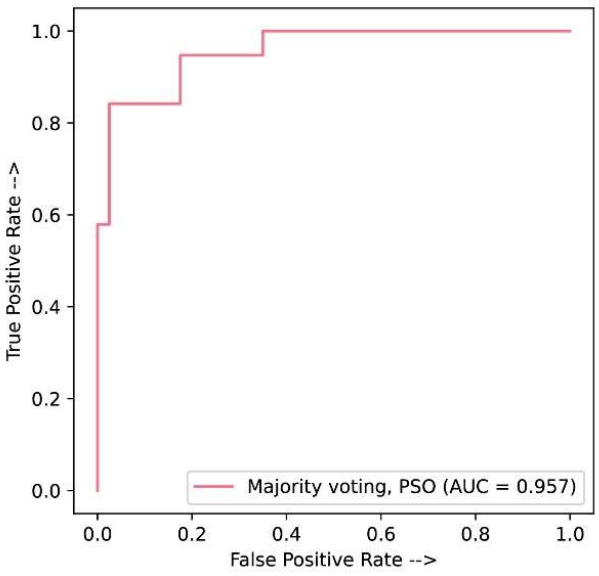

Figure 9. ROC AUC curve

According to Fig. 9, ensemble majority voting with PSO optimization results in an AUC of 0.957. Fig. 8 shows the results of comparing the accuracy level of the EMVPSO model with individual classifier models that have previously been subjected to hyperparameter tuning.

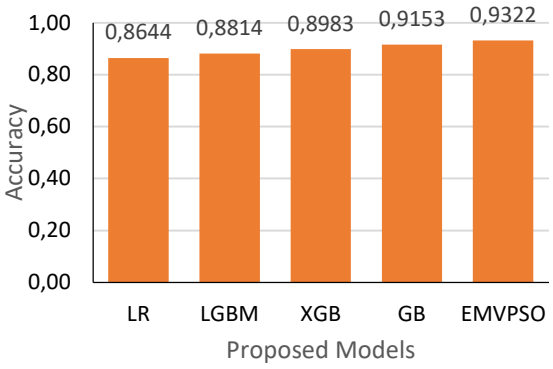

Figure 10. Accuracy comparison with individual models

According to the comparison results, the EMVPSO model has the highest accuracy value of 0.9322, followed by the GB model of 0.9153, XGB 0.8983, LGBM 0.8814, and LR 0.8644. The proposed model is also compared to standard ensemble voting (EMV) after weight optimization with PSO (EMVPSO).

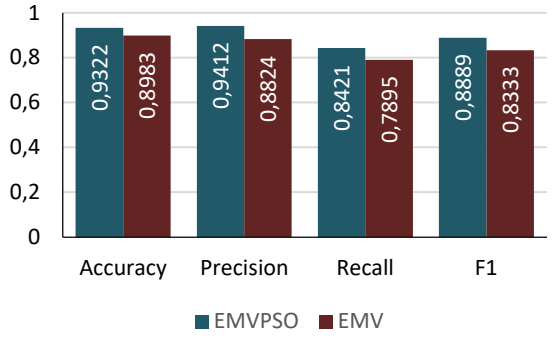

Figure 11. Accuracy of standard and optimization

According to Fig. 11, the EMVPSO model has the highest accuracy value of 0.9322 when compared to the standard voting model, which has an accuracy rate of 0.8983. By using "soft voting," the weights in the EMV model are equalized or averaged so that each classifier in the ensemble has the same level of power. In the EMVPSO model with weighted voting, where the default weight is optionally determined by the user, the weight is determined based on the results of the optimal solution to produce the best accuracy by applying the weighted results of the PSO optimization. The results of this comparison show that applying PSO to ensemble voting can increase accuracy; in this study, the percentage increase in accuracy was 4%, followed by 7% increases in precision, recall, and F1.

Table 9. Performance comparison with related research

Model	Classifier	Meta Learning	Evaluation	
Gopi et al [9]	LR	-	Accuracy	77.00
			Recall	77.00
			Precision	76.00
			F1	76.00
Satish et al [14]	ADB, SVM, MLP	Stacking	Accuracy	78.20
			Recall	51.00
			Precision	72.20
			F1	59.40
Saloni et al [15]	RF, LR, NB	Soft Voting	Accucary	79.04
			Recall	71.45
			Precision	73.48
			F1	80.60
Proposed Method	**XGB, GB, LGBM, LR**	**Weighted Voting + PSO**	Accuracy	**93.22**
			Recall	**84.21**
			Precision	**94.12**
			F1	**88.89**

In this case, the activities completed as a whole will be displayed in tabular form. Experiments using the Pima Indian Diabetes Dataset (PIDD) dataset have been carried out in related studies. Each of the three related research models, as shown in Table 9, has a preprocessor tuned to improve accuracy. Two of the prediction models used the ensemble method, namely stacking and majority voting with various classifier combinations, and validation was performed during training. According to research [15] and [14], the ensemble method model outperforms the individual model [9] in terms of accuracy. This study's model employs several preprocessors and employs the ensemble majority voting method, which has been optimized using PSO and validated using a 10-fold CV. With an accuracy of 93.22, a precision of 94.12, a recall of 84.21, and an F1 of 88.89, the proposed method outperforms other models.

III. CONCLUSION AND FUTURE WORK

The proposed prediction model is capable and accurate enough to be used to predict type 2 diabetes mellitus outcomes from the Pima Indian Diabetes Dataset (PIDD) data. This is demonstrated through testing with public datasets, where the data is pre-processed in ways such as imputation to determine empty values in the data, removal of outlier values in the data distribution, feature scaling to normalize data features to have the same range, and feature importance to determine features that affect the target data prediction results. It is concluded by applying to pre-process that clean data can affect the accuracy of the prediction model used. The accuracy of all models increased following the pre-processing stage, according to testing of the 12 base models. The ensemble-based prediction model uses the XGB, GB, LGBM, and LR classifier combinations during the training phase. The EMV model produced a performance of 0.8983 with 0.8824 precision, 0.7895 recall, and an F1 of 0.8333. The test was repeated while utilizing PSO optimization to determine the weights for the EMVPSO model. Accuracy increased to 0.9322, precision to 0.9412, recall to 0.8421, and F1 to 0.8889 as a result of the optimized weights. Performance, as measured by the ROC-AUC, generated a result of 0.9566. According to the experimental findings, PSO can create weights for ensemble majority voting that are more accurate than those produced by traditional ensemble majority voting. PSO results show a 4% improvement in accuracy performance, and a 7% improvement in precision, recall, and F1 score.

Future research ideas can be generated by adjusting the prediction model more thoroughly. Then, to achieve higher performance, it is possible to run tests using a wider variety of classifier combinations in the ensemble.

References

[1] S. Li, J. Wang, B. Zhang, X. Li, and Y. Liu, "Diabetes mellitus and cause-specific mortality: A population-based study," *Diabetes Metab. J.*, vol. 43, no. 3, p. 319, 2019. https://doi.org/10.4093/dmj.2018.0060.

[2] M. S. Rahman et al., "Role of insulin in health and disease: An update," *Int. J. Mol. Sci.*, vol. 22, no. 12, p. 6403, 2021. https://doi.org/10.3390/ijms22126403.

[3] A. Z. Woldaregay et al., "Data-driven modeling and prediction of blood glucose dynamics: Machine learning applications in type 1 diabetes," *Artif. Intell. Med.*, vol. 98, no. April 2018, pp. 109–134, 2019. https://doi.org/10.1016/j.artmed.2019.07.007.

[4] H. Qteat and M. Awad, "Using hybrid model of particle swarm optimization and multi-layer perceptron neural networks for classification of diabetes," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 3, pp. 11–22, 2021. https://doi.org/10.22266/ijies2021.0630.02.

[5] N. Esser, K. M. Utzschneider, and S. E. Kahn, "On the causal relationships between hyperinsulinaemia, insulin resistance, obesity and dysglycaemia in type 2 diabetes: Reply to Johnson JD [letter]," *Diabetologia*, vol. 64, no. 10, pp. 2345–2347, 2021. https://doi.org/10.1007/s00125-021-05511-6.

[6] P. Kostagiolas, P. Tsiligros, P. Theodorou, N. Tentolouris, and D. Niakas, "A cross-sectional survey interconnecting health information seeking behavior with clinical data of type 2 diabetes mellitus patients," *Libr. Hi Tech*, vol. 39, no. 2, pp. 448–461, 2020. https://doi.org/10.1108/LHT-02-2020-0030.

[7] International Diabetes Federation, *IDF Diabetes Atlas*, 9th edition, 2019.

[8] A. U. Haq et al., "Intelligent machine learning approach for effective recognition of diabetes in e-healthcare using clinical data," *Sensors (Switzerland)*, vol. 20, no. 9, article no. 2649, 2020. https://doi.org/10.3390/s20092649.

[9] G. Battineni, G. G. Sagaro, C. Nalini, F. Amenta, and S. K. Tayebati, "Comparative machine-learning approach: A follow-up study on type 2 diabetes predictions by cross-validation methods," *Machines*, vol. 7, no. 4, pp. 1–11, 2019. https://doi.org/10.3390/machines7040074.

[10] Y. Jian, M. Pasquier, A. Sagahyroon, and F. Aloul, "A machine learning approach to predicting diabetes complications," *Healthcare*, vol. 9, no. 12, p. 1712, 2021. https://doi.org/10.3390/healthcare9121712.

[11] 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, pp. 1–14, 2021. https://doi.org/10.3390/diagnostics11091714.

[12] H. Jafarzadeh, M. Mahdianpari, E. Gill, F. Mohammadimanesh, and S. Homayouni, "Bagging and boosting ensemble classifiers for classification of multispectral, hyperspectral and PolSAR data: A comparative evaluation," *Remote Sens.*, vol. 13, no. 21, p. 4405, 2021. https://doi.org/10.3390/rs13214405,

[13] L. J. Muhammad, E. A. Algehyne, and S. S. Usman, "Predictive supervised machine learning models for diabetes mellitus," *SN Comput. Sci.*, vol. 1, no. 5, pp. 1–10, 2020. https://doi.org/10.1007/s42979-020-00250-8,

[14] S. K. Kalagotla, S. V. Gangashetty, and K. Giridhar, "A novel stacking technique for prediction of diabetes," *Comput. Biol. Med.*, vol. 135, no. June, p. 104554, 2021. https://doi.org/10.1016/j.compbiomed.2021.104554,

[15] S. Kumari, D. Kumar, and M. Mittal, "An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier," *Int. J. Cogn. Comput. Eng.*, vol. 2, no. November 2020, pp. 40–46, 2021. https://doi.org/10.1016/j.ijcce.2021.01.001.

[16] M. Shahhosseini, G. Hu, and H. Pham, "Optimizing ensemble weights and hyperparameters of machine learning models for regression problems," *Mach. Learn. with Appl.*, vol. 7, p. 100251, 2022. https://doi.org/10.1016/j.mlwa.2022.100251,

[17] Y. Xu, C. Wu, K. Zheng, X. Wang, X. Niu, and T. Lu, "Computing adaptive feature weights with PSO to improve android malware detection," *Secur. Commun. Networks*, vol. 2017, Article ID 3284080, 2017. https://doi.org/10.1155/2017/3284080.

[18] M. Maniruzzaman et al., "Accurate diabetes risk stratification using machine learning: Role of missing value and outliers," *J. Med. Syst.*, vol. 42, no. 5, p. 92, 2018. https://doi.org/10.1007/s10916-018-0940-7.

[19] G. Aksu, C. O. Güzeller, and M. T. Eser, "The effect of the normalization method used in different sample sizes on the success of artificial neural network model," *Int. J. Assess. Tools Educ.*, vol. 6, no. 2, pp. 170–192, 2019. https://doi.org/10.21449/ijate.479404.

[20] M. S. Sainin, R. Alfred, and F. Ahmad, "Ensemble Meta Classifier with Sampling and Feature Selection for Data with Multiclass Imbalance Problem," *J. Inf. Commun. Technol.*, vol. 20, no. 2, pp. 103–133, 2021. https://doi.org/10.32890/jict2021.20.2.1.

[21] S. Y. Kim and A. Upneja, "Majority voting ensemble with a decision trees for business failure prediction during economic downturns," *J. Innov. Knowl.*, vol. 6, no. 2, pp. 112–123, 2021. https://doi.org/10.1016/j.jik.2021.01.001.

[22] H. Bouziane, B. Messabih, and A. Chouarfia, "Profiles and majority voting-based ensemble method for protein secondary structure prediction," *Evol. Bioinforma.*, vol. 7, no. 7, p. EBO.S7931, 2011. https://doi.org/10.4137/EBO.S7931.

[23] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceedings of the International Conference on Neural Networks ICNN'95*, 1995, vol. 4, pp. 1942–1948.

[24] D. Ratnawati, M. Marjono, W. Widodo, and S. Anam, "PSO-ELM with time-varying inertia weight for classification of SMILES codes," *Int. J. Intell. Eng. Syst.*, vol. 13, no. 6, pp. 522–532, 2020. https://doi.org/10.22266/ijies2020.1231.46.

[25] Y. Hayashi and S. Yukita, "Rule extraction using recursive-rule extraction algorithm with J48graft combined with sampling selection techniques for the diagnosis of type 2 diabetes mellitus in the Pima Indian dataset," *Informatics Med. Unlocked*, vol. 2, pp. 92–104, 2016. https://doi.org/10.1016/j.imu.2016.02.001.

***MULJONO** is with Informatics Engineering Department, Dian Nuswantoro University, Semarang, Jawa Tengah, Indonesia. He received his Doctorate Degree in Electrical Engineering Department, Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia in 2016. He joined an Internship Program at School of Media Science, Tokyo University of Technology Japan in 2014. His research interests are speech processing, artificial intelligence, machine learning and natural language processing.*

***NOVIAN ADHIPURNA,** currently works in Department of Pharmacy, Universitas Borneo Lestari, Kalimantan, Indonesia. He received his bachelor degree of Informatics Engineering from STMIK Banjarbaru in 2016. He also earned his master's degree of Informatics Engineering from Universitas Dian Nuswantoro in 2022.*