

Received 16 January 2024, accepted 25 January 2024, date of publication 30 January 2024, date of current version 5 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3359760



## RESEARCH ARTICLE

# Improving Healthcare Prediction of Diabetic Patients Using KNN Imputed Features and Tri-Ensemble Model

KHALED ALNOWAISER<sup>ID</sup>

Department of Computer Engineering, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

e-mail: k.alnowaiser@psau.edu.sa

This work was supported by Prince Sattam bin Abdulaziz University under Project PSAU/2024/R/1445.

**ABSTRACT** **Objective:** Diabetes ranks as the most prevalent ailment in developing nations. Vital steps to mitigate the consequences of diabetes include early detection and expert medical intervention. A highly effective approach for identifying diabetes involves assessing the specific indicators associated with this condition. When it comes to automated diabetes detection, frequently encountered datasets frequently exhibit gaps in data, which can markedly impact the effectiveness of machine learning models. **Methods:** The aim of this study is to propose an automated method for predicting diabetes, with a focus on appropriately dealing with missing data and improving accuracy. The proposed framework makes use of K-Nearest Neighbour (KNN) imputed features along with a Tri-ensemble voting classifier model. **Results:** By incorporating the KNN imputer, the presented model demonstrates impressive performance metrics, including an accuracy of 97.49%, precision of 98.16%, recall of 99.35%, and an F1 score of 98.84%. The study conducted a thorough comparison of this proposed model against seven alternative machine learning algorithms, assessing them under two conditions: one with omitted missing values and another with the KNN imputer applied. These findings support the proposed model's efficacy, highlighting its superiority over currently established state-of-the-art techniques. **Conclusion:** This research explores the problem of missing data in diabetes diagnosis and highlights the efficacy of the KNN-imputed technique. The results are promising for healthcare practitioners as they could facilitate early detection and improve the quality of diabetic patient care.

**INDEX TERMS** Diabetes detection, ensemble learning, missing values, KNN Imputer, healthcare.

## I. INTRODUCTION

Healthcare professionals play a pivotal role in diagnosing and treating various medical conditions, whether they be diseases, injuries, or mental and physical impairments in individuals. This group encompasses dentists, physicians, surgeons, and their associated colleagues. Additionally, healthcare extends to fields such as nursing, optometry, physical therapy, pharmacy, athletic training, and more. The healthcare system is regarded as a fundamental element in preserving both the physical and mental well-being of

The associate editor coordinating the review of this manuscript and approving it for publication was Yizhang Jiang<sup>ID</sup>.

individuals. Diagnosing a disease is an essential first step in its treatment; it is especially important to detect conditions like Diabetes Mellitus (DM), also known as diabetes, as soon as possible. It is a condition where either insufficient insulin is produced or ineffectively managed by the body. The blood sugar level is regulated by a hormone called Insulin. Uncontrolled diabetes results in hyperglycemia (a condition of high blood sugar). Hyperglycemia causes serious damage to essential organs and their functioning, mainly impacting nervous and blood vessel systems [1]. Diabetes affected 8.5% of the population over the age of 18 in 2014. Diabetes was responsible for 2.2 million fatalities globally in 2012, while roughly 1.6 million individuals died as a result of diabetes

in 2016 [1], [2], [3], [4]. The year 2019 saw the death toll exceed 1.5 million due to diabetes [5]. With the increasing number of diabetes patients, it poses a significant financial burden on global healthcare services, becoming a widespread economic concern. The global estimate has reached almost 825 billion dollars annually to care for diabetes patients [6]. Diabetes sufferers would number 629 million by the end of 2045, according to statistics [7].

There are four types of Diabetes Mellitus (DM) [8]: Type 1 or juvenile diabetes, which is also called Insulin Dependent Diabetes Mellitus (IDDM); Type 2 or Non Insulin Dependent Diabetes Mellitus (NIDDM); Type 3 or Gestational Diabetes (GD); and Type 4 or impaired Glucose Regulation (GR), which is also known as pre-diabetes. The cause of Type 1 diabetes is the body's failure to produce enough insulin, which requires external insulin injections. Type-2 DM, on the other hand, is non-insulin dependent, meaning insulin supplementation from an external source is not required. This disorder is distinguished by the body's inability to appropriately use insulin. The blood sugar level increases in pregnant women who had no diabetes symptoms before due to gestational diabetes, which is the third type of diabetes. Type-4 is pre-diabetes or impaired glucose, where the level of blood sugar is lower than Type-2 but higher than the normal range. Normal glucose level ranges between 100 and 125mg/dL. The following are the few problems that may lead to DM according to WHO

- An increase in blood glucose levels above the usual range,
- Constantly higher fasting blood sugar levels above the normal range,
- High levels of triglycerides in the blood,
- People rarely work out and are age 45 and above.,
- High-pressure level of blood,
- Pregnant women age 30 and above,
- More than 24 kg/m<sup>2</sup> body mass index.,
- Having a family history of diabetes.

Statistics by the WHO show that the number of patients with DM is increasing day by day. A large number of those living in developing and developed countries are increasingly becoming inactive. Furthermore, eating habits like obesity, fast and junk food, irregular eating, etc. may also lead to DM. Depression, work anxiety, and job-related pressure disturb the stomach and may cause DM. Lack of skills and knowledge to use existing technologies to live a healthier life is hampering the management and control of diabetic conditions. Diabetic complications may be avoided by adopting regular exercise and eating habits. Thoughtful exertions are needed to change the current lifestyle. Meal suggestion systems, tracking, and monitoring of physical activity, drug warning systems, and interactive chatbots are the examples of latest technologies being deployed for the betterment of patients.

Data mining is important in health analytics and medical-related databases because it improves the specificity and sensitivity of disease diagnosis. Diagnosis using better

resources and technologies may prove to be more accurate and helpful [9]. Using machine learning (ML) and data mining approaches for diabetes detection is extremely beneficial for quick and accurate diagnosis. ML models have been employed to achieve automated and better results [10]. Diverse methodologies and models have been used to attain the highest classification accuracy. The data used for such models have been collected from heterogeneous sources and contain incomplete and null records. Consequently, the performance of the models is affected. The goal of this study is to create a predictive model for diabetes prediction using machine learning techniques. The study looks into the impact of missing data on diabetes diagnosis and puts forth the following contributions:

- A Novel machine learning-based ensemble approach is proposed for diabetes prediction. Proposed ensemble model utilizes the strength of the three different learning models (Extra Tree Classifier (ETC), Extreme Gradient Boosting (XGB), and Random Forest (RF)) for final prediction utilizing a voting mechanism. The proposed model is named as Tri-Ensemble Model.
- For missing values, experiments are performed with the original data, as well as, using the K-Nearest Neighbour (KNN) imputer to complement the missing data. The performance of models with and without KNN is performed.
- The assessment of the proposed system involves the utilization of various machine learning models including Decision Tree (DT), Stochastic Gradient Descent (SGD), Gaussian Naive Bayes (GNB), Random Forest (RF), Logistic Regression (LR), Gradient Boosting Machine (GBM), Extra Tree Classifier (ETC), and Support Vector Classifier (SVC). Ultimately, the performance of the proposed approach is compared with other state-of-the-art techniques.

The paper's structure is delineated as follows: Section II provides a summary of previous literature concerning Diabetes Prediction. Section III encompasses details about the dataset, experimental methodologies, and the proposed approach. Section IV delivers the achieved results, and Section V encapsulates the conclusion while outlining potential avenues for future research.

## II. RELATED WORK

The combination of data mining [11], Internet of Things (IoT) [12], and Machine Learning (ML) [13] represents a potential solution for addressing various problems. Medical data is difficult to analyze due to its sheer size and enormous feature set [14]. ML has demonstrated its significance in numerous domains including healthcare. It has enabled the development of precise and reliable systems for medical applications, all while ensuring the protection of sensitive medical data. Likewise, ML models have been utilized to identify early-stage risks associated with DM. Risk factors such as body mass index (BMI), insulin levels,

glucose levels, and blood pressure are commonly taken into account.

The authors describe a machine learning-driven method in [15] that predicts the probability of Type-2 onset of diabetes in the following year ( $y + 1$ ) based on data from the current year ( $y$ ). The dataset is sourced from electronic health records (EHR) obtained from a private medical institute covering the years 2013 to 2018. They select features using two methods: analysis of variance (ANOVA) and chi-square. The method obtains an accuracy rate of 81% by utilizing an ensemble classifier that combines Confusion Matrix Based Integration (CIBM) and Soft Voting (SV). Rupapara et al. [16] presented an ensemble learning method for diabetes detection. In their study, they utilized a publicly available dataset (Pima India). To assess the effectiveness of the system, they also employed eight individual machine learning models. They conducted various experiments using Principal Component Analysis (PCA), Chi-2, and the original features. The results indicate that the Chi-2 features exhibit superior performance compared to other features, and the proposed Tri-ensemble model attained an accuracy value of 85% for diabetes prediction.

Deng et al. [17] applied transfer learning and data augmentation techniques to tackle challenges stemming from imbalanced datasets and limited training data. They explored three different neural network architectures, augmentation methods, transfer learning strategies, and various loss functions, including generative and mixed-up models. The OhioT1DM public dataset was utilized to develop a similar network architecture for detecting Type 1 diabetes, with experiments approved by the Beth Israel Deaconess Medical Center and the International Review Board. The authors reported an accuracy of 95%. Similarly, Butt et al. [18] introduced a machine-learning approach with the aim of detecting and categorizing early-stage diabetes. Furthermore, they proposed a hypothetical Internet of Things (IoT)-based system for monitoring blood glucose (BG) levels in both healthy individuals and those with diabetes. The diabetes classification involved the utilization of RF, LR, and Multilayer Perceptron (MLP). For predictive analysis, they employed linear regression Moving Average (MA), and Long Short-Term Memory (LSTM) techniques. The results showed an 86.08% classification accuracy achieved by the MLP model and an 87.26% prediction accuracy by the LSTM model.

Shamreen Ahamed and colleagues [8] proposed a machine learning approach incorporating feature augmentation and oversampling. They applied this technique to two benchmark datasets, namely Pima and DMS. The authors conducted experiments under two scenarios: one without preprocessing, and another with preprocessing and augmentation. A comparative analysis of the outcomes reveals that the highest accuracy of 92.5% was attained on the Pima Indian dataset, whereas for the DMS dataset, an accuracy of 98.99% was reported. Similarly, Pethunachiyar [19] introduced a machine learning-based system for classifying patients with diabetes

mellitus. The study demonstrated that linear function-enabled SVM outperformed decision trees (DT), NB, and neural network-type classifiers. However, it's worth noting that the paper lacks a state-of-the-art comparison, and detailed parametric information about the employed models is not provided.

In the domain of early-stage diabetes prediction, Laila et al. [20] proposed an ensemble learning system. The authors utilized the UCI diabetes dataset, which includes 17 attributes for each record. Additionally, they employed Chi-2 for feature selection. The experiments involved three predictive models: AdaBoost, bagging, and Random Forest (RF). The experimental results showcased an impressive accuracy of 97% for the RF model, surpassing the performance of the other models. For the detection of Type-2 diabetes mellitus, Madan et al. [21] introduced an ensemble system based on deep learning. The research included testing with both independent and ensemble deep learning models. The results showed that the ensemble model CNN-BiLSTM outperformed the other models, with an accuracy of 88.33%.

For the classification of type-2 diabetes, Kannadasan et al. [22] presented a specialized Deep Neural Network (DNN) model. For the feature engineering, they used utilized stacked encoders, implemented a backpropagation method for network fine-tuning, and incorporated a softmax function for classification. The model was trained on the PIDD dataset, which comprises 786 patient records. The reported classification accuracy was 86.26%, demonstrating the model's effectiveness. Dutta et al. [23] presented an automated pipeline for the early prediction of diabetes. The study involved the utilization of five distinct machine learning models, with their parameters fine-tuned to optimize performance. The pipeline also integrated feature selection, missing value imputation, and K-fold cross-validation techniques. Through statistical analysis of variance (ANOVA) testing, it was demonstrated that the projected weighted ensemble approach (which combines Random Forest, Decision Trees, LightGBM, and XGBoost models) along with the pre-processing techniques significantly enhanced the performance of diabetes prediction. The study reported an accuracy of 73.5% using this ensemble approach.

Despite the research works discussed above and their high performance, several limitations can be observed. For example, the challenge of missing values is not investigated although a few studies involve data augmentation to deal with class imbalance problems. It is important to note that some of the models discussed earlier may have been evaluated using limited data, which restricts the generalizability of the results. Additionally, the aspect of feature selection for DM prediction requires further study and investigation to improve its effectiveness and accuracy. Further research efforts are needed to improve the models' performance for DM detection. This may encompass the exploration of innovative architectures, fine-tuning of parameters, and the utilization of ensemble

**TABLE 1.** Overview of related work.

Reference	Models	Database	Reported result
[15]	SVM, RF, XGBoost, LR, CIM, SV, ST Ensemble (CIM, SV, ST)	Private institute 2013–2018	81% Ensemble
[16]	LR, ADA, DT, RF, ETC, GNB, SVC, LTC (ensemble), KNN	Pima India	85% LTC (ensemble)
[17]	RNN, CNN, SAN	BIDMC dataset	95% CNN
[18]	RF, MLP, LR, LSTM, MA, Linear regression	Pima Indian	87.26% LSTM
[8]	LGBM, RF, GBM	Pima, DMS	98.99% LGBM (DMS) 92.5% LGBM (Pima)
[19]	Linear SVM, Radial SVM, Polynomial SVM, NB, DT and NN	UCI diabetes dataset	100% Linear SVM
[20]	ADA, Bagging, RF	UCI diabetes dataset	97% RF
[21]	CNN, BiLSTM, CNN-LSTM, Dense-NN, Proposed (CNN-BiLSTM)	Pima	88.37% CNN-BiLSTM
[22]	DNN	Pima	86.26%
[23]	NB, RF, DT, XGB, LGB and proposed (DT+RF+XGB+LGB)	DDC dataset (Bangladesh)	73.5% proposed

**TABLE 2.** PIMA Indian diabetes dataset description.

Dataset attribute	Description	Type of attribute	Mean ± SD
Age	Age in years	Continuous value	33.24 ± 11.76
Class (Target)	Diabetic vs control	Categorical	
Glucose	Level of Plasma glucose (2-h)	Continuous	121.67 ± 30.46
Mass	Body mass index (weight kg/height m <sup>2</sup> )	Continuous	32.43 ± 6.88
Insulin	Two hours serum-insulin (μ U/ml)	Continuous	141.76 ± 89.10
Pedigree	Diabetes pedigree function	Continuous	0.47 ± 0.33
Pressure	Diastolic blood pressure (mm Hg)	Continuous	72.38 ± 12.10
Triceps	Triceps skin fold thickness (mm)	Continuous	29.08 ± 8.89
Pregnant	Number of times pregnant	Continuous	3.84 ± 3.36

models. A summary of pertinent studies can be found in Table 1.

### III. MATERIALS AND METHODS

#### A. DIABETES DATASET

The data employed in this research originates from the Kaggle data repository, specifically the “Pima Indians Diabetes Database.” This dataset has been utilized in numerous previous studies. It encompasses a collective of 768 instances from female patients with Pima Indian ancestry [24], all of whom are 21 years of age. 268 of these samples are from diabetic individuals, whereas the rest 500 are from non-diabetic individuals. The occurrence of several zero values for various attributes is one noticeable facet of the dataset. For instance, 27 patients exhibit a BMI value of zero, 35 patients have a diastolic blood pressure reading of zero, and 227 patients have a skinfold thickness of zero. In order to address this issue, KNN imputer is used to handle the presence of zero, which holds significant predictive power for determining the final class (diabetic or non-diabetic). The dataset description is given in Table 2.

#### B. DATA PREPROCESSING

For the significant improvement in the performance of machine learning models data preprocessing played a vital

role. This phase entails removing redundant or irrelevant data that does not provide meaningful information to the models [25]. Through proper preprocessing, the efficacy of learning models can be substantially improved. Furthermore, it aids in reducing computational time. In the context of this research, it was observed during the data preprocessing phase that the dataset included numerous zero values in various features. For instance, 27 patients had a BMI value of zero, 35 patients had a diastolic blood pressure reading of zero, 374 patients had a serum insulin level of zero, and 227 patients had a skinfold thickness of zero. Addressing these zero values is a crucial aspect of the preprocessing procedure. Attributes with zero or null values in the dataset can exert a noteworthy influence on model performance. To address this concern, we utilized the KNN imputer. The attributes containing zero (missing value) in the dataset are outlined in Table 3.

The table 3 clearly demonstrates that there is some presence of missing values in the dataset. Given that the dataset is categorical, there are two viable methods for addressing these missing values:

**KNN imputer:** One approach is to employ the KNN imputer. This method entails estimating the missing values by taking into account the values of its K nearest neighbors. The imputer leverages the existing data

**TABLE 3.** Missing values details attribute-wise.

Dataset attribute	Missing value
Age	0
Diabetes (Target)	0
Glucose	5
Insulin	374
Mass	11
Pressure	35
Pedigree	0
Pregnant	0
Triceps	227

points to make informed estimations and fill in the gaps.

**Removing missing values:** Alternatively, another approach is to straightforwardly eliminate the instances or rows containing missing values from the dataset. This involves excluding any samples with missing values from the analysis, leading to a smaller but complete dataset.

Both methods have their advantages and considerations. The choice between them depends on the specific characteristics of the dataset, the extent of missing values, and the goals of the analysis.

### C. KNN IMPUTER

Data is obtained from different sources in the modern world to enable analysis, draw insights, and confirm hypotheses. However, encountering missing information in collected data is not uncommon, often attributable to extraction issues or human errors during collection. As a result, in the data preprocessing step handling missing values is a very important step. The selection of an appropriate method for filling in these missing values is of great significance, as it can significantly impact model performance [26]. The KNN imputer, which is freely accessible in the sci-kit-learn package, is a widely used method for filling in missing data. This method is an alternative to standard methods. The KNN imputer uses the Euclidean distance matrix to calculate the nearest neighbors, allowing for the imputation of missing values in the observations. When computing the Euclidean distance, it ignores missing values and gives higher weight to non-missing coordinates. The Euclidean distance may be calculated using the following equation:

$$D_{xy} = \sqrt{\text{weight} * \text{squared distance from present coordinates}} \quad (1)$$

The weight is undoubtedly defined by the ratio of the total number of coordinates to the number of present coordinates. This ratio holds a pivotal role in computing the Euclidean distance, allowing for a meaningful estimation of missing values.

### D. DELETING MISSING VALUES

Another option for dealing with missing values in data is to completely eliminate them. Using this technique, all missing values are removed from the dataset and its size

is decreased. The decreased size dataset may result in over fitting or under-fitting due to a lesser/imbalance number of records.

### E. MODELS USED FOR DIABETES PREDICTION

Presently, a variety of ML algorithms is available for diabetes prediction. Several ML models have already been applied for the same task and different results are reported. Keeping in view the reported results of such models, the most accurate models are employed in this research. This research utilizes RF, XGB, GNB, ETC, DT, SGD, SVM, and LR. A concise explanation of these algorithms is provided in this section of the study.

#### 1) XGBOOST

XGBoost is a popular model utilized for supervised regression models, specifically designed to define the accuracy of the base learners and objective functions [27]. By employing the ensemble learning concept, it combines the outcomes of individual models through training and generates a unified prediction. XGBoost is classified as an ensemble learning technique.

#### 2) DECISION TREE

Decision Trees (DT) are predictive models that utilize a hierarchical tree structure, as cited in [27] and [28]. They organize features, depicted as branches, to make predictions about the target value, which is located in the tree leaves. When the target parameter has a finite set of values, classification trees are utilized. The branches in these tree models indicate the feature specifications that lead to these class labels, while the leaves represent class labels. Regression trees, on the other hand, are used when the target parameter may take on continuous values, often real numbers. The choice of the root node in a Decision Tree primarily depends on two criteria: information gain (IG) and the Gini index. IG is employed to determine the top node in a Decision Tree, playing a crucial role in its construction.

#### 3) GAUSSIAN NAIVE BAYES

The GNB model utilizes Bayes' theorem. The output of an event is predicted using unconditional probabilities in GNB [29]. If a sample is classified into  $k$  categories represented by  $k = \{c_1, c_2, \dots, c_k\}$ , the resulting output is denoted as  $c$ . The GNB function is expressed as under, where the class is represented by  $c$  and the sample by  $d$

$$P(c|d) = (P(c) \times P(d|c))/P(d) \quad (2)$$

In the provided equation, the probability of class  $c$  is denoted as  $P(c|d)$ , given the sample  $d$ . The prior probability of class  $c$  is represented as  $P(c)$ , while the likelihood of observing sample  $d$  for a given class  $c$  is denoted as  $P(d|c)$ . Additionally,  $P(d)$  signifies the probability of observing sample  $d$ .

#### 4) LOGISTIC REGRESSION

LR is a flexible regression technique used to construct predictors by combining binary covariates using boolean operations. LR derives its name from the logistic function, which forms the basis of this method [30]. A sigmoid function, also known as the logistic function, is distinguished by an S-shaped curve that transfers real-valued integers to values ranging from 0 to 1. LR is particularly well-suited for situations where the dependent variable is categorical, making it an optimal choice in such scenarios.

#### 5) STOCHASTIC GRADIENT CLASSIFIER

SGD is a powerful classifier for multi-class classification problems. Its foundation lies in the principles of SVM and LR which employ convex loss functions [31]. SGD leverages the one-vs-all (OvA) technique to combine multiple binary classifiers. One of the key advantages of SGD is its ability to handle large datasets efficiently. It achieves this by using an extreme approach, processing only a single example (with a batch size of 1) per iteration. The simplicity of SGD, rooted in its regression technique, makes it easily comprehensible and implementable. However, it is important to note that SGD can be quite noisy due to its random selection of examples from the batch. Additionally, achieving accurate results with SGD requires careful calibration of its hyperparameters, and it exhibits high sensitivity toward feature scaling.

#### 6) RANDOM FOREST

RF integrates the outputs of multiple decision trees to yield a single outcome [32]. This is accomplished by employing decision trees as a foundational technique for sampling rows and columns. The number of decision trees, known as base learners, is optimized based on the input, resulting in reduced variance and enhanced accuracy. RF is widely recognized as a significant approach within the realm of bagging methodologies.

#### 7) EXTRA TREE CLASSIFIER

ETC constructs an ensemble of unpruned decision trees using a top-down approach. During the node splitting process, ETC introduces strong randomization by randomizing both the selection of attributes and cut points [33], [34]. In extreme cases, ETC can generate fully randomized trees that are independent of the output values in the training sample. The primary distinction between ETC and the well-known machine learning model RF is as follows

- ETC employs the entire dataset for training the model, whereas Random Forest uses bootstrap replicas (random subsets) for training.
- Best features along with their corresponding values are randomly selected by ETC to split the nodes.

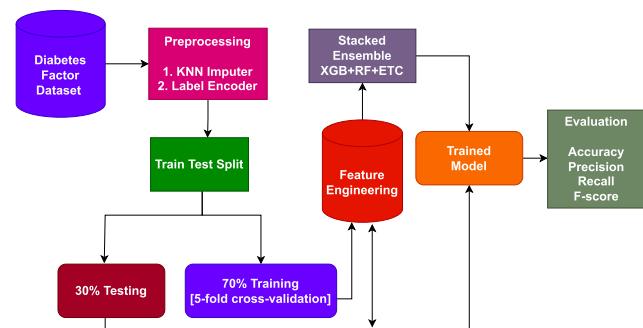
Due to these characteristics, the ETC is less prone to overfitting and often achieves better performance on datasets.

#### 8) SUPPORT VECTOR CLASSIFIER

Support Vector Classification (SVC) is a widely used machine learning technique designed to identify a hyperplane in an N-dimensional space for classifying data points [35]. The key objective of this algorithm is to locate the hyperplane that maximizes the margin between different classes. This emphasis on maximizing the margin helps enhance the robustness and generalization capabilities of the classifier. The dimensionality, represented by  $N$ , varies depending on the number of features. While comparing two features is relatively straightforward, dealing with multiple features for classification can be more complex. By maximizing the margin, the accuracy of prediction is enhanced by SVC.

#### F. PROPOSED TRI-ENSEMBLE MODEL FOR DIABETES PREDICTION

This research work makes use of a dataset collected from Kaggle, a well-known platform for publicly available datasets, to optimize the performance of learning models and manage missing values. The dataset is preprocessed, and missing values were addressed using the KNN imputer approach. The dataset is divided into a 70:30 ratio, with 70% used for model training and 30% utilized for testing. This study presents an automated diabetes prediction system called the Tri-Ensemble model that employs an ensemble technique to identify diabetes. Ensemble models are a strong tool for improving accuracy and resilience by combining predictions from many models. Three common algorithms are combined in the proposed model: XGBoost (XGB), Random Forest (RF), and Extra Trees Classifier (ETC). Figure 1 depicts the process of the suggested technique.



**FIGURE 1.** Proposed methodology for diabetes detection.

By combining predictions from three different machine learning algorithms, the ensemble model is built. The common way to build an ensemble model is to train multiple models on the same data and then combine their predictions. For the XGB+RF+ETC ensemble model, this means training the XGBoost (XGB), Random Forest (RF), and Extra Trees Classifier (ETC) models separately on the same data. Each model produces a probability prediction for each class of the outcome variable. The final prediction for each data

point is then obtained by adding up these probabilities. A weighted average of the expected probabilities is a typical method for aggregating predictions. Each model's weights are assigned depending on its performance on a validation set. This guarantees that models that outperform on the validation set are given greater weight in the ensemble fusion. By training many models on the same dataset and combining their predictions, we may increase the model's overall performance while minimising the risk of overfitting. The proposed ensemble model's operation is outlined in Algorithm 1, which may be summarised as follows:

$$\hat{p} = \operatorname{argmax}\{\sum_i^n XGB_i, \sum_i^n RF_i, \sum_i^n ETC_i\}. \quad (3)$$

In the case of  $\sum_i^n XGB_i$ ,  $\sum_i^n RF_i$ , and  $\sum_i^n ETC_i$  models, each of them generates prediction probabilities for each test sample. These probabilities are then run through the soft voting criterion, as seen in Figure 2. The soft voting criterion involves aggregating the probabilities from all three models and making a final prediction based on the combined probabilities. This approach allows the ensemble model to leverage the collective insights from XGB, RF, and ETC to arrive at a more robust and accurate prediction for each test case.

#### Algorithm 1 Ensembling of XGB, RF, and ETC

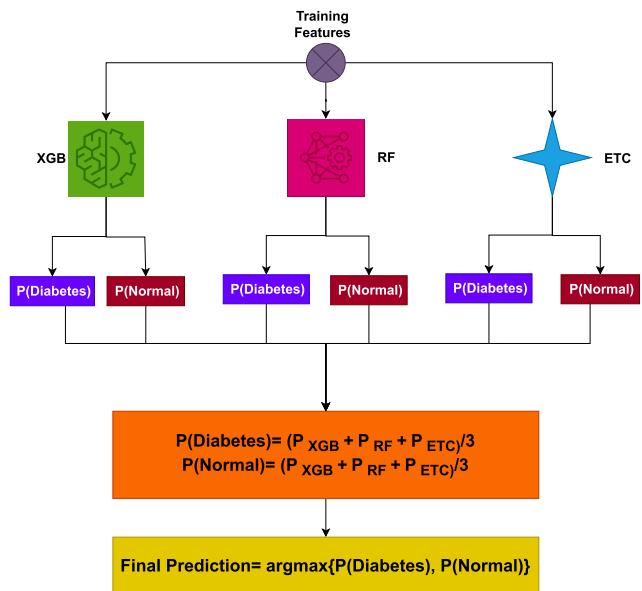
```

Input: input data  $(x, y)_{i=1}^N$ 
 $M_{XGB}$  = Trained_XGB
 $M_{RF}$  = Trained_RF
 $M_{ETC}$  = Trained_ETC

1: for  $i = 1$  to  $M$  do
2:   if  $M_{XGB} \neq 0$  &  $M_{RF} \neq 0$  &  $M_{ETC} \neq 0$  &  $training\_set \neq 0$  then
3:      $ProbXGB - 1 = M_{XGB}.probability(Diabetes - class)$ 
4:      $ProbXGB - 2 = M_{XGB}.probability(Normal - class)$ 
5:      $ProbRF - 1 = M_{RF}.probability(Diabetes - class)$ 
6:      $ProbRF - 2 = M_{RF}.probability(Normal - class)$ 
7:      $ProbETC - 1 = M_{ETC}.probability(Diabetes - class)$ 
8:      $ProbETC - 2 = M_{ETC}.probability(Normal - class)$ 
9:     Decision function =  $\max(\frac{1}{N_{classifier}} \sum_{classifier} (Avg(ProbXGB - 1, ProbRF - 1, ProbETC - 1), Avg(ProbXGB - 2, ProbRF - 2, ProbETC - 2))$ 
10:    end if
11:    Return final label  $\hat{p}$ 
12: end for
```

The ensemble model ultimately arrives at a final class prediction by selecting the one with the highest average probability score across the combined predictions of the individual classifiers. The probabilities produced by each

classifier are pooled together, and the class with the highest probability is designated as the final prediction. This approach leverages the collective knowledge of the individual models to arrive at a more robust and accurate prediction.



**FIGURE 2.** Architecture of the proposed voting classifier.

#### G. EVALUATION PARAMETERS

To assess the performance of a model, various commonly used evaluation metrics are employed. These include recall (sensitivity), precision, accuracy, and the F1 score. These metrics are computed using a confusion matrix, which consists of the following components: true negative (TN), false negative (FN), false positive (FP), and true positive (TP).

**True Positive (TP):** This occurs when the model correctly predicts a record as diabetic, and the actual label of the record is indeed diabetic.

**False Positive (FP):** This happens when the model incorrectly predicts a record as diabetic, while the actual label of the record is normal.

**True Negative (TN):** This takes place when the model accurately predicts a record as normal, and the actual label of the record is indeed normal.

**False Negative (FN):** This occurs when the model mistakenly predicts a record as normal, but the actual label of the record is diabetic.

These elements form the basis for evaluating the model's performance and provide valuable insights into its strengths and weaknesses.

The performance of the model can be assessed by computing recall, precision, accuracy, and F1 score using the values extracted from the confusion matrix. These assessment metrics are outlined below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

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

$$F1 - Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

#### IV. RESULTS AND DISCUSSION

Python 3.8 was used to develop the machine learning models in this study, and they were run in a Jupyter Notebook environment with the sci-kit Learn and Tensorflow libraries. The system used for the experiments had a 64-bit Windows 10 operating system and a 7th-generation Core i7 CPU with a speed of 2.8 GHz. The models' performance was measured using parameters like accuracy, precision, recall (sensitivity), and F1-score. These metrics are commonly used to evaluate how well machine learning algorithms can predict diabetes risk. The code snippet of the proposed model is as follows:

```
#' sign denotes comments to explain the code
from sklearn.impute import KNNImputer
#(importing KNN imputation library)
imputer = KNNImputer(n_neighbors=4)
# (Setting KNN imputation neighbouring value)
imputer.fit(X_train)
#(fitting training data for imputation)
from sklearn.ensemble import VotingClassifier
#(importing voting classifier library)
vc = VotingClassifier(estimators=[('XGB', XGB), ('RF', RF),
('ETC', ETC)], voting='hard')
#(making ensemble of 3 models)
VCPred=vc.fit(X_train,y_train).predict(X_test)
#(fitting training data on voting model and testing its
performance)
print("Voting accuracy score:", vc.accuracy_score(y_test,
VCPred))
#(calculating accuracy value of proposed model)
```

#### A. EXPERIMENTAL RESULTS BY DELETING MISSING VALUES

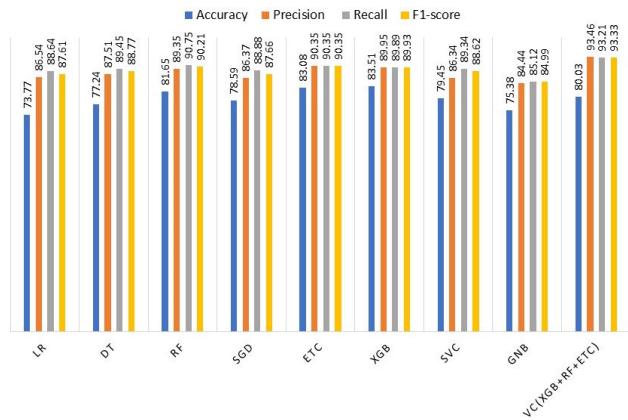
At the start of the experiments, missing values in the dataset are erased before applying machine learning models to the updated data. In machine learning, this is one of the most frequent methods for dealing with missing values. The performance of these models is detailed in Table 4, which provides a full assessment of their accuracy, precision, recall, and F1 score.

The classification report of learning models utilized in this research work is shown in Table 4. The highest accuracy rates are achieved by the RF, ETC, and XGB classifiers, that is 82.29%, 84.08%, and 84.51%, respectively. The RF classifier had a precision of 90.37%, a recall of 89.75%, and an F1 score of 89.21%. Likewise, the ETC classifier showed an accuracy, recall, and F1 score of 89.35%. The XGB model has a precision of 89.95%, a recall of 89.89%, and an F1 score of 89.93%. On the other hand, the LR classifier performed poorly, with an accuracy of 74.66%,

**TABLE 4.** Classification report of all learning models by deleting missing values.

Model	Accuracy	Precision	Sensitivity	F1-Score
DT	78.42	88.15	88.34	88.24
LR	74.66	87.39	87.57	87.48
SGD	79.59	87.37	89.88	88.66
RF	82.29	90.37	89.75	89.21
ETC	84.08	89.35	89.35	89.35
XGB	84.51	89.95	89.89	89.93
SVC	79.45	86.34	89.34	88.62
GNB	75.38	84.44	85.12	84.99
Tri-Ensemble Model	90.03	93.46	93.21	93.33

a precision of 87.39%, a recall of 87.57%, and an F1 score of 87.48%. The proposed Tri-Ensemble model surpassed all learning models. It had an accuracy of 90.03%, a precision of 93.46%, a recall of 93.21%, and an F1 score of 93.33%. The performance of individual models using the data with missing values removed is not satisfactory. Figure 3 displays the results of the machine learning models under this condition. It is clear that, except for RF, ETC, XGBoost, and Tri-Ensemble model, the performance of the other models is average.



**FIGURE 3.** Learning models results obtained by deleting missing values.

#### B. EXPERIMENTAL RESULTS USING KNN-IMPUTED DATASET

The second set of experiments deals with missing values in the data by using the KNN imputer to impute the missing values. The KNN imputer fills the gaps of missing values that are found to be missing during the preprocessing stage. The KNN imputer is a well-known method for imputing missing data, as it uses the k-nearest neighbor algorithm to estimate these values. This method blends the mean of existing values with the Euclidean distance metric to replace the missing data. The data with the KNN imputed values is then used to train and test various machine learning models. Table 5 shows the performance of these models, which used the data improved by the KNN imputer.

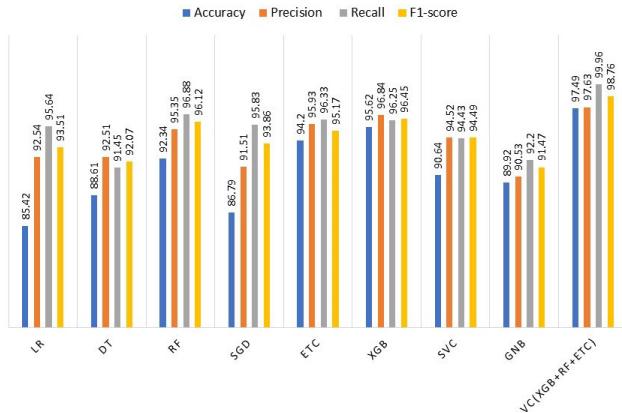
The results of the second set of experiments are shown in Table 5, where the KNN imputer is used to impute

**TABLE 5.** Classification report of all learning models using KNN imputer.

Model	Accuracy	Precision	Sensitivity	F1-Score
LR	85.42	92.54	95.64	93.51
DT	88.61	92.51	91.45	92.07
RF	92.34	95.35	96.88	96.12
SGD	86.79	91.51	95.83	93.86
ETC	94.20	95.93	96.33	95.17
XGB	95.62	96.84	96.25	96.45
SVC	90.64	94.52	94.43	94.49
GNB	89.92	90.53	92.20	91.47
<b>Tri-Ensemble Model</b>	<b>97.49</b>	<b>98.16</b>	<b>99.35</b>	<b>98.84</b>

missing values in the dataset. Again, RF, ETC, and XGBoost achieves good accuracy result of 92.32%, 94.20%, and 95.62%, respectively. These results support the claim of this research work that machine learning model performance is improved by using the KNN imputer to fix missing values instead of removing the missing values. The proposed Tri-ensemble model surpasses all other models with an accuracy of 97.49%. Moreover, the ensemble model had a recall of 99.35%, an F1 score of 98.84%, and an accuracy of 98.16%.

The linear model LR had the worst accuracy, with 85.42%. A visual display of the results from the machine learning models using the KNN imputer is given in Figure 4. The graph clearly shows that the performance of the machine learning models is improved by using the KNN imputer.

**FIGURE 4.** Learning models results obtained using KNN Imputer.

### C. PERFORMANCE COMPARISON OF BOTH SETS OF EXPERIMENTS

This research underscores a fundamental principle in data preprocessing: the significance of addressing missing values rather than resorting to their deletion. The findings unequivocally establish that employing an imputing technique, particularly the KNN imputer, is instrumental in enhancing the performance of machine learning models when confronted with missing data.

The second experiment, incorporating the KNN imputer, yielded marked improvements across all learning models, surpassing the outcomes achieved when missing values were

simply removed from the dataset. The detailed summary provided in Table 6 facilitates an in-depth examination of model performance under both conditions, offering a comprehensive perspective on the superiority of employing the KNN imputer.

This comparison effectively illuminates the positive impact and inherent advantages of integrating the KNN imputer in fortifying the predictive capabilities of the models. As acknowledged in the literature, various strategies exist for handling missing data, ranging from instance deletion to replacement with possible or approximated values. However, our study affirms that prioritizing the treatment of missing values before data analysis is paramount. Neglecting or deleting these values introduces biases and compromises the integrity of subsequent analyses, potentially leading to erroneous conclusions.

This research reinforces the pivotal role of thoughtful missing data handling, particularly through the application of the KNN imputer, as a crucial step in ensuring the reliability and accuracy of machine learning models in the realm of diabetes prediction.

**TABLE 6.** Accuracy comparison of all learning models using both sets of experiments.

Model	With KNN	Without KNN
LR	73.77	85.42
DT	77.24	88.61
RF	81.65	92.34
SGD	78.59	86.79
ETC	83.08	94.20
XGB	83.51	95.62
SVC	79.45	90.64
GNB	75.38	89.92
<b>Tri-Ensemble Model</b>	<b>80.03</b>	<b>97.49</b>

### D. K-FOLD CROSS VALIDATION RESULTS OF PROPOSED MODEL

K-fold cross-validation was used to improve the models' reliability. Table 7 shows that the proposed approach outperforms existing models in terms of accuracy, precision, recall, and F1 score after 5-fold cross-validation. Additionally, the proposed approach exhibits a low standard deviation, emphasizing its reliability and consistency. This indicates that the suggested approach consistently performs well across multiple folds, reinforcing confidence in its dependability and robustness.

**TABLE 7.** 5 fold cross-validation results of the proposed model.

Model	Accuracy	Precision	Sensitivity	F1-Score
1st-Fold	96.34	97.24	98.49	98.35
2nd-Fold	96.38	97.43	98.55	98.49
3rd-Fold	97.72	98.29	99.94	99.79
4th-Fold	97.08	98.78	99.99	99.85
5th-Fold	96.98	97.15	98.86	98.33
<b>Average</b>	<b>96.89</b>	<b>97.81</b>	<b>99.23</b>	<b>98.87</b>

## E. PERFORMANCE COMPARISON WITH EXISTING STUDIES

In order to assess the effectiveness of the proposed model compared to well-established state-of-the-art models, a thorough comparison was carried out with five pertinent research endeavors that focused on improving accuracy. These chosen works were used as benchmarks to gauge the efficacy of the proposed model and highlight its advancements over current methodologies. Through this juxtaposition of results, this study offers valuable insights into the superior performance of the proposed approach, particularly in the realm of accuracy enhancement. For instance, in [16], an ensemble model called LTC was introduced, achieving an accuracy score of 85%. In [21], an ensemble of deep-learning models was employed for diabetes prediction, resulting in the highest accuracy of 88.37%. Similarly, in [8], the individual learning model LGBM was utilized for diabetes prediction, achieving an accuracy of 92.5%. Table 8 offers a performance comparison between the proposed model and the existing studies. The results unequivocally demonstrate that the proposed model surpasses the existing models across various performance metrics.

**TABLE 8.** Performance comparison with state-of-the-art studies.

Ref	Technique	Accuracy
[16]	LTC (ensemble)	85%
[18]	LSTM	87.26%
[8]	LGBM	92.5%
[21]	CNN-BiLSTM	88.37%
[22]	DNN	86.26%
<b>Proposed</b>	<b>Tri-Ensemble Model</b>	<b>97.49%</b>

## V. CONCLUSION

In recent times, there has been a noticeable surge in the prevalence of diabetes, impacting millions of individuals worldwide. Timely interventions hold the key to mitigating the intricate complications associated with diabetes. Within this context, the utilization of machine learning techniques has shown promise in achieving higher levels of detection precision. This study introduced a framework that incorporates the KNN imputer during the data preprocessing phase to address missing values and employs an ensemble model to bolster classification accuracy. The stacked ensemble voting classifier (comprising XGBoost, Random Forest, and Extra Trees) exhibited experimental results with an impressive accuracy of 97.49%, underscoring the effectiveness of integrating the KNN imputer with an ensemble model for significantly improved outcomes. Additionally, the proposed model demonstrated superior performance when compared to other cutting-edge models. In future, the research aims to apply deep learning models in diabetes prediction. This approach is expected to yield further enhancements in the model's performance, particularly when dealing with datasets of larger dimensions, ultimately leading to more resilient and versatile results. This direction seeks to unlock further

performance enhancements, especially in handling larger datasets.

## REFERENCES

- [1] Diabetes Gojka. (Jul. 2019). *Diabetes: World Health Organization (WHO)*. Accessed: May 25, 2023.
- [2] S. El-Sappagh, F. Ali, S. El-Masri, K. Kim, A. Ali, and K.-S. Kwak, "Mobile health technologies for diabetes mellitus: Current state and future challenges," *IEEE Access*, vol. 7, pp. 21917–21947, 2019.
- [3] L. Mertz, "Automated insulin delivery: Taking the guesswork out of diabetes management," *IEEE Pulse*, vol. 9, no. 1, pp. 8–9, Jan. 2018.
- [4] H. A. Klein and A. R. Meining, "Self management of medication and diabetes: Cognitive control," *IEEE Trans. Syst., Man, Cybern., A, Syst. Hum.*, vol. 34, no. 6, pp. 718–725, Nov. 2004.
- [5] WHO. (Apr. 2023). *Diabetes: World Health Organization (WHO)*. Accessed: May 25, 2023.
- [6] A. A. Al Jarullah, "Decision tree discovery for the diagnosis of type II diabetes," in *Proc. Int. Conf. Innov. Inf. Technol.*, Apr. 2011, pp. 303–307.
- [7] G. D. Kalyankar, S. R. Poojara, and N. V. Dharwadkar, "Predictive analysis of diabetic patient data using machine learning and Hadoop," in *Proc. Int. Conf. I-SMAC*, Feb. 2017, pp. 619–624.
- [8] B. S. Ahamed, M. S. Arya, and A. O. V. Nancy, "Diabetes mellitus disease prediction using machine learning classifiers with oversampling and feature augmentation," *Adv. Hum.-Comput. Interact.*, vol. 2022, pp. 1–14, Sep. 2022.
- [9] S. Perveen, M. Shahbaz, A. Guergachi, and K. Keshavjee, "Performance analysis of data mining classification techniques to predict diabetes," *Proc. Comput. Sci.*, vol. 82, pp. 115–121, Jan. 2016.
- [10] I. Kavakiotis, O. Tsavos, and A. Salifoglou, "Machine learning and data mining methods in diabetes research," *Comput. Struct. Biotechnol. J.*, vol. 15, no. 9, pp. 104–116, 2017.
- [11] A. K. Bashir, N. Victor, S. Bhattacharya, T. Huynh-The, R. Chengoden, G. Yenduri, P. K. R. Maddikunta, Q.-V. Pham, T. R. Gadekallu, and M. Liyanage, "Federated learning for the healthcare metaverse: Concepts, applications, challenges, and future directions," *IEEE Internet Things J.*, vol. 10, no. 24, pp. 21873–21891, Mar. 2023.
- [12] U. Tariq, I. Ahmed, A. K. Bashir, and K. Shaukat, "A critical cybersecurity analysis and future research directions for the Internet of Things: A comprehensive review," *Sensors*, vol. 23, no. 8, p. 4117, Apr. 2023.
- [13] S. Saranya and S. Bobby, "COVID-19 patient health prediction using boosted random forest algorithm," *Data Anal. Artif. Intell.*, vol. 3, no. 2, pp. 64–68, Feb. 2023.
- [14] P. He, C. Lan, A. Kashif Bashir, D. Wu, R. Wang, R. Kharel, and K. Yu, "Low-latency federated learning via dynamic model partitioning for healthcare IoT," *IEEE J. Biomed. Health Informat.*, vol. 27, no. 10, pp. 4684–4695, Oct. 2023.
- [15] H. M. Deberneh and I. Kim, "Prediction of type 2 diabetes based on machine learning algorithm," *Int. J. Environ. Res. Public Health*, vol. 18, no. 6, p. 3317, Mar. 2021.
- [16] V. Rupapara, F. Rustam, A. Ishaq, E. Lee, and I. Ashraf, "Chi-square and PCA based feature selection for diabetes detection with ensemble classifier," *Intell. Autom. Soft Comput.*, vol. 36, no. 2, pp. 1931–1949, 2023.
- [17] Y. Deng, L. Lu, L. Aponte, A. M. Angelidi, V. Novak, G. E. Karniadakis, and C. S. Mantzoros, "Deep transfer learning and data augmentation improve glucose levels prediction in type 2 diabetes patients," *NPJ Digit. Med.*, vol. 4, no. 1, p. 109, Jul. 2021.
- [18] U. M. Butt, S. Letchumanan, M. Ali, F. H. Hassan, A. Baqir, and H. H. R. Sherazi, "Machine learning based diabetes classification and prediction for healthcare applications," *J. Healthcare Eng.*, vol. 2021, pp. 1–17, Sep. 2021.
- [19] G. A. Pethunachiyar, "Classification of diabetes patients using kernel based support vector machines," in *Proc. Int. Conf. Comput. Commun. Informat. (ICCCI)*, Jan. 2020, pp. 1–4.
- [20] U. E. Laila, K. Mahboob, A. W. Khan, F. Khan, and W. Taekeun, "An ensemble approach to predict early-stage diabetes risk using machine learning: An empirical study," *Sensors*, vol. 22, no. 14, p. 5247, Jul. 2022.
- [21] P. Madan, V. Singh, V. Chaudhari, Y. Albagory, A. Dumka, R. Singh, A. Gehlot, M. Rashid, S. S. Alshamrani, and A. S. AlGhamdi, "An optimization-based diabetes prediction model using CNN and bi-directional LSTM in real-time environment," *Appl. Sci.*, vol. 12, no. 8, p. 3989, Apr. 2022.

- [22] K. Kannadasan, D. R. Edla, and V. Kuppili, "Type 2 diabetes data classification using stacked autoencoders in deep neural networks," *Clin. Epidemiol. Global Health*, vol. 7, no. 4, pp. 530–535, Dec. 2019.
- [23] A. Dutta, M. K. Hasan, M. Ahmad, M. A. Awal, M. A. Islam, M. Masud, and H. Meshref, "Early prediction of diabetes using an ensemble of machine learning models," *Int. J. Environ. Res. Public Health*, vol. 19, no. 19, p. 12378, Sep. 2022.
- [24] UCI Machine Learning. (Jul. 2016). *Diabetes: World Health Organization (WHO)*. Accessed: May 5, 2023.
- [25] U. Hafeez, M. Umer, A. Hameed, H. Mustafa, A. Sohaib, M. Nappi, and H. A. Madni, "A CNN based coronavirus disease prediction system for chest X-rays," *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 10, pp. 13179–13193, Oct. 2023.
- [26] A. Juna, M. Umer, S. Sadiq, H. Karamti, A. A. Eshmawi, A. Mohamed, and I. Ashraf, "Water quality prediction using KNN imputer and multilayer perceptron," *Water*, vol. 14, no. 17, p. 2592, Aug. 2022.
- [27] Y. Zhang, H. Zhang, J. Cai, and B. Yang, "A weighted voting classifier based on differential evolution," *Abstract Appl. Anal.*, vol. 2014, pp. 1–6, Jan. 2014.
- [28] M. Brijain, R. Patel, M. R. Kushik, and K. Rana, "A survey on decision tree algorithm for classification," *Int. J. Eng. Develop. Res.*, vol. 2, no. 1, pp. 1–5, 2014.
- [29] M. Karim, M. M. S. Missen, M. Umer, S. Sadiq, A. Mohamed, and I. Ashraf, "Citation context analysis using combined feature embedding and deep convolutional neural network model," *Appl. Sci.*, vol. 12, no. 6, p. 3203, Mar. 2022.
- [30] F. Sebastiani, "Machine learning in automated text categorization," *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1–47, Mar. 2002.
- [31] B. Zadrozny and C. Elkan, "Transforming classifier scores into accurate multiclass probability estimates," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2002, pp. 694–699.
- [32] B. Gregorutti, B. Michel, and P. Saint-Pierre, "Correlation and variable importance in random forests," *Statist. Comput.*, vol. 27, no. 3, pp. 659–678, May 2017.
- [33] F. Rustam, I. Ashraf, A. Mehmood, S. Ullah, and G. Choi, "Tweets classification on the base of sentiments for U.S. airline companies," *Entropy*, vol. 21, no. 11, p. 1078, Nov. 2019.
- [34] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Trans. Syst., Man, Cybern.*, vol. 21, no. 3, pp. 660–674, Jun. 1991.
- [35] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.



**KHALED ALNOWAISER** received the Ph.D. degree in computer science from Glasgow University, Scotland. He is currently an Assistant Professor with the Computer Engineering Department, Prince Sattam bin Abdulaziz University, Saudi Arabia. His research interests include computer vision, optimization techniques, and performance enhancement.

• • •