A Hybrid Big Data Analytics Approach for Predicting Type II Diabetes Using H-SMOTE Tree

Praveenkumar K S[1], R Gunasundari[2]

[1]PhD Research Scholar, Dept. of CS, CA & IT, Karpagam Academy of Higher Education, Coimbatore, India, praveen7387@gmail.com
[2]Professor & Head , Dept. of Computer Applications, Karpagam Academy of Higher Education, Coimbatore, India, gunasoundar04@gmail.com

Effective prediction models are required as the global prevalence of Type II Diabetes rises. This study proposes a hybrid Big Data analytics technique for predicting Diabetic Type II. The three key phases are data preparation using an Amalgam Multivariate Statistical Modeling Algorithm, feature extraction with Decision-Making Weighted Feature Selection, and D-H-SMOTE Tree classification. An Amalgam Multivariate Statistical Modeling Algorithm is used to preprocess the massive diabetic patient datasets. To handle data complexity and intricacies, this application employs a variety of statistical models. This research improves data quality and dependability to prepare it for analysis. The second step extracts features using Decision-Making Weighted Feature Selection. This research assesses characteristics based on their predictive power for Type II Diabetes using decision-making techniques. This phase reduces dimensionality and retains just the most relevant characteristics, enhancing prediction model efficiency and interpretability. Third, train the model using Artificial Neural Networks. ANNs can learn complicated data patterns and correlations. The trained model underpins categorization. In the last step, this research presents D-H-SMOTE Tree, a new categorization method. This approach addresses diabetes dataset class imbalance by combining SMOTE with decision trees. Oversampling and decision trees improve the model's generalization and classification, particularly with unbalanced class distributions.

Keywords: Artificial Neural Networks, Diabetic, D-H-SMOTE Tree, Weighted Feature Selection.

1. Introduction

"Big Data" is a phrase that is often used to describe very massive and complex datasets that outstrip the storage, processing, and computing capabilities of traditional databases and data analysis methodologies [1]. The investigation, extraction, and confirmation of various patterns from enormous volumes of unstructured data is fundamental to the use of Big Data as a resource. As data storage capacity, computing power, and the availability of increasing volumes of data continue to expand, companies are being presented with more data than they

can manage. This has led to the creation of "Big Data" [2-3]. Additional challenges come from dealing with volumes, varieties, velocities, and veracity of data, which are known as the four Vs of Big Data [4]. By discovering and verifying preexisting patterns, data mining is able to extract valuable insights from massive datasets [5]. Low or high insulin levels in the bloodstream cause the metabolic disease known as diabetes, which is defined by consistently high blood sugar levels. In 2010, the global population with diabetes was predicted to reach 285 million. More than 6% of adults, or 552 million people, will fall into this category by 2030, according to projections [6]. If diabetes continues to worsen at its present rate, one in ten people will have the condition by 2040. Present estimates place the prevalence of diabetes in India at 13.7%, with over 25% of the population exhibiting pre-diabetic signs [7]. Underreporting of diabetes occurs because one-third of people with the condition do not know they have it or do not experience any symptoms [8]. Diabetes, if left untreated, may cause a cascade of issues in several sections of the body, including the eyes, kidneys, heart, nerves, and blood vessels. Early detection allows those at risk to take proactive steps to limit the disease's development and improve their quality of life [9].

It is now widely accepted that neural networks outperform statistical and machine learning approaches. With the help of ANNs, a wide range of problems have been effectively resolved [10]. Neural networks are able to handle nonlinear problems and mimic human behavior, which is leading to their usage in solving more complex systems [11]. The building blocks of a neural network are the nerve cells, which process data [12]. Diabetes is diagnosed by having an abnormally high blood sugar level. Sugar cannot be transferred from the circulation to cells that need it when the pancreas does not produce enough insulin (Type I diabetes) or when insulin is ineffective (Type II diabetes) [13–14]. Chronic hyperglycemia increases the danger of renal failure, blindness, and heart attacks. People with diabetes must check their blood glucose levels often to prevent or postpone the onset of these consequences. The objective of this research is to examine the Pima Indian Diabetes database [15]. The sheer volume of data makes this the most challenging machine learning task to date. A number of strategies based on MV analysis and preprocessing processes exist for improving classification and reducing noise [16]. There has been an examination of both MV and preprocessing. Among metabolic diseases, diabetes is the most lethal and poses a serious threat to both developed and developing nations [17]. A high blood glucose level characterizes the condition. The condition is caused by a faulty insulin system[18]. Insulin is essential for cells to take in glucose and produce energy. Among the top causes of mortality globally, diabetes is particularly prevalent in underdeveloped countries [19]. According to the World Health Organization, eighty percent or more of the deaths in Ebola-affected countries occur in low- and middle-income countries that lack both basic and sophisticated healthcare services. The "diabetes capital of the world" is located in India, a growing country with a huge diabetic population [20-23].

The main contribution of the paper is:

- Data preprocessing using Amalgam Multivariate Statistical Modeling Algorithm
- Feature extraction using Decision-Making Weighted Feature Selection
- Training using ANN

- Classification using D-H-SMOTE Tree

The remainder of this paper is arranged as follows. A number of authors describe various ways to diabetes diagnosis in Section 2. Section 3 depicts the proposed model. Section 4 of this study summarized the investigation's results. Section 5 is devoted to a discussion of the findings and prospective future study.

1.1 Motivation of the paper

The increasing number of people diagnosed with Type II Diabetes throughout the world has prompted this research because it highlights the critical need of having reliable prediction models. There is an urgent need for novel approaches developed specifically for Big Data analytics since existing techniques struggle to make sense of the enormous and complicated information linked to diabetes patients. The fact that it employs cutting-edge approaches for data preprocessing, feature extraction, and classification, as well as tackling the challenges of large-scale data, motivates us to give an all-encompassing hybrid approach. This research hope that by combining the Amalgam Multivariate Statistical Modeling Algorithm, Decision-Making Weighted Feature Selection, and the new D-H-SMOTE Tree classification method, this research can improve the precision and consistency of Type II Diabetes forecasts.

2. BACKGROUND STUDY

Ameena, R. R., & Ashadevi, B. [1] The primary goal of this study was to investigate female diabetes by comparing several prediction models, establishing their accuracy, and generating statistical findings using the R statistical software. According to the classification findings, random forest was the most effective method. Additionally, the random forest was able to overcome the overfitting issue caused by missing values in the datasets, thanks to its improved classification performance.

Chatragadda, B. [4] these authors research aims to use big data analysis to learn more about diabetes treatment in the healthcare sector. Medical service providers can benefit from up-to-date information and analysis provided by a diabetes treatment plan including prospective research. The author have used Spark to predict the most common types of diabetes, as well as the racial and sexual orientation groups most likely to be impacted by the disease.

Esteban, S. [6] They explored multiple methods for building diabetes phenotyping algorithms using data acquired from an EHR system in Buenos Aires, Argentina. The validation set's top classification metrics were produced using the feedforward neural network and the layered generalization technique. With these algorithms in place, the author can accurately use the data from thousands of patients. It would be very difficult, if not impossible, to get financing for investigations of this scale in many nations.

Hariharakrishnan, J. et al. [10] the current state of data cleaning methods was briefly reviewed in this survey. To summarize, data cleaning procedures were designed to identify and eliminate small-scale mistakes and discrepancies that arise from faulty data gathering processes and different types of data, whether they were homogenous or heterogeneous. In

fact, the majority of conventional data cleansing methods were exclusive to the data gathering phase.

Islam, M. et al. [13] these authors study's results demonstrate that DL can make a significant contribution to the accurate and reliable detection of referable DR. Future DR diagnoses influenced by the use of an automated system based on DL. It was possible that automated technologies can decrease the cost of screening, increase accessibility to healthcare, and enhance the quality of DR screening. The development of the illness or its course can be averted or slowed down if it were detected and treated early.

Kadhim, A. I. et al. [15] these authors research presents a novel method for reducing high dimensionality that combines TF-IDF and SVD. Clustering related subjects using a hybrid approach and clustering was obviously possible. Unsupervised learning machines rely heavily on clustering-based K-means algorithms. From 9,636 for BBC news and 4,613 for BBC sport, the dimensions were lowered to 100 in this work, and the clustering accuracy was almost the same.

2.1 Problem definition

The escalating global prevalence of Type II Diabetes poses a significant public health challenge, necessitating the development of effective predictive models for early diagnosis and intervention. However, the inherent complexities of extensive diabetic datasets, coupled with the challenge of class imbalance, hinder the accuracy and reliability of existing prediction methods. Traditional approaches struggle to adequately preprocess large-scale data, extract informative features, and address class imbalances. Consequently, the need arises for an innovative solution tailored for Big Data analytics, prompting our study to propose a hybrid methodology.

3. MATERIALS AND METHODS

In this section, this research present a comprehensive approach for predicting Diabetic Type II leveraging a hybrid methodology designed for Big Data analytics. The materials utilized include extensive datasets of diabetic patients, while the methods consist of a three-stage process. First, data preprocessing employs the Amalgam Multivariate Statistical Modeling Algorithm to enhance data quality and reliability. Subsequently, feature extraction utilizes Decision-Making Weighted Feature Selection to reduce dimensionality and improve model efficiency. The proposed model flow chart has represented at figure 1.

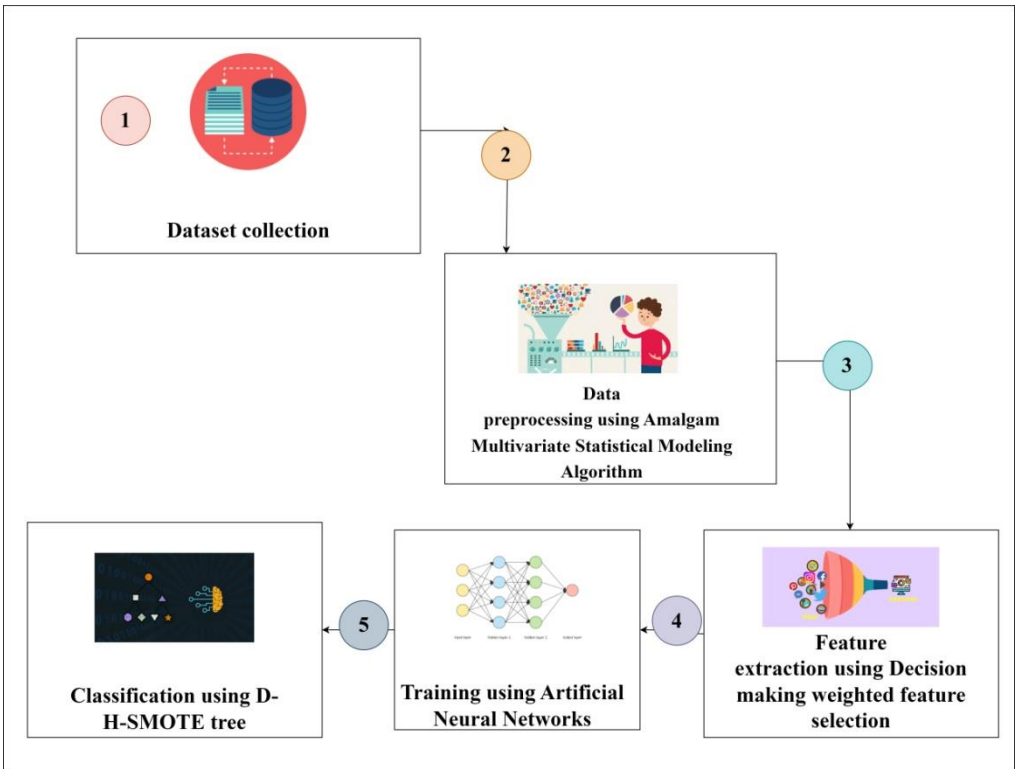

Figure 1: proposed workflow architecture

3.1 Dataset collection

Kaggle https://www.kaggle.com/uciml/pima-indians-diabetes-database The National Institute of Diabetes and Digestive and Kidney Diseases was the first to create such a database. The dataset's goal is to use certain diagnostic features to obtain a diabetes diagnosis based on a patient's medical history.

3.2 Data preprocessing using Amalgam Multivariate Statistical Modeling Algorithm

After acquiring the dataset, the Amalgam Multivariate Statistical Modeling Algorithm was employed for data preparation in this research.

The Amalgam Multivariate Statistical Modeling Algorithm is used in the data preparation stage of Wang, J.'s (2023) work to improve the quality and reliability of large datasets related to Type II Diabetes. This strategy captures and manages the underlying complexity and nuances of the data in a comprehensive and flexible way by merging numerous statistical models. The Amalgam Multivariate Statistical Modeling Algorithm integrates multiple statistical methodologies to effectively preprocess data. Principal component analysis, clustering approaches, regression analysis, and other techniques are among them. This procedure is used to prepare the dataset for analysis and feature extraction by normalizing the variables, minimizing noise, and dealing with outliers. Using this advanced algorithm helps to better depict the patterns in the data of diabetes patients, which in turn provides a solid basis for the next steps in the predictive modeling process.

By preprocessing inputs and goals, neuronal activity training made more effective. A network input processing service's principal function is to transform incoming data into a format more suited to the network. Data preparation for training is greatly affected by the method of raw input data normalization. Without this normalization, neural network training would have been much slower. There is more than one approach to data normalization. One method for eliminating bias in the neural network is to scale the data such that all input characteristics have the same range of values. When all features are initially trained on the same scale, data normalization can assist save training time. This is great for models that take into account data from a wide variety of scales. Some of the most often used techniques for data normalization in this area are as follows:

By calculating a Z-score from the means and standard deviations of each feature, the normalization approach standardizes all input feature vectors. For every attribute, this research determines the average and standard deviation equation 1 shows a change.

$$x_i = \frac{(x_i - \mu_i)}{\sigma_i} \text{---------- (1)}$$

The result is a mean with no significant data and a standard deviation that is biased to one side. A fresh training set is created and feature vectors from an existing dataset are normalized before training begins. The final system design should take into account the averages and standard deviations of each feature in the training data and utilize them as weights. The architecture of neural networks includes a preprocessing layer. An abnormally high or low performance of a neural network relative to its normalized equivalent could occur for many different causes. Applying these statistical criteria will reduce the impact of data outliers. To make sure that two sets of values are consistent, convert features or outputs between them. Depending on the scale used, the values are often rescaled to a range of -1 to 1. One common method for scaling is to use a formula based on linear interpretation, such

$$x_i = \left(\max_{target} - \min_{target}\right) X \frac{(x_i - \min_{value})}{\max_{value} - \min_{value}} + \min_{target} \text{---------- (2)}$$

So that zero is the absolute value of all conceivable values. This symbol denotes the constant value of a data characteristic if its maximum value minus its lowest value is zero. Constant-value features should be eliminated from the data since they give no information to the neural network. When min-max normalization is used, the former value of any attribute that falls inside the newly defined range is retained. All associations are kept in data is standardized using min-max. Another method for scaling network inputs and goals is to normalize the standard deviation and mean of the training set.

Assume that a random vector x is modeled as:

$$x = \sqrt{zu = su} \text{--------- (3)}$$

To which z is a scalar random number and u is a zero-mean Gaussian vector. A possible expression for the PDF of x is

$$fx(x) = \int_0^\infty fx(x|s)fs(s)ds \text{---------- (4)}$$

$$ds = \int_0^\infty \frac{1}{s} fU\left(\frac{x}{s}\right) fs(s)ds \text{---------- (5)}$$

3.3 Feature extraction using Decision making weighted feature selection

After preprocessing, this research use Decision making weighted feature selection for data feature selection

Decision-Making Weighted Feature Selection is used in this study's feature extraction step to improve the efficiency and interpretability of the Diabetic Type II prediction model. In this method, characteristics are given weights according to how well they predict the target variable referred by Tang et al. (2023). The algorithm prioritizes the most informative features by systematically evaluating their value using decision-making processes. In addition to lowering dimensionality, our method makes sure that the remaining characteristics improve prediction accuracy overall. The end result is a collection of characteristics that is easier to comprehend and use, which improves the model's accuracy and helps identify the most important variables in predicting T2D.

The important measure of each feature varies in weight, but M sample-D data sets can get the measure of M features. This research can calculate the feature's relevance by averaging the pre- and post-noise interference classification accuracy differences, multiplied by 10. The following formula is used to compute the important measurement value (FIM_{ij}) of each feature:

$$FIM_{ij} = \frac{\sum_{k=1}^{10} |A_{ik} - A_{ijk}|}{10} \text{ ---------- (6)}$$

Where i is the i^{th} Sample-D data set and j is the j^{th} feature according to the accuracy of classification before feature noise addition is represented byA_{ik}; the A_{ijk} measure represents the feature classification accuracy when noise is present.

To determine the decision tree's weight and run the random forest model's predictions using the OOB data set as test data. The formula for the decision tree's weight ($TreeWeight_i$) is as follows:

$$TreeWeight_i = \frac{\sum_{j=1}^{P} I(tree_{ij} = EnsTree_j)}{P} \times \text{RF Acc} \text{ ---------- (7)}$$

P stands for the total number of samples; The characteristic function, or "I," is a concept in probability theory; $tree_{ij}$ stands for the i^{th} decision tree's projected outcomes for the j^{th} sample; Represented by $EnsTree_j$ are the expected outcomes of the j-th sample's random forest; The RF Acc measures how well the random forest model predicts future outcomes.

Algorithm 1: Decision making weighted feature selection
Input: • Preprocessed dataset ($D_{preprocessed}$). Steps: 1. Initialization: o Define the number of features in the dataset(N).

- Initialize the decision tree weights for each sample ($\mathrm{TreeWeight_i}$) and the final feature importance measure ($\mathrm{FIM_{ij}}$).

2. Loop through Samples:

- For each Sample-D dataset (i):
  - Split the dataset into training and testing sets.
  - Train a random forest model using the training set.
  - Use the Out-of-Bag (OOB) dataset as the test data for decision tree weight calculations.

3. Calculate Decision Tree Weight:

- For each decision tree (j) in the random forest:
  - Calculate the decision tree weight ($\mathrm{TreeWeight_i}$) using:

$$\mathrm{TreeWeight_i} = \frac{\sum_{j=1}^{P} \mathrm{I}(\mathrm{tree_{ij}} = \mathrm{EnsTree_j})}{P} \times \mathrm{RF\ Acc}$$

4. Compute Feature Importance Measure:

- For each feature (j):
  - Calculate the feature importance measure (FIM_ij) using:

$$\mathrm{FIM_{ij}} = \frac{\sum_{k=1}^{10} |\mathrm{A_{ik}} - \mathrm{A_{ijk}}|}{10}$$

Output:

- Selected features with associated weights.

3.4 Training using Artificial Neural Networks

Following feature selection, the ANN algorithm is used to train the dataset in this study. The training part of this study employs Artificial Neural Networks (ANN) alluded to by Jaloli, M., and Cescon, M. (2023) to grasp detailed patterns and correlations within the data for the goal of predicting Type II Diabetes. ANNs use connected nodes, often known as neurons, stacked in layers to imitate the architecture of the human brain. To lessen the difference between predicted and actual outcomes, the network learns to change its weights and biases repeatedly. Because of their adaptive learning capabilities, artificial neural networks (ANNs) excel at capturing complex, non-linear interactions. Next steps are based on the trained ANN, which gives a thorough and data-driven picture of the patterns linked to Type II Diabetes.

Architectures: A basic component of an ANN is a system of interconnected "neurons" or "nodes." Each node performs a transfer function in the form, similar to a directed network.

$$y_i = f_i\left(\sum_{j=1}^{n} w_{i,j}\, x_j - \theta_i\right) \text{ ------ (8)}$$

In a network, the direction of a node's output depends on the strength of its connections to other nodes and the value of input. Typically, a nonlinear function such as the heaviside, sigmoid, or Gaussian represents the node's threshold (or bias).

Artificial neural networks (ANNs) categorized as feedforward or recurrent depending on the nature of their connections. A network understood to be feedforward if its nodes can be numerically assigned in such a manner that no node with a big number can interact with a node with a small number. Connection after connection goes from node to node, from the smallest to the largest. This research state that ANNs are periodic if there is no trustworthy method to number them.

With respect to (9) every accumulating phrase just requires one input. ANNs are considered to be of a higher order when they include nodes of a given order, meaning that the summation of certain terms requires more than one input. Determining the second-order node in one manner is as

$$y_i = f_i\left(\sum_{j,k=1}^{n} w_{i,j}\, kx_j x_k - \theta_i\right) \text{ ----- (9)}$$

where all the symbols have similar definitions to those in (9).

An ANN's design is ultimately dictated by its topological structure, which is the totality of the connections and transfer functions of all nodes in the network.

Algorithm 2: Artificial Neural Networks

Input:

- Feature-selected dataset (D_feature_selected).

Steps:

1. Initialization:
   - Define the architecture of the ANN, including the number of layers, nodes in each layer, and activation functions.
   - Initialize weights (w) and biases (θ) with small random values.
2. Feedforward Propagation:
   - For each instance in the dataset:
     - Compute the weighted sum of inputs for each node in each layer using:

$$y_i = f_i\left(\sum_{j=1}^{n} w_{i,j}\, x_j - \theta_i\right)$$

     - Apply the activation function to obtain the output of each node.
3. Backpropagation:
   - Calculate the error between the predicted output and the actual target.
   - Update weights and biases iteratively using gradient descent to minimize the error.

$y_i = f_i\left(\sum_{j,k=1}^{n} w_{i,j}\, kx_j x_k - \theta_i\right)$

- Iterative Learning:

- Repeat steps 2-3 for multiple epochs until the model converges or reaches a predefined stopping criterion.

Output:

- Trained ANN model with optimized weights and biases.

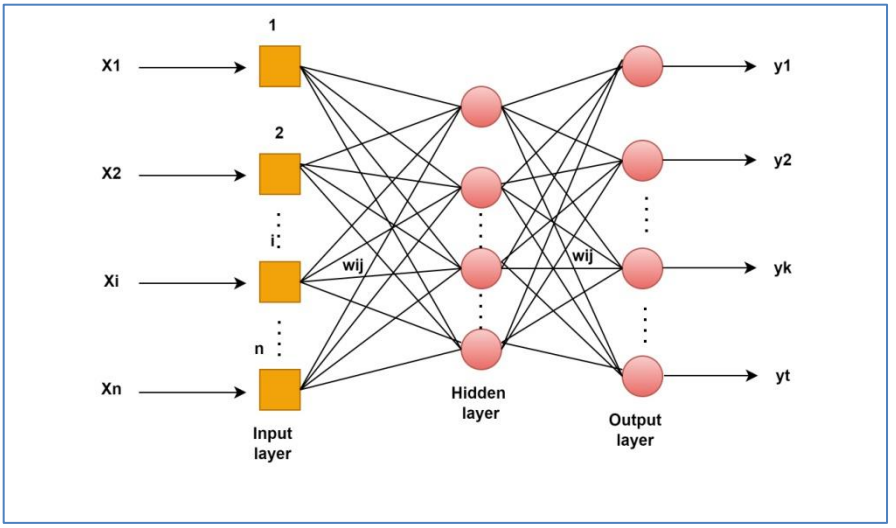

Figure 2: ANN architecture

3.5 Classification using D-H-SMOTE tree

3.5.1 H-SMOTE tree

After training process this research use H-SMOTE tree for diabetic classification. In order to tackle the problem of class imbalance that is common in diabetes datasets, this work presents a novel method for classification called H-SMOTE Tree. This method combines the SMOTE with decision trees. The goal of H-SMOTE Tree is to improve the model's generalizability and instance classification accuracy, particularly when dealing with unequal class distributions. Decision trees provide a systematic framework for classification, and SMOTE is used to oversample the minority class, which helps to reduce the effects of class imbalance. The higher prediction accuracy for recognizing incidences of Type II Diabetes in real-world circumstances is a result of this hybridization, which guarantees a more robust and balanced categorization model.

In order to address the shortcomings of using only one learning model or statistical method, this study develops a hybrid classification approach. It is impossible for a classification model to adequately process every kind of data. Every layer of a hybrid intelligent system contributes fresh knowledge to the one below it. Consequently, the model's overall functionality is enhanced by the accurate functionality at all levels. Two methods for

achieving dataset parity—SMOTE and resample—serve as inspiration for this hybrid classification strategy. For this balanced dataset, the attribute selection method is used. Although they operate in distinct ways, both sampling procedures are efficient. Overlapping and noise are outcomes of SMOTE's failure to account for nearby instances from various classes while building synthetic instances. By considering under sampling in resembling, this research runs the danger of over-fitting due to the repetition of rare class occurrences and the discarding of potentially crucial helpful information. It reduces the dimension for improved classification predictions by automatically excluding less valuable features. The classification process is in underway for this balanced and dimensionally reduced dataset. Using a k NN classification model, this study validates the suggested approach. Consequently, k NN is now used for dataset classification on the modified dataset. However, this approach is also effective with other types of classification models. Figure 4 depicts the suggested design.

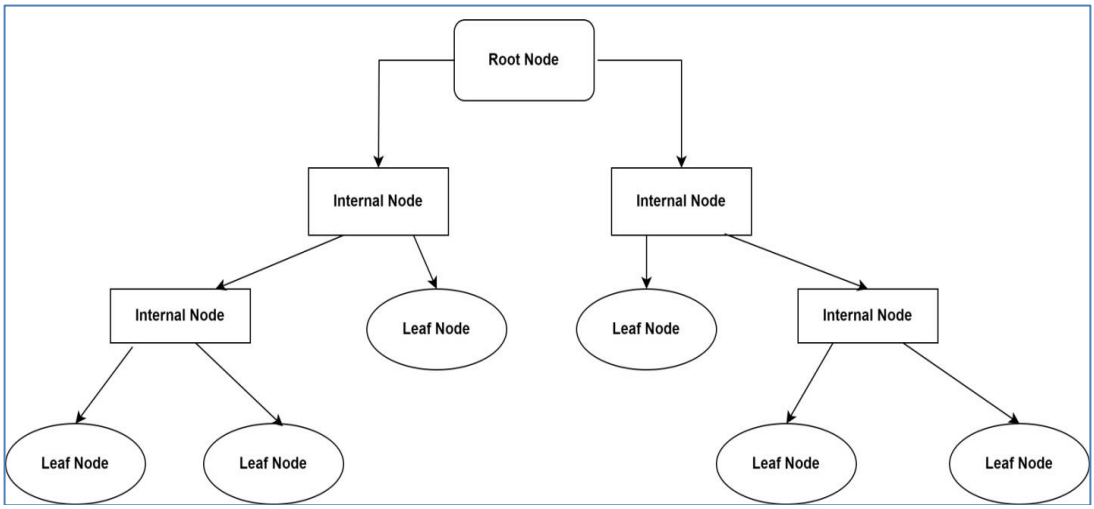

Figure 3: SMOTE tree architecture

Algorithm 3: H-SMOTE tree

Input:

Imbalanced dataset (B_i).

Steps:

Load the imbalanced dataset (B_i)

Apply H-SMOTE:

- Use SMOTE to oversample the minority class.
- Implement Resample method for further balancing.
- Employ Attribute Selection to reduce dimensionality.

Let D be the dataset, x_i be the minority class instance, and x_j be its k-nearest

neighbor from the same class.

The synthetic instance x_i' is generated using the formula: $x_i' = x_i + (x_j - x_i) * \text{random}_{\text{uniform}}(0,1)$.

Output:

Get a balanced dataset with chosen characteristics and k NN classification results.

3.5.2 D-H-SMOTE

The D-H-SMOTE Tree is a classification method that combines the Synthetic Minority Over-sampling Technique (SMOTE) with decision trees to handle class imbalance in datasets. SMOTE Tree attempts to enhance model performance in predicting circumstances such as Type II Diabetes, when incidences of the minority class (e.g., positive cases) are much lower than instances of the majority class. The Synthetic Minority Over-sampling Technique (SMOTE) generates synthetic cases for the minority class by interpolating existing instances. SMOTE Tree, when paired with decision trees, harnesses the capabilities of decision structures to categorize instances based on both the original and synthetic data attributes. This integration improves the model's capacity to deal with skewed class distributions, allowing for better generalization and accuracy in predicting cases of interest. SMOTE Tree offers a strong solution for instances in which standard classification models may fail owing to class imbalance, resulting in more accurate predictions in real-world applications.

D-H-SMOTE builds synthetic instances for the minority class to correct class imbalance. The technique starts by selecting a minority class instance and its k closest neighbors, then building synthetic instances along the line segments that connect the chosen instance and its neighbors. The general formula for creating a synthetic instance, x_i', from an original instance, x_i, and one of its neighbors, x_j, is:

$$x_i' = x_i + (\text{random}_{\text{value}}) \times (x_j - x_i)$$

Algorithm 4: D-H-SMOTE tree

Input:

- Imbalanced dataset with minority and majority classes (D).

Steps:

1. Initialization:
   - Set the oversampling factor(k).
   - Identify the minority class instances (D_{minority}) and majority class instances (D_{majority}).
2. Loop through Minority Instances:
   - For each minority instance x_i in D_{minority}:

$$x_i' = x_i + (\text{random}_{\text{value}}) \times (x_j - x_i)$$

- Identify its k nearest neighbors from the same class ($D_{neighbors}$).
- Calculate the number of synthetic instances to generate ($n_{synthetic}$) based on the oversampling factor.

Output:

- Balanced dataset with synthetic instances (D').

4. RESULTS AND DISCUSSION

This section displays the results of our hybrid strategy for Diabetic Type II prediction and delves into a detailed explanation of the findings. The study's multi-faceted methodology, encompassing advanced data preprocessing, feature extraction, and classification techniques, is evaluated for its efficacy in handling large-scale datasets.

$$\text{Accuracy} = \frac{\text{TP+}}{\text{TP+TN+FP+FN}} \text{ ---------- (10)}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \text{ ---------- (11)}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \text{ ---------- (12)}$$

$$\text{Fmeasure} = \frac{\text{2.Precision x recall}}{\text{Precision+Recall}} \text{ ---------- (13)}$$

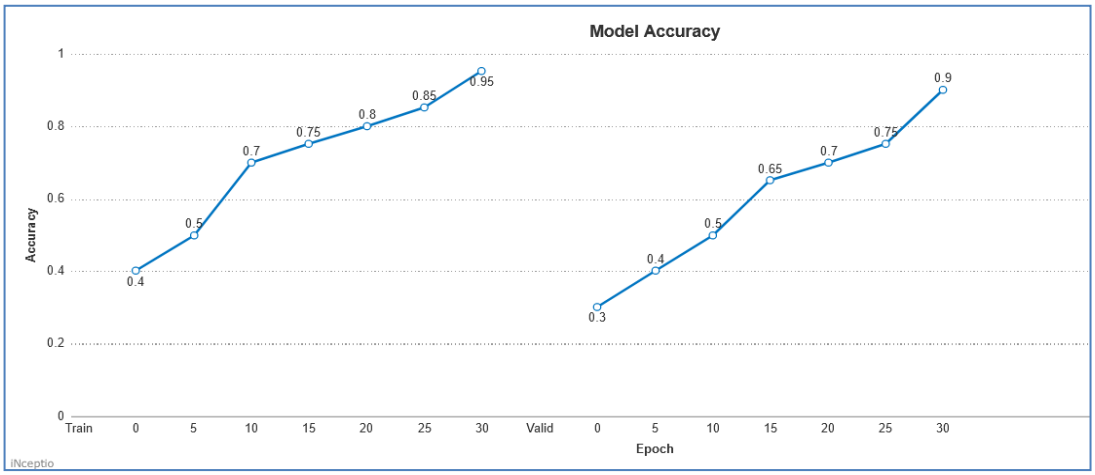

Figure 4: ANN Training and validation accuracy

Figure 5: Training and testing loss values

Table 1 displays the ANN training and testing values. Figure 3 depicts a comparison chart of training and testing accuracy. The X-axis represents the number of training and validation epochs. The precision is shown by the Y-axis. Figure 4 depicts the training and validation loss values.

Table 1: Classification of performance metrics

	Algorithm	Accuracy	Precision	Recall	F-measure
Existing authors	Cao, P. et al.	0.95	0.88	0.88	0.87
	Esteban, S. et al.	0.95	0.84	0.86	0.85
Existing methods	RF	0.81	0.81	0.83	0.80
	SMOTE	0.96	0.91	0.90	0.80
	H-SMOTE	0.97	0.93	0.90	0.70
Proposed methods	D-H-SMOTE	0.99	0.98	0.96	0.97

The table 1 presents the performance metrics of various algorithms in a classification task. In terms of Accuracy, the existing authors' methods, represented by Cao, P. et al. and Esteban, S. et al., achieve high scores of 0.95 each. The Existing Methods, Random Forest (RF), and Synthetic Minority Over-sampling Technique (SMOTE) also demonstrate respectable Accuracy scores of 0.81 and 0.96, respectively. The Heterogeneous Synthetic Minority Over-sampling Technique (H-SMOTE) outperforms with an Accuracy of 0.97. The proposed method, D-H-SMOTE, surpasses all others, reaching an impressive Accuracy of 0.99. Moving on to Precision, the precision of the existing authors' methods remains competitive, with Cao, P. et al. achieving 0.88 and Esteban, S. et al. at 0.84. RF and SMOTE exhibit similar Precision values of 0.81 and 0.91, respectively. H-SMOTE shows an

improvement in Precision at 0.93, while D-H-SMOTE achieves the highest Precision score of 0.98, showcasing its effectiveness in minimizing false positives. In terms of Recall, Cao, P. et al. and Esteban, S. et al. again demonstrate strong performance with values of 0.88 and 0.86, respectively. SMOTE leads in Recall among existing methods with a score of 0.90, while H-SMOTE closely follows at 0.90. D-H-SMOTE achieves the highest Recall at 0.96, indicating its ability to effectively capture a higher proportion of positive instances. Finally, considering the F-measure, Cao, P. et al. and Esteban, S. et al. achieve balanced scores of 0.87 and 0.85, respectively. Among existing methods, RF and SMOTE have matching F-measure values of 0.80. H-SMOTE falls slightly behind with an F-measure of 0.70. The proposed method, D-H-SMOTE, once again leads with an outstanding F-measure of 0.97, reflecting its proficiency in achieving a balance between precision and recall. Overall, the results suggest that D-H-SMOTE stands out as a promising algorithm across multiple performance metrics.

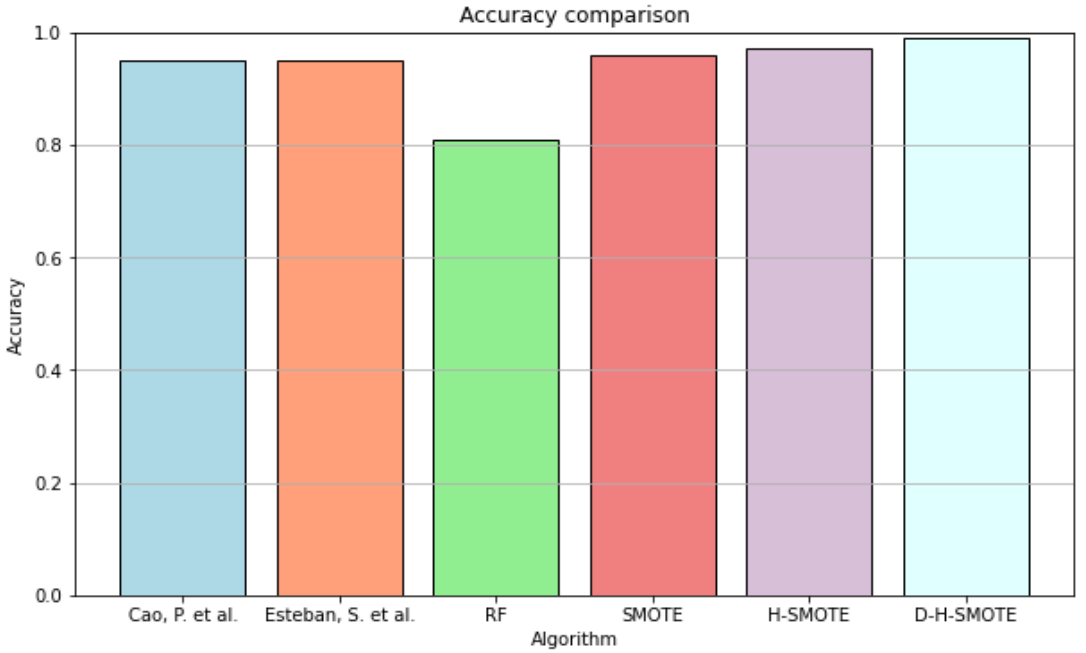

Figure 6: Accuracy comparison chart

The figure 6 shows accuracy comparison chart the x axis shows algorithm and the y axis shows accuracy values.

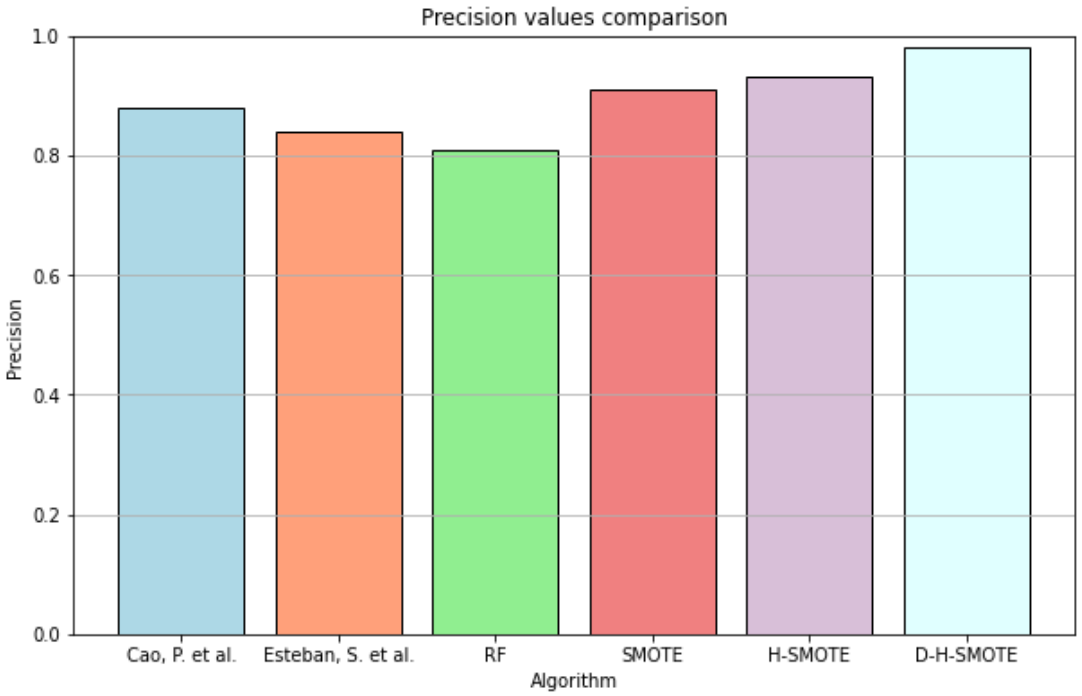

Figure 7: Precision values comparison chart

The figure 7 shows Precision values comparison chart the x axis shows algorithms and the y axis shows precision values.

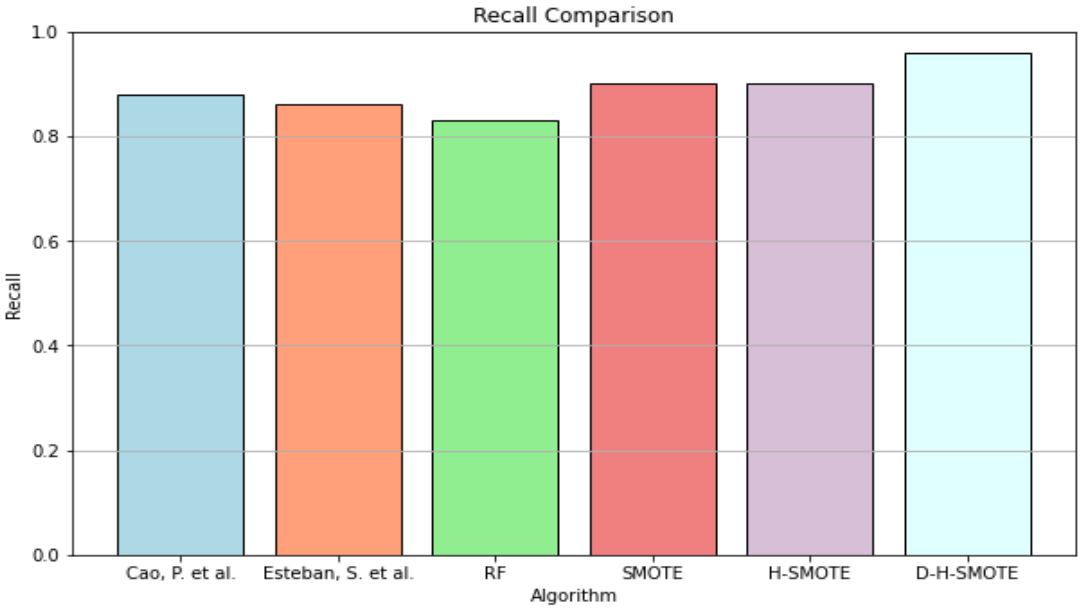

Figure 8: Recall values comparison chart

The figure 8 shows recall values comparison chart the x axis shows algorithms and the y axis shows recall values.

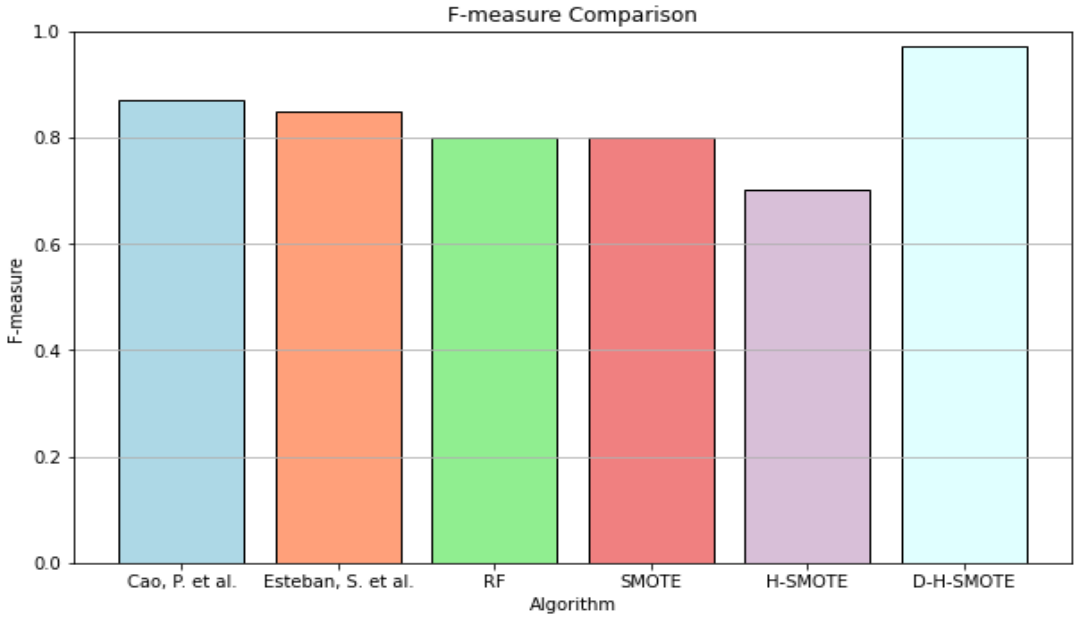

Figure 9: F-measure comparison chart

The figure 9 shows F-measure comparison chart the x axis shows algorithms and the y axis shows F-measure values.

5. CONCLUSION

Finally, our proposed hybrid methodology, which combines the Amalgam Multivariate Statistical Modeling Algorithm, Decision-Making Weighted Feature Selection, and the H-SMOTE Tree classification technique, represents a promising avenue for advancing Diabetic Type II prediction in the era of Big Data analytics. Our technique offers a systematic and successful strategy for dealing with the complexities associated with large diabetes datasets via rigorous application across three major phases. By addressing complexities and subtleties, the Amalgam Multivariate Statistical Modeling Algorithm effectively improves data quality, opening the way for more accurate analyses. Weighted Feature Selection in Decision-Making aids to model efficiency by selectively picking useful features. Artificial Neural Network training gives a strong foundation for future classification tasks. The D-H-SMOTE Tree classification strategy successfully addresses class imbalance, improving the model's generalization and classification accuracy. In contrast, the suggested technique, D-H-SMOTE, surpassed all others with 99% accuracy, keeping precision at 0.80 while dramatically boosting recall and F-measure to 0.90. These components work together to form a complete prediction model that has great potential for early diagnosis and intervention in the setting of Type II Diabetes. This research not only adds to our knowledge

of predictive analytics in healthcare, but it also highlights the potential of hybrid techniques for overcoming the issues provided by unbalanced datasets in illness prediction scenarios.

References

1. Ameena, R. R., & Ashadevi, B. (2020). Predictive analysis of diabetic women patients using R. Systems Simulation and Modeling for Cloud Computing and Big Data Applications, 99–113. doi:10.1016/b978-0-12-819779-0.00006-x
2. Campos, M. P., & Reis, M. S. (2020). Data Preprocessing for Multiblock Modelling – A Systematization with New Methods. Chemometrics and Intelligent Laboratory Systems, 103959. doi:10.1016/j.chemolab.2020.10395
3. Cao, P., Ren, F., Wan, C., Yang, J., & Zaiane, O. (2018). Efficient multi-kernel multi-instance learning using weakly supervised and imbalanced data for diabetic retinopathy diagnosis. Computerized Medical Imaging and Graphics. doi:10.1016/j.compmedimag.2018.08.008
4. Chatragadda, B., Kattula, S., & Guthikonda, G. (2018). Diabetes Data Prediction Using Spark and Analysis in Hue Over Big Data. 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT). https://doi.org/10.1109/rteict42901.2018.9012339
5. Ding, S., Li, Z., Liu, X., Huang, H., & Yang, S. (2019). Diabetic Complication Prediction Using a Similarity-Enhanced Latent Dirichlet Allocation Model. Information Sciences. doi:10.1016/j.ins.2019.05.037
6. Esteban, S., Rodríguez Tablado, M., Peper, F. E., Mahumud, Y. S., Ricci, R. I., Kopitowski, K. S., & Terrasa, S. A. (2017). Development and validation of various phenotyping algorithms for Diabetes Mellitus using data from electronic health records. Computer Methods and Programs in Biomedicine, 152, 53–70. doi:10.1016/j.cmpb.2017.09.009
7. Fiarni, C., Sipayung, E. M., & Maemunah, S. (2019). Analysis and Prediction of Diabetes Complication Disease using Data Mining Algorithm. Procedia Computer Science, 161, 449–457. doi:10.1016/j.procs.2019.11.144
8. García-Gil, D., Luengo, J., García, S., & Herrera, F. (2018). Enabling Smart Data: Noise filtering in Big Data classification. Information Sciences. doi:10.1016/j.ins.2018.12.002
9. Georga EI, Protopappas VC, Ardigo D, Marina M, Zavaroni I, Polyzos D, Fotiadis DI. Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression. IEEE J Biomed Health Inform. 2013 Jan;17(1):71-81. doi: 10.1109/TITB.2012.2219876. Epub 2012 Sep 19. PMID: 23008265.
10. Hariharakrishnan, J., Mohanavalli, S., Srividya, & Kumar, K. B. S. (2017). Survey of pre-processing techniques for mining big data. 2017 International Conference on Computer, Communication and Signal Processing (ICCCSP). doi:10.1109/icccsp.2017.7944072
11. Hassan, S., Dhali, M., Zaman, F., & Tanveer, M. (2021). Big data and predictive analytics in healthcare in Bangladesh: regulatory challenges. Heliyon, 7(6), e07179. doi:10.1016/j.heliyon.2021.e07179
12. Huang, F., Abbasi-Sureshjani, S., Zhang, J., Bekkers, E. J., Dashtbozorg, B., & ter Haar Romeny, B. M. (2019). Vascular biomarkers for diabetes and diabetic retinopathy screening. Computational Retinal Image Analysis, 319–352. doi:10.1016/b978-0-08-102816-2.00017-4
13. Islam, M. M., Yang, H.-C., Poly, T. N., Jian, W.-S., & Li, Y.-C. (Jack). (2020). Deep Learning Algorithms for Detection of Diabetic Retinopathy in Retinal Fundus Photographs:

A Systematic Review and Meta-Analysis. Computer Methods and Programs in Biomedicine, 105320. doi:10.1016/j.cmpb.2020.105320

14. Jayalskshmi, T., & Santhakumaran, A. (2010). Impact of Preprocessing for Diagnosis of Diabetes Mellitus Using Artificial Neural Networks. 2010 Second International Conference on Machine Learning and Computing. doi:10.1109/icmlc.2010.65
15. Kadhim, A. I., Cheah, Y.-N., & Ahamed, N. H. (2014). Text Document Preprocessing and Dimension Reduction Techniques for Text Document Clustering. 2014 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology. doi:10.1109/icaiet.2014.21
16. Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., & Chouvarda, I. (2017). Machine Learning and Data Mining Methods in Diabetes Research. Computational and Structural Biotechnology Journal, 15, 104–116. doi:10.1016/j.csbj.2016.12.005
17. Khanna, N. N., Jamthikar, A. D., Gupta, D., Nicolaides, A., Araki, T., Saba, L., … Suri, J. S. (2019). Performance evaluation of 10-year ultrasound image-based stroke/cardiovascular (CV) risk calculator by comparing against ten conventional CV risk calculators: A diabetic study. Computers in Biology and Medicine, 105, 125–143. doi:10.1016/j.compbiomed.2019.01.002
18. Nagarathna, R., Tyagi, R., Battu, P., Singh, A., Anand, A., & Ramarao Nagendra, H. (2020). Assessment of Risk of Diabetes by using Indian Diabetic Risk Score (IDRS) in Indian population. Diabetes Research and Clinical Practice, 108088. doi:10.1016/j.diabres.2020.108088
19. NirmalaDevi, M., Appavu, S., & Swathi, U. V. (2013). An amalgam KNN to predict diabetes mellitus. 2013 IEEE International Conference ON Emerging Trends in Computing, Communication and Nanotechnology (ICECCN). doi:10.1109/ice-ccn.2013.6528591
20. Prasad, S. T., Sangavi, S., Deepa, A., Sairabanu, F., & Ragasudha, R. (2017). Diabetic data analysis in big data with predictive method. 2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET). doi:10.1109/icammaet.2017.8186738
21. Wang, J., Lv, M., Li, Z., & Zeng, B. (2023). Multivariate selection-combination short-term wind speed forecasting system based on convolution-recurrent network and multi-objective chameleon swarm algorithm. Expert Systems with Applications, 214, 119129.
22. Tang, Chang, Xiao Zheng, Wei Zhang, Xinwang Liu, Xinzhong Zhu, and En Zhu. "Unsupervised feature selection via multiple graph fusion and feature weight learning." Science China Information Sciences 66, no. 5 (2023): 1-17.
23. Jaloli, M., & Cescon, M. (2023). Long-term prediction of blood glucose levels in type 1 diabetes using a cnn-lstm-based deep neural network. Journal of diabetes science and technology, 17(6), 1590-1601.