IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol. 10, Issue.6, June 2022 E-ISSN: 2347-2693

A Machine Learning Based Diabetes Prediction Using Stacking and Stacking With Hyperparameter Tuning

Sadhana Tiwari[1], Awadhesh Kumar[2*], Aasha Singh[3]

[1]P.G. Scholar, Computer Science & Engineering, KNIT, Sultanpur, India
[2]Associate Professor, Computer Science & Engineering, KNIT, Sultanpur, India
[3]Research Scholar, Maharishi University of Information Technology, Lucknow, India

**Corresponding Author: awadhesh@knit.ac.in*

DOI: https://doi.org/10.26438/ijcse/v10i6.1621 | Available online at: www.ijcseonline.org

Received: 19/May/2022, Accepted: 08/Jun/2022, Published: 30/Jun/2022

***Abstract*—** Due to the high blood sugar or blood glucose, the problem of diabetes will occur, and it's also referred to as a metabolic disorder. Long-term high blood glucose levels can result in several heart-related disorders, strokes, renal illness, vision difficulties, dental problems, nerve damage, and other problems. The latest recent information about diabetes worldwide may be found in the IDF Diabetes Atlas, ninth edition 2021.There are 537 million adults facing the problem of diabetes according to the measurement of 2021 year. And we are guessing that there will be total diabetes patients will number 643 million by 2030 and 783 million by 2045. To predict the diabetes, we generally use machine learning algorithms. Here we have executed various machine learning algorithms like K-Nearest Neighbor, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, Stacking and Stacking with Hyperparameter Tuning. Each model will have different accuracy in compared to other models. The most accurate result can be achieved by the stacking and stacking with hyperparameter tuning.

***Keywords*—** Machine Learning, Diabetes, Random Forest, Stacking, Hyperparameter Tuning, LogisticRegression.

I. INTRODUCTION

As we can see diabetes is a very common, complicated and serious problem nowadays. It may be due to the ineffective insulin of our body or when our body doesn't produce insulin. It's a chronic health condition which will turn food into energy. The food which we eat, firstly broke into sugar and then goes into our bloodstreams.
Diabetes can be categorized mainly in three different types:

1. Type 1 diabetes
2. Type 2 diabetes
3. Gestational

Diabetes. Percentage of type 1 patients is 5-10%, percentage of type 2 diabetes is 90-95%. Diabetes affects food to energy conversion in your body. Due to the diabetes, our body can't be able to produce enough insulin as well as can't be able to make use of insulin effectively.
Diabetes symptoms are –

1. Urinate
2. Feeling thirsty again & again
3. Losing weight
4. Feeling hungry even after having food
5. Vision starts to blur
6. Tingling hands
7. Feeling tired
8. Skin feels dry all time
9. Cause infections

Use of technology in medical field has helped to improve the worlds health system. Everyday new innovations are coming in the medical field. Machine learning is helping to improve the existing health system and make more accurate suggestions related to the health problems. In machine learning there are many different kinds of models are present that are best suited for particular types of problems. In this research paper, to improve the accuracy of diabetes prediction, I will use stacking classifier with hyper-parameters tuning. Every model has some advantages and disadvantages depending on the type of data is used to train that model. So, we used ensemble method i.e., stacking classifier, which is a combination of more than one model. By this we tried to reduce the disadvantages of using single model and improving overall accuracy of the system.

II. RELATED WORK

In paper [1], it has been achieved that highest accuracy is 77%. The model which has been used are Random Forest, Decision tree, SVM, KNN, Logistic Regression, Gradient Boosting and out of them random forest gives highest accuracy of 77%.

In paper [2], it has been achieved that hybrid approach gives better accuracy than single classifier, so it has been found that stacking and Adaboost gives better accuracy than single classifier.

In paper [3], it has been found that hybrid approach gives better accuracy than single classifier and fuzzy logics gives better results when the techniques are integrated with them.

In paper [4], using principal component Analysis method (which basically reduce the dimension and select the specific feature) Random Forest gives 83% accuracy and SVM gives 81.4% accuracy.

An approach developed by Dilip Kumar Choubey [5], was used like Genetic Algorithm as an Attribute Selection procedure and NBs for Classification process on PIDD and it had been obtained from the UCI machine learning library. The required dataset has been divided into training and test portions of 70% and 30%, respectively. Highest accuracy attained is 79 percent.

This research [6], carefully examined a number of data mining strategies, and it was concluded Nave Bayes, K-NN, Decision Tree, and Support Vector Machines were employed by researchers in the vast majority of instances working alone or using combining methods to improve the predictability accuracy.

The MV dataset, which has 1024 instances and 26 attributes overall, was used by Vinaytosh Mishra [7]. Some of the categorization models used are Multilayer Perceptron, BayesNet, JRip, C4.5, and Fuzzy Lattice Reasoning (FLR). The JRip model had the highest accuracy, coming in at 86%.

In this study [8], authors used a variety of methods used in machine learning, including SVM, KNN, decision trees, random forests, to forecast how well various categorization methods will perform. The conceptual prediction model that incorporates several machine learning classifiers is also proposed. As a result, it can be inferred by comparing the accuracy of various models of machine learning technique where Random Forest performs better than other Classification Techniques. With the right datasets, this study may be performed in the future on many models for other illnesses.

In paper [9], the authors wanted to determine a method for eliminating unnecessary variables from machine learning models in order to improve their accuracy. They used Binary Logistic Regression for classification and IBM SPSS for data analysis. While sex, BMI, and HBA1C were shown to be insignificant within the 95 percent confidence interval, age, smoking, parental diabetes mellitus, hypertension, and waist circumference were found to be significant.

Based on the sensitivity of the dataset and the issue statement, the R. Manimaran [10] attempted to choose the appropriate qualities from a big database. The selection of appropriate qualities for the problem necessitates a comprehensive examination of the attributes and the exclusion of extraneous features. NB obtained the highest accuracy of 82.30 percent using the recommended strategy.

III. METHODOLOGY

Our mission is to find the ideal model to predict diabetes in early stage with more accurate accuracy. We tested with different types of classification and ensemble algorithms to predict diabetes. In the following, we have shown the stages of our work.

A. Dataset: The dataset utilised is Pima-Indian-Diabetes. It was introduced by the National Institute of Diabetes and Digestive and Kidney Diseases. Objective of this dataset is to predict if a patient has diabetes or not, these predictions are done based on measures provided in the dataset. The Pima-Indian-Diabetes dataset includes some medical parameters such as the number of pregnancies, blood glucose levels, body pressure, and skin thickness, as well as insulin, BMI, the patient's age, the function of their diabetes pedigree, and one dependent variable (outcome) parameter, the value of which is binary. The dataset is mainly for female gender, and we have described the dataset as following-

There are 9 columns and 768 rows observation having 268 positives patients of diabetes (dependent variable value 1) and 500 negatives patients of diabetes (dependent variable value 0).

1. ***Pregnancies:*** How many times pregnant
2. ***Glucose:*** Result of Oral Glucose Tolerance
3. ***Blood Pressure:*** Diastolic blood pressure measurements are third
4. ***Skin Thickness:*** Triceps skin fold thickness (mm)
5. ***Insulin:*** Serum insulin of 2 hours (mu U/ml)
6. ***Body Mass Index (BMI):*** Mass index of body
7. ***Diabetes Pedigree Function:*** Based on family history, this function determines the chance of diabetes.
8. ***Age:*** In years
9. ***Outcome:*** 1 indicates that a person has diabetes and 0 indicate that a person has not diabetes.

B. Data Preprocessing: In this step data is transformed from raw data into a more meaningful format that will be suitable for machine learning model is called as Data preprocessing. This step is considered very important for data mining because working on raw data is not recommended as it may give accurate result. Before applying the machine learning algorithm. For Pima Indian diabetes dataset, we have to perform preprocessing in following steps:

1) Replacing zero value: We can see that Glucose, Insulin, Skin Thickness, BMI and Blood Pressure which have value as 0. That's not possible. We can either remove such data or simply replace it with their respective mean values.

2) Remove and detect the outliers: Outlier are data-items that does not represent relevant information related to actual data in dataset. The reason behind this is measurement or execution errors. Outlier mining is the analysis for outlier detection.

3) The Training and Test sets are separated from the Dataset: As per the need the dataset available for the experiment must be divided into training and testing portions throughout the machine learning data preparation procedure. This is a crucial stage in the preparation of data.

We can improve the performance of our machine learning model by doing this.

It will be challenging for our machine learning model to identify the relationships between the models if, for example, it was trained using one dataset and then tested on a completely different dataset.
Our machine learning model training accuracy will very high if we train our model very well but if we provide the new dataset, it degrades the model's overall performance.
Performance of machine learning models should be well for training data set and with the test dataset.

C. Apply Machine Learning Technique: Preprocessing of data is completed, now it's time to apply different machine learning algorithms on this pre-processed dataset. The techniques are:

1. KNN Algorithm
It's a supervised learning algorithm as well. It uses already available data to decide the class of new data by using resemblance between new and existing data. When any new data comes its similarity is checked with all currently available classes, out of all present classes with which new data represents most similarity will be assigned to it. This approach may be used to address regression and classification issues, but mostly it is used for solving classifications problems. As this algorithm does not learn from new data it is also called lazy learner algorithm.

Need of KNN Algorithm: We can understand it with the given diagrams which is from javatpoint. We have a point, and we don't know the category of this point, so we will pass it from KNN model, and it will automatically predict its category.

In KNN, similar types of data are grouped together and the points inside a group has least distance between other points inside the same group. As in Figure 1, two categories (A and B) of points are formed. When any new points come its Euclidean distance (shown in Figure 2) is calculated from the centroid of different groups. Group with which point has smallest Euclidean distance will be the group of new point.

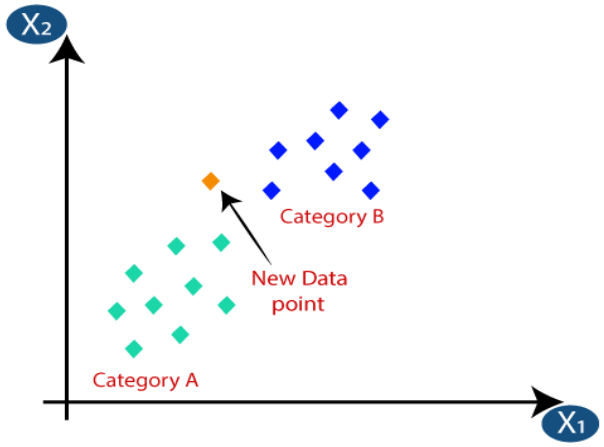

Figure 1. Classification of new point in KNN algorithm

Working of KNN algorithm: There are some steps included in the working of this algorithm:
Step-1: First take a neighboring number K Step-2: Now for K numbers calculate Euclidean distance
Step-3: Based on Euclidean distance found in step two, K nearest neighbors are taken
Step-4: Group which has maximum number of neighbors, new data will be assigned to it
Step-5: Finally, our model is ready. Formula for finding Euclidean Distance: Let's take A (x1, y1) and B (x2, y2) points in space we can see it with the given diagram taken from javatpoint.

Formula for finding Euclidean Distance:

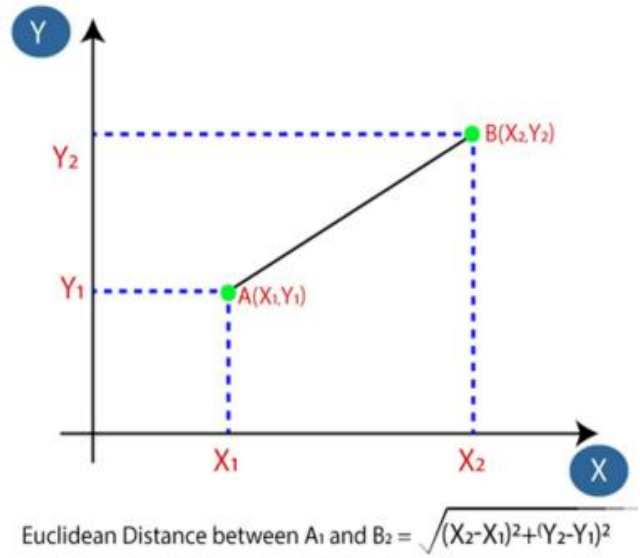

Figure 2. Calculating euclidean distance between two points

2. Random Forest Algorithm
Multiple decision trees are built using this machine learning approach. It is extension of the decision tree where only one tree is created. This may be used to tackle classification and regression difficulties. As it has multiple decision tree so final decision is not just taken by one decision tree, it is taken on the outcome of the majority of decision trees. It we have large number of trees in the forest, then it will have higher accuracy and also it will not create the problem of overfitting.

Working: The working process can be understood by the following steps:
- From entire training data randomly select K data points
- Using randomly selected k data points create a decision tree
- For decision trees select number N
- Step 1 and 2 repeated again
- Now using decision tree prediction on test data is found and the category which have most votes will contain the new data point. The working of random forest algorithm can be understood from given flow diagram which is taken from javatpoint.

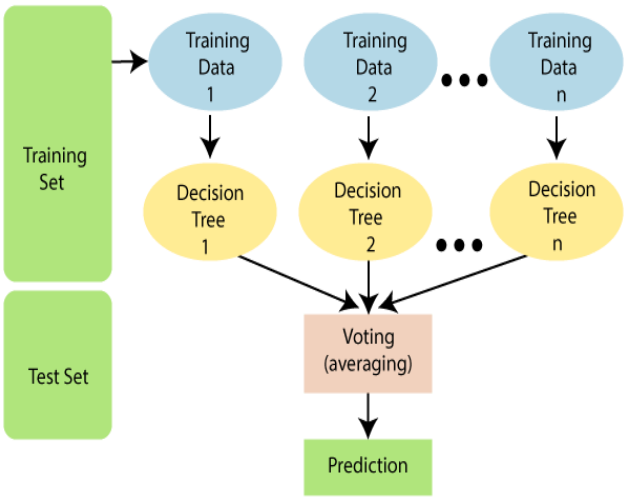

Figure 3. Random Forest Classifier architecture

3.Decision Tree

Decision tree algorithm requires labelled data for its training, and we know any algorithm that uses labelled data falls under supervised machine learning algorithms. It may be applied to both classification and regression. Decision and leaf are two sorts of nodes present in it. Decision nodes are also called internal nodes. They represent a condition, based on the result of the condition next branch is decided. Leaf nodes represents final result to which a dataset can belongs to. It is a graphical representation to get all the possible solution of any decision-based problem. For building the tree, we use CART algorithm.

Working: The working of decision tree algorithm will follow below steps:

- Start tree from root node S
- Using ASM find required attribute in dataset
- Subsets will be created by dividing the root node, these subsets will contain the possible values for the best attributes
- Using best attributes make decision tree having all the attributes
- Now using same process generate new decision trees using dataset divided in step three. Repeat this process till we reach leaf node.

4. Logistic Regression

It comes under supervised machine learning algorithms because labelled data is required for training. It is used for binary classifications. It is different from the Linear regression model on prediction of type of data. In this set of independent variables are used to predict the categorically dependent variable. It is used to find the probability of dependent variable. It establishes a link between dependent and independent variables by calculating probabilities. The probability will be converted to binary value by logistic function. It is widely used in many different fields such as technology, medical, trading and business and many more. Representation of Logistic Regression in mathematical equation: Given below equation is for logistic regression.

Types of Logistic Regression model:

- ***Binomial:*** It can have 2 possible dependent variables.
- ***Multinomial***: It can have 3 or more unordered dependent variable.
- ***Ordinal:*** It can have 3 or more ordered dependent variable.

Steps of logistic regression:

- Pre-processing step
- Fitting Logistic Regression to training data
- Prediction of the test data
- Verify that the test data is accurate.
- Draw graphs for better visualization of result

5. Support Vector Machine

The Support Vector Machine (SVM) is an important and most fundamental classification methods. It is a supervised machine learning algorithm, meaning that it needs labelled training data. Each item to be classified is represented as a point in n-dimensional space, with its coordinates referred to as features. By creating a hyperplane, SVM carries out categorization. This, hyperplane can be a line or a plane, that depends on the type of space it is drawn like 2-dimensional or 3-dimensional. The points belonging to the same kind are positioned on one side of the hyperplane of this plane, while the points belonging to the other category are located on a different side of the aircraft. In a space multiple such planes can exists, SVM tries to find the plane that best separates the two categories. It tries to maximize the distance between the two category points. Points that fall exactly on the margin are called support vectors.

Types of SVM:

- Linear SVM
- Non-Linear SVM

6. Stacking

Stacking is an ensemble technique used in machine learning. It is a hybrid machine learning model. In this more than one machine learning models are combined in two different layers. First layer models are called baseline models. It can have n-number of baseline models. Number and types of machine learning models in baseline are depends on our requirement and problem for which this model is developed Second layer of this has meta classifier, it receives the prediction from first layer models as input. Stacking improves model predictions by combining the results of many models and running them through a meta-learner, another machine learning model. The advantage of stacking is that it can provide predictions that are more accurate than those made by using only one model by combining the talents of a number of high-performing models to tackle a variety of problems. Stacking classifier uses Random Forest Classifier as its meta classifier i.e., second layer model and Random Forest Classifier, Logistic Regression as its base classifiers.

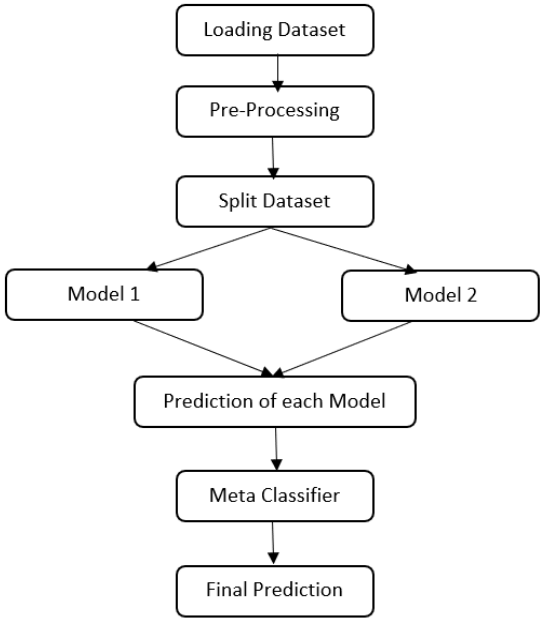

Figure 4. Stacking classifier architecture

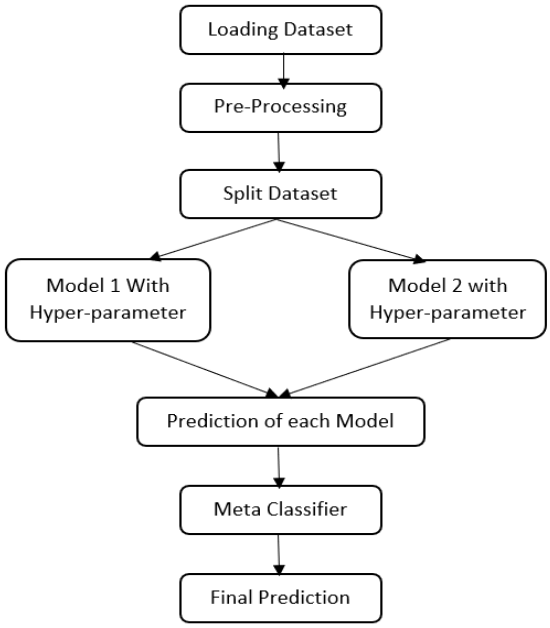

Figure 5. Stacking with Hyper-Parameter Tuning classifier architecture

8. Stacking with Hyper-Parameter Tuning

Parameters in machine learning models are of two types.

- ***Model Parameter:*** Parameter of this type are created and updated by learning process.
- ***Hyper Parameter:*** This type of parameter should be set before the actual learning process of model starts, because the architecture of machine learning is defined by them. Through data learning directly estimation of hyperparameter is not done.

To develop a machine learning model that can work on different types of problems, it's hyper-parameters tuning is required. There is a direct correlation between hyper-parameters and the performance of machine learning models.

Hyper-parameters can be categorical, discrete and continuous, then they will be different for different machine learning models. So, the process of tuning hyper-parameters is also different. In our work we used hyper-parameters tuning with stacking to improve the accuracy of over-all model. First, we defined Logistic Regression hyper-parameters model and Random Forest Classifier hyper-parameters. Hyper-parameters used in training Logistic Regression model are C, Penalty and Solver. Hyper-parameters used for Random Forest Classifier are _Estimators, Max_Depth, Max_features, Bootstrap and Criterion. First, we train Logistic Regression and Random Forest Classifier model on dataset using these hyper-parameters. After training predictions are done on these two models. Then result of prediction of these models is used to train the final model of stacking. Final model gives an accuracy of 88 percent.

Table 1. Accuracy Comparison

Sr. No.	Model	Accuracy
1	Decision Tree	69.8225
2	KNN	75.1479
3	SVC	76.3314
4	Logistic Regression	77.5148
5	Random Forest	78.1065
6	Stacking	82.3529
7	Stacking with Hyper-parameter tuning	88.2352

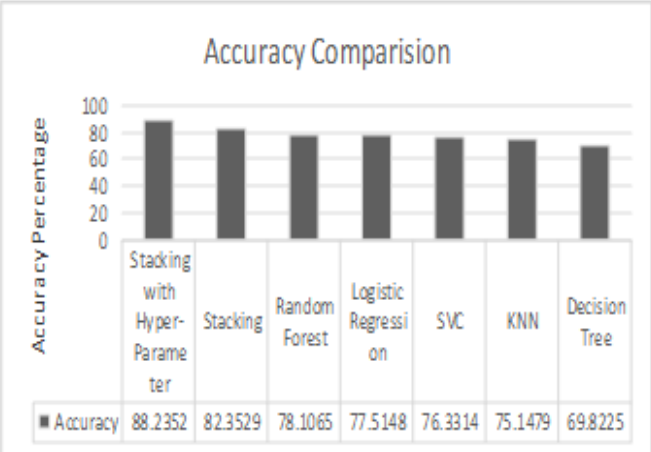

Figure 6. Accuracy Comparison Graph

IV. RESULTS AND DISCUSSION

This study's primary goal is to compare some existing research papers in the same field with this one and try to discuss improvement areas that can be done. In this research I have tried to use different machine learning

models on same data set. I have tried some algorithms of different types like K-Nearest neighbors, LR, SVM, decision tree and random forest classifier. The goal of our work was to increase the accuracy of diabetes prediction model. We have achieved it by getting accuracy of 88 percent using stacking classifier with hyperparameter tuning. Without stacking classifier, we get accuracy of 78 percent using Random Forest classifier and when we have taken stacking classifier without hyper-parameter tuning, the accuracy is 82 percent. We have implemented many other classifiers, but all are having less accuracy in comparison to these three methods. Comparison of accuracy obtained using different classifiers is shown in Table 1.

V. CONCLUSION AND FUTURE SCOPE

The aim of our research work is to implement diabetes prediction in early stage by using machine learning models with increased accuracy of model and it has been achieved successfully. We found that hybrid approaches with hyper-parameter tuning yield better results than hybrid classifier without hyper-parameter tuning and single classifiers. Using stacking in some cases, it is very helpful to use different kinds of prediction models to get higher accuracy and hyper-parameter tuning is the process of finding the right set of hyper-parameter values for achieving the maximum performance on the dataset which maximize the accuracy of our model.

The dataset which is used in our research has not some important feature like polycystic ovary syndrome, weight, gestational diabetes, family history, cholesterol level etc. so dataset with more number of feature increases the accuracy of our model and also maximize the performance of the model and in next years to find people who do not have diabetes can be effected with diabetes. With continued technological development, it's possible that we'll soon be able to predict a person's chance of having diabetes.

REFERENCES

[1] M. Soni, Dr. S. Varma, "*Diabetes Prediction Using Machine Learning Techniques*". International Journal of Engineering Research & Technology, Vol.**9**, Issue. **9**, pp. **921-925**, **2020**.

[2] K. Patil, S.D. Sawarkar, "*Designing a Model to Detect Diabetes Using Machine Learning*". International Journal of Engineering Research & Technology, Vol. **8**, Issue. **11**, pp. **512-515**, **2019**.

[3] S. Gujral, "*Early Diabetes Detection Using Machine Learning*". International Journal for Innovative Research in Science & Technology, Vol. **3**, Issue. **10**, pp. **57-62** , **2017**.

[4] S. Sivaranjani, S. Ananya, J. Aravinth, R. Karthika, *"Diabetes Prediction using Machine Learning Algorithms with Feature Selection and Dimensionality Reduction*", International Conference on Advanced Computing and Communication Systems, **India**, pp. **141-146**, **2021.**

[5] D. K. Choubey, S. Paul, "*A Hybrid Intelligent System for Diabetes Disease Diagnosis*", International Journal of Intelligent Systems and Applications, Vol. **08,** Issue. **01**, pp. **49-59** , **2016**.

[6] M.Shuja, S. Mittal, M. Zaman, "*Diabetes Mellitus and Data Mining Techniques: A survey",* International Journal of Computer Science and Engineering, Vol. **7**, Issue. **01**, pp. **856-862**, **2019**.

[7] V. Mishra, C. Samuel, S.K. Sharma, "*Use of Machine Learning to Predict the Onset of Diabetes*", International Journal of Recent advances in Mechanical Engineering, Vol. **4**, Issue **2**, pp. **9-14** , **2015**.

[8] N.A. Farooqui, Ritika, A. Tyagi, "*Prediction Model for Diabetes Mellitus Using Machine Learning Techniques",* International Journal of Computer Science and Engineering, Vol. **6**, Issue. **3**, pp. **292-296, 2018.**

[9] A. Vaghela, G. S. Pandit, "*A Fusion Approach for Prediction of Diabetes sing machine learning Techniques*", International Research Journal of Engineering and Technology, Vol. **8**, Issue **01**, pp. **808-813** , **2021**.

[10] R. Manimaran, M. Vanitha, "*Prediction of Diabetes Disease Using Classification Data Mining Techniques*", International Journal of Engineering and Technology, Vol **9**, Issue **5**, pp. **3610-3614** , **2017**.

AUTHORS PROFILES

Sadhana Tiwari has completed his B.Tech. in Computer Science from Kamla Nehru Institute of Technology, Sultanpur, UP, India in 2020 and currently she is pursing her M.Tech. in Computer Science from Kamla Nehru Institute of Technology, Sultanpur. Her area of interest is machine learning and data mining.

Dr.Awadesh Kumar has completed his B.E. degree from G.B. Pant Engineering College, Puri(Garwal) in 1999 and M.Tech. in Computer Science from A.K.T.U. Lucknow, U.P. in 2006 and Ph.D. from M.N.N.I.T. Allahabad, U.P., India in 2017. He is presently working as Associate Professor in the departments of Computer Science & Engineering in KNIT Sultanpur, U.P., India since 2000. His teaching and research interests include Computer Networks, Mobile Ad-Hoc Networks, Wireless Sensor Networks.

Aasha Singh, has completed her B.Sc. from University of Lucknow in 2004 and M.C.A. from KNIT Sultanpur in 2011. She is pursuing her Ph.D. from M.U.I.T. Lucknow. Her research areas are Machine Learning, Software Engineering, Data Mining.

