Journal of Research Administration
Society of Research Administrators International

DATA DRIVEN MACHINE LEARNING ENSEMBLE APPROACH FOR DIABETES RISK PREDICTION AT EARLY STAGES

S. Sutha[1*], N. Gnanambigai[2], P. Dinadayalan[3]

[1*]Research Scholar, Bharathiar University, Coimbatore - 641046, Tamil Nadu, India,

[2]Department of Computer Science, Indira Gandhi College of Arts and Science, Puducherry - 605009,

[3]Department of Computer Sci, Kanchi Mamunivar Centre for Postgraduate Studies, Puducherry - 605008.

***Corresponding Author:** S. Sutha

*Research Scholar, Bharathiar University, Coimbatore - 641046, Tamil Nadu, India,

Abstract

Diabetes mellitus is characterized as severe illness with disruption in glucose, lipid, and protein metabolism. Hyperglycemia, or high blood sugar levels, is the most common symptom of all types of diabetes. Diabetes has become much more common as a result of contemporary living. As a result, early illness detection is critical. ML is grown among health care professionals and clinicians as it has tremendous potentials for generating tool for disease management, risk prediction, therapy, prognosis. This article, offers an ensemble strategy for diabetes prediction at the early stages that combines AdaBoost and CatBoost. The suggested technique is called Sel stack AdaCat, and it attempts to produce high-efficiency risk prediction tools for type2 diabetes incidence. Characteristics analysis are performed to assess significance and investigate relationships with diabetes. These include the most common diabetic symptoms, which normally grow gradually, and can serve as tools to train and assess various ML algorithms. Different ML algorithms are evaluated and compared in regards to Precision, Recall, F-Measure, Reliability, and AUC utilizing 10-fold cross-validation and information splitting.

Keywords- diabetics, machine learning, preprocessing, ensemble classifier, meta-heuristic optimization.

Introduction

Diabetes mellitus is a group of metabolic diseases characterized by hyperglycemia due to abnormalities in insulin synthesis, activity, or even both [1]. Type2 diabetes, also called as insulin resistance (insulin deficit), develops whenever cell react badly to insulin, resulting in decreased glucose absorption [2]. The American Diabetes Association's diagnosis requirements are as follows: glycated hemoglobin (HbA1c) level larger than or equal to 6.5 percentage; fasting blood glucose level greater than or equal to 126mg/dL; blood glucose level greater/equal 200mg/dL 2 hours following oral glycemic control with 75g of glucose[3]. Diabetes is a global public health issue. In 2k19, the World Diabetes Federation reported that 463 million people worldwide had diabetes, with a 51percentage increase anticipated by 2k45. It is believed that one undiagnosed individual exists for every identified diabetic person [4]. There are 3 forms of diabetes and a prediabetes state.

- Diabetes Type1. It occurs when pancreatic cells make inadequate insulin and inject it into the body through external sources to maintain body glucose levels[5]. This kind of diabetes is more common among younger people.
- Diabetes Type 2. When the body's metabolic process is unable to thoroughly digest meals, sugar levels in the blood rise. This kind of diabetes can also be hereditary. This kind of diabetes is more common in adults between the ages of 45 and 60.
- Gestational Diabetes . This kind of diabetes is caused by hormonal changes and an increase in insulin production during pregnancy.
- Prediabetes. This disease, also called as borderline diabetes, occurs when blood sugar levels are high but not high enough to be identified as diabetes.

Machine learning is a notion that learns from examples and previous information and makes predictions for future information based on the analysis of prior information. Programmers are not required here since logic is developed on taught information and evaluated on test information[6]. It is a subfield of AI in which predictions are produced based on prior experience. It falls into one of 2 categories. Learning that is supervised. A trained algorithm guides the learning process. A new algorithm is trained utilizing the supplied input trained information set or algorithm, and predictions are generated once the new algorithm has been trained [7]. Learning without supervision. Observation is the primary system of learning. The programmer attempts to detect certain linkages [8]. Our machine learning system's novelty and contributions are as follows:

- Optimizing the PIMA Indians diabetes informationset by rejecting outliers and imputing missing values.
- Feature extraction utilizing the hybrid Bumblebees and Flower Pollination Optimization algorithm (Hy BFPO).
- Utilizing the Sel-Stacking algorithm fusion approach, the prediction algorithm, which is a mix of AdaBoost and CatBoost, was built to predict the risk of diabetes.

Section 2 summarises some prior research efforts, Section 3 shows the suggested approach and procedures, Section 4 demonstrates experimental findings as well as discussions, and Section 5 finishes with a conclusion and future research.

Related works

The toolbox has lately received a slew of new ML algorithms. These approaches anticipate new occurrences according to trends uncovered in training data from previous examples.

[9] defines super learner as a cross-validation-based technique for improving predictions by combining predictions from multiple algorithms utilizing ML. After a case study, proposed super learner algorithm was constructed utilizing 4 systems (regression, reptree, randomized jungle, as well as support vector) and macro learners (support vector machines). The development illustrates adaptability of proposed algorithm.

[10] employed ResNet50 with VGG16 DL algorithms for feature extraction. To choose unique traits for categorization, utilizing Quantum Fruit Fly Algorithm (QFFA) technique employs Archimedes spirals for enhancing algorithm exploit. Archimedes spiral provides spiraling searches in top Fruit Fly system solution, facilitating in the avoidance of global optimal traps and enhancing exploitation.

[11] employ CNN—ResNet 50 structure for extracting features and suggested RBFNN approach for segmentation based upon automatic encoder learning. For feature extraction, CNN—ResNet 50 design is employed, and for classification, recommended RBFNN technique based on auto encoder learning mechanism is applied.

In [12], optimum feature selection is achieved by mergin DL algorithms such as CNN utilizing MFO and CSA, resulting in MF-CSA. After categorising levels, enhanced RNNforecasts their range based on proposed MF-CSA.

[13] offers ML system for diabetic forecasting and diagnosis relying on PIMA Indian dataset and Medical City Hospital Diabetes Laboratory (LMCH). We hypothesise that employing feature extraction as well as incomplete data imputation systemologies would enhance diabetes prognosis and diagnostic classification algorithm effectiveness.

[14] offers a pipeline for predicting diabetes patients relying on DL technologies. It entails augmentation of information with a variational-auto-encoder (VAE), augmenting of features with such sparse autoencoder (SAE), as well as categorization with CNN .

[15] compares ML based forecasts (e.g., Glmnet, RF, XGBoost, LightGBM) against commonly utilized regress techniques to forecast undetected T2DM. Since higher stability of factor s examined over time aids algorithm understanding, medical systemologies must consider comprehensibility and system calibrating.

[16] created and compared deep learning techniques based on RNN LSTM and RNN GRU utilizing randomized forests with multi - layer perceptron neural classical frameworks. [17] presents a fusion machine learning strategy that reports an increase in accuracy to detect diabetes and predict start of critical stages in diabetic patients.

Thus far, all approaches reported for diabetes diagnosis have concentrated on feature selection technique along with certain machine learning algorithms including randomized forests, naive-Bayes, SVM, as well as decision-trees, only with characteristics chosen for prediction purposes. We faced following issues when studying all of these papers: (1) The lack of a bigger information was a critical concern in prediction since publicly accessible information only includes 9 characteristics, 1 of which is a class property. Resources and time are being spent on qualities that are unlikely to be picked for prediction purposes. (2) majority of writers eliminated missing data from standard information, that could have an effect on the conclusions as size of information decreases.

System algorithm

Figure 1 depicts the whole workflow of this paper, which mainly combines and explores a pre-processing approach based on Missing value imputation (MVI). The pre-processed information is subjected to feature extraction utilizing a bumble bee hybridization and flower pollination optimisation system. Finally, the collected characteristics are fed into an ensemble classifier to determine the kind of diabetes.

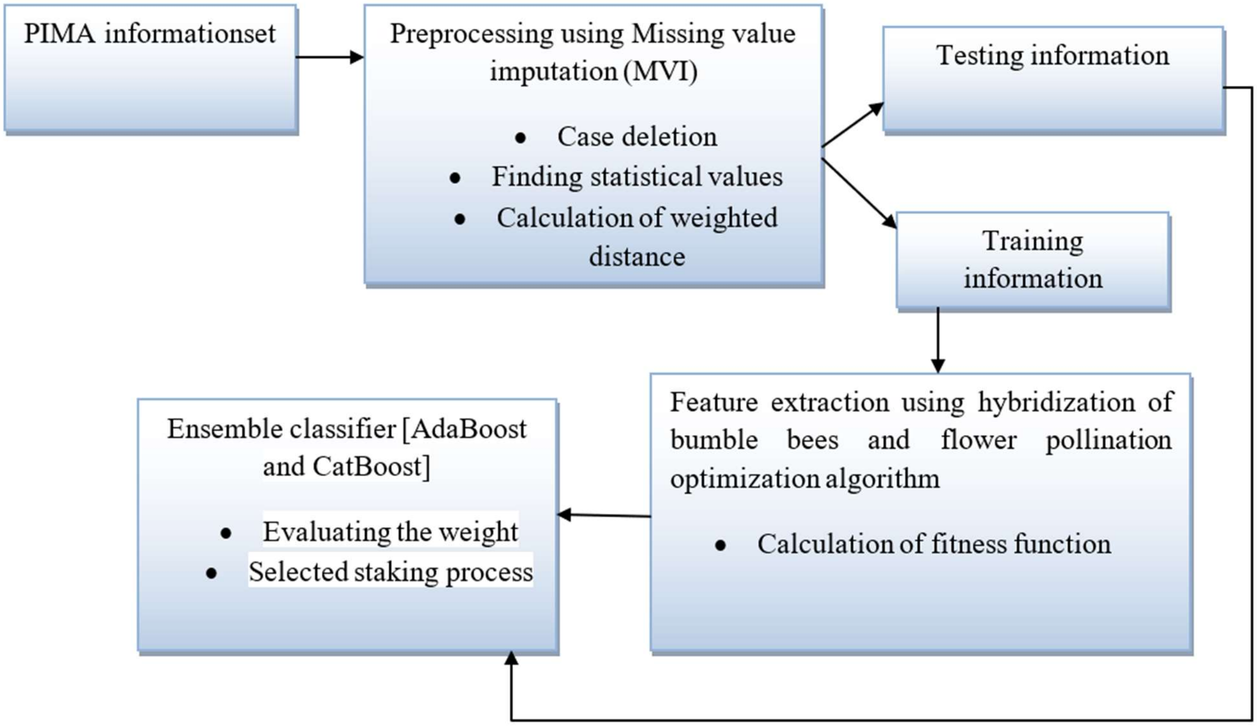
PIMA informationset
Preprocessing using Missing value imputation (MVI)
• Case deletion
• Finding statistical values
• Calculation of weighted distance
Testing information
Training information
Feature extraction using hybridization of bumble bees and flower pollination optimization algorithm
• Calculation of fitness function
Ensemble classifier [AdaBoost and CatBoost]
• Evaluating the weight
• Selected staking process

Figure-1 Block diagram of the proposed workflow

Dataset description

PIMA Indian Diabetes information was used in this investigation. The major purpose of information is to ascertain the patient's diabetes state. The information contains 1 outcome factor and a no., of medical prediction factor s. Age, no., of births, Obesity, hypertension, insulin, skin-thickness, glucose, and diabetes pedigree function are all diabetes prediction factor s. All participants in PIMA are females over the age of 21. Several limitations governed selection of these cases from a wider information base. We compare information pre-processing, extraction of features, and prediction systems in our research. The study's purpose is to minimize errors in early diabetic patients diagnosis. The following are some drawbacks: 1. A considerable no., of missing values leads to inaccurate estimates. 2. Unbalanced information has an impact on the algorithm's performance.

Preprocessing of information

Preprocessing uses missing value imputation (MVI). A information is the sole source of information for a trainable automatic classification decision-making mechanism. The practical information, on the other hand, frequently contains an extraordinary proportion of missing values, which are generally represented as NaNs, null, blanks, undefined, or similar placeholders. To construct a general, robust, and successful classification algorithm, missing values in a information must be deleted or imputed. In contrast to the case deletion technique, several statistical and machine learning approaches are widely used to deal with information missing in an incomplete information. As a result, for MVI purposes, this article employs median-based statistical imputation approaches.

STD is a typical metric for measuring variance in statistic. A small standard deviation number suggests that information points are often near to standard, whereas a large one implies a broad range of values. (STD)S is a common variation measure. A big STD value implies that information points vary widely, while a low STD value shows that education assists are near to typical. Outliers may be addressed two ways: (1) When STD = 1, compute average distance between missing value and class centre. If distance exceeds threshold, missing value is regarded outlier data and replaced with median value. (2) Missing numbers are anomalies if STD is greater than 1. Weight between missing values is closest neighbours for completing information to calculate average weight distance.

Step1: Incomplete information ($D_{i\_incomplete}$) for Class I is made up of a missed information sample (Num).

Step2: This method has 2 cases: (1) if STD = 1 from Step's first estimate. If gap is smaller than T 1, information j is imputed; otherwise, every outlier datum is fed into the Class 1 median. Outlier in case (2) was informed when STD > 1. Equation (3) calculates the average weight distance by weighting attribute data and its closest neighbors in full data.

$$.W_i = average\ [\frac{1}{dist(y_{1,}x_1\ )} + \frac{1}{dist(y_{i,}x_2\ )} + \cdots + \frac{1}{dist(y_{i,}x_j\ )}$$

here W_i is weighted gap of ith outlier information, y_i is W_i instance of outlier information, and x_1 is First complete information. From “Missing value imputation” section, function $dist(y_{i,}x_j\ )$ compute gap in-between $y_{i,}$ and x_j .

Feature extraction utilizing hybrid Bumblebees and Flower Pollination Optimization algorithm (Hy_BFPO)

Considering set of n missing value imputation information $S\ =\ \{M1, \dots, Mn\}$, and information with set of d features $\{F1, \dots, Fd\}$. For every feature Fi at wave t (Fi, t) in information, Data-Driven approach creates a subset of original information composed of all instances with called values for Fi, t (removing instances where Fi, t’s value is missing). This subset is called as the called information subset for $Fi, t.$ The average estimation error rate of every technique from S is then assessed in a 5-fold cross-validation done in that called information sample. In other words, called information subset for the current feature Fi,t is randomly partitioned into 5 folds of about equal size, and every imputation value is run 5 times, every time utilizing a different fold as a held-out "validation" subset and the other four folds as the "estimation" subset. For extracting the Bumblebee feature hybridization and Flower Pollination Optimization system (Hy BFPO). The empress, laborers, and drones (males) are picked in hive by utilizing 3 types of bumble bees. At 1st, no., of bees are chosen at random. Every bee (every bee represent a member in the population) indicates a potential solution to the problem. Consider total no., of factor s to be n. Vectors of size n are used to depict the bees. The empress picks the drones utilized for mating in algorithm by assuming that fittest males leave more pheromone in their flight patterns, and so the empress selects most promising paths. Initially, a crossover operator no., Cr_1 is chosen, which determines proportion of measurements chosen from the drones and empress. The value of Cr_1 is compared to the output of a random no., generator, $rand_j(0,1)$. If random no., is less or equal to Cr_1, corresponding value is inherited from empress, otherwise selected randomly, from solutions of 1 of drones' genotypes that have been stored in spermatheca. Thus, if solution of brood i is denoted by $b_{ij}(t)$ [t is iteration no., and j is dimension of problem $(j = 1,2, \ldots n)$], solution of empress is denoted by $q_j(t)$ and solution of drone k is denoted by $d_{kj}(t)$, then:

$$b_{ij}(t) = \begin{cases} q_j(t), & if\ rand_j(0,1) \leq Cr_1 \\ d_{kj}(t) & otherwise \end{cases}$$

The fittest broods are chosen as new empresses, while remainder are laborer. The no., of new empresses is chosen to be equal to maximum no., of empresses. The new empresses are initially fed by old empress (or empresses), and then by laborers and old empress (or empresses). We apply this system in order to enhance solution of every new empress. This is accomplished through a local search phase in which every new empress chooses which of the laborers and the old empress (or empresses) will feed her utilizing the following eq.,:

$$nq_{ij} = nq_{ij} + \left(b_{max} - \frac{(b_{max} - b_{min}) \times l_{si}}{l_{si_{max}}}\right) \times (nq_{ij} - q_j) + \frac{1}{M} \times \sum_{k=1}^{M} (b_{min} - \frac{(b_{min} - b_{max}) \times l_{si}}{l_{si_{max}}}) \times (nq_{ij} - W_{kj})$$

Here, nq_{ij} is solution of new empress i, q_j is solution of old empress (or empresses), W_{kj} is solution of worker, M is no., of laborers that every empress selects for feeding her and it is different for every empress, b_{max}; b_{min} are 2 measurements with values in interval (0,1), that control if new empress is fed from the old empress (or empresses), from the laborers or from both of them, l_{si} is current local search iteration and $l_{si_{max}}$ is maximum no., of local search iterations. The drones then depart hive in search of new empresses to breed with. The drones leave hive in a swarm to locate optimum sites to wait for new empresses to discover them via their programmed flight pathways. The following eq., is used to determine migration of drones away from hive:

$$d_{ij} = d_{ij} + \propto_1 \times (d_{kj} - d_{lj})$$

here d_{ij} ; $d_{kj}$and d_{lj} are the solutions of the drones i; k; l and $\propto_1$ is measurement that defines how much drone I is impacted by other 2 drones, k and l. The bumblebee approach employs both global and local pollination. If the pollination operations include local pollination, pollen is transferred to a nearby neighbor, then algorithm may be constructed as follows utilizing Rules 2 and 3:

Rule2: Self-pollination on neighboring flowers is considered a local pollination technique.

Rule3: Floral constancy is regarded as the reproduction rate, (i.e.,) direct proportion to the similarity of 2 involved flowers.

$$SX_i^{+1} = SX_i^t + \rho(SX_j^t - SX_k^t)$$

here SX_j and SX_k are pollen randomly selected from different flowers in the same plant, where j and k$\epsilon\{1,2,\ldots NP\}$ and ρ is a D-dimensional random vector in $[0,1]^D$. Furthermore, according to Rule4, global pollination and local pollination are executed based on a switch probability, implying that 2 pollination activities occur at random and are decided by a probability ρ. (i.e.,) if a random no., rand in the range [0, 1] is less than ρ, global pollination is performed; or vice-versa. STD is a typical metric for measuring variance in statistic. A small standard deviation number suggests that information points are often near to standard, whereas a large one implies a broad range of values. (STD)S is a common variation measure. A big STD value implies that information points vary widely, while a low STD value shows that education assists are near to typical. Outliers may be addressed two ways: (1) When STD = 1, compute average distance between missing value and class centre. If distance exceeds threshold, missing value is regarded outlier data and replaced with median value. (2) If STD is more than 1, the missing no., is considered an anomaly. Weight inbetween missing value are nearest neighbours for completing information is used to determine the average weight distance. In order to determine the fitness function of the new empresses in every iteration:

$$sig\left(nq_{ij}\right) = \frac{1}{1 + \exp\left(-nq_{ij}\right)}$$

and, , activated features are calculated by:

$$y_{ij} = \begin{cases} 1, & if\ rand_j\ (0,1) < sig(nq_{ij}) \\ 0, & ifrand_j\ (0,1) \geq sig(nq_{ij}) \end{cases}$$

Here y_{ij} is modified approach. The new empress chooses drones for mating utilizing previously outlined approach. The best fertilized empresses survive the following generation, whereas all other individuals of population perish.

Prediction process utilizing ensemble system

Following extracting features AdaBoost and CatBoost combo. As illustrated in figure-2, CatBoost is an effective classifier technique that utilizes gradient boosting on decision trees and handles categorical characteristics in information.

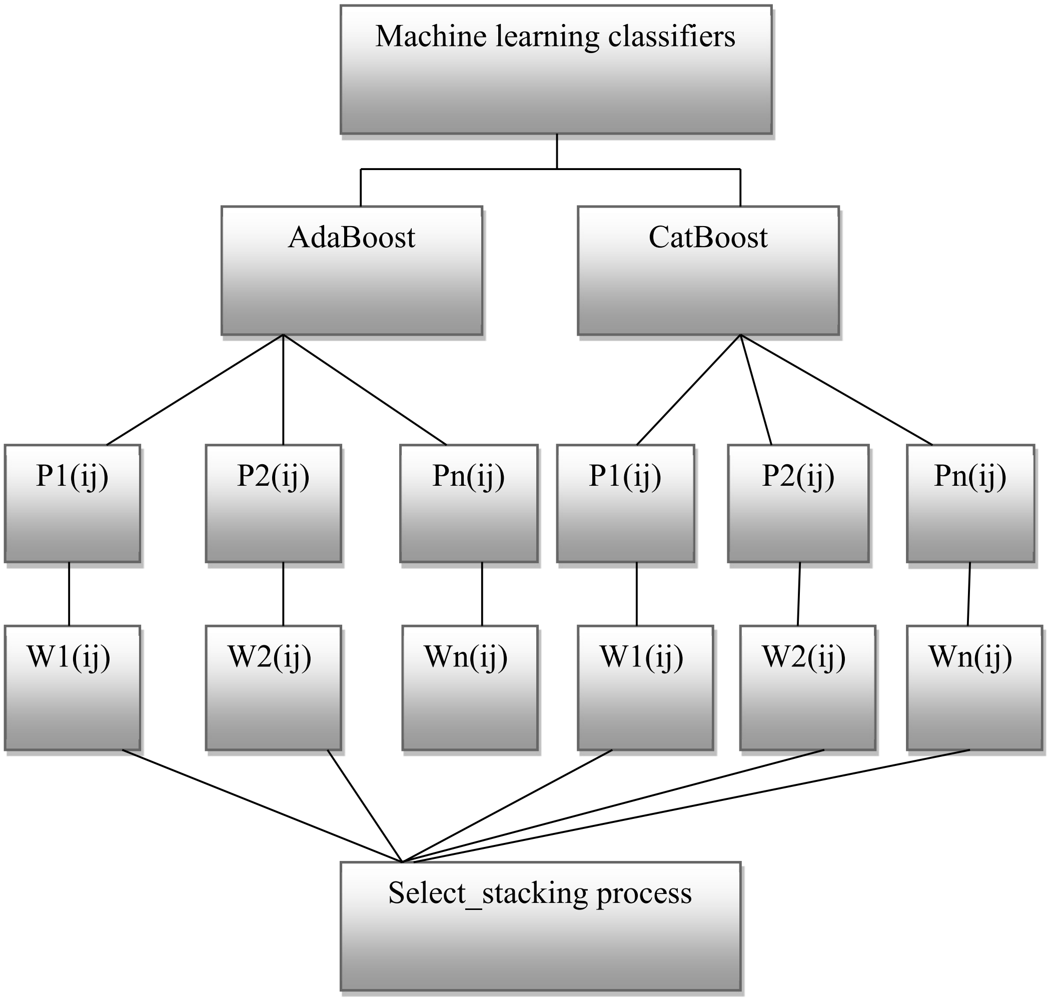

Figure-2 architecture of Sel_stack_AdaCat

It handles categorical information automatically utilizing statistical systems, whereas other systems require categorical information to be suited beforehand. CatBoost may avoid information overfitting by optimizing numerous input measurements. Weighted sampling happens at tree level rather than split into level in this approach. CatBoost was used to construct a balanced tree. The feature-split pair that results in the lowest loss is picked for every level of such a tree and is utilized for all level nodes. AdaBoost is a well-called algorithm in information science. Freund and Shapire invented it for 1st time in 1996 [18]. It is based on notion of boosting type ensemble system, in which numerous soft learners are joined to form a robust algorithm utilizing voting technique. The in-sample loss rate is no., of incorrectly categorized samples. (i,e.,) $(y_i \neq G(x_i))$, divided by total information samples' size (N), as given in Eq.,

$$\boldsymbol{error'} = \frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{I}(\,y_i \neq G(x_i))$$

Multiple soft learners are trained sequentially utilizing a successive modified version of information points in boosting. This means that during 1st boosting cycle, a soft learner is trained and prediction outcomes are gained, with some instances misclassified. In 2nd boosting cycle, a weight (W i) is applied to every example; previously misclassified records are weighted more heavily than successfully classified records in order to drive 2nd soft learner to learn and correctly categories them. Previously misclassified observations are now correctly classified by the second soft learner. After M iterations of this system, soft learners are paired with a robust meta-learner (G(x)) Meta-learner provides a prediction label to every record utilizing weighted majority voting technique described in Eq., below.

$$G(x) = sign\,(\sum_{m=1}^{M} \propto_m G_m\,(x))$$

Where α is importance of soft learners in ultimate majority voting system. Multiple soft learners are progressively trained in AdaBoost and CatBoost. These poor learners generate a meta-learner that predicts utilizing a weighted majority voting technique. More weights are allocated to previously incorrectly predicted samples in every boosting cycle. Sel-Stacking technique introduces feature selection procedure between base classifiers and metaclassifier by doing a global search to find the optimal collection of base classifiers. Sel-Stacking algorithm fusion algorithm's computational complexity is separated into 2 components. When utilizing M base learners to fit a information containing N rows of information, the initial step is K-fold stacking, which has temporal complexity of $O(K * \sum_{m=1}^{M} o_m$ time complexity of base classifiers m is m is Om. 2nd part trains beta learner with information generated by M base classifier with time complexity $O(2^M * O_{ada\_cat})$. Hence, time complexity of Sel-Stacking is given as

$$o(sel_{stack}) = o(K * \sum_{m=1}^{M} o_m + 2^M * O_{ada\_cat})$$

Performance analysis

The suggested technique combines 2 machine learning algorithms, AdaBoost and CatBoost, with a select- stacking classifier. The PIMA diabetes information was used in the experiments. The information consists of 769 pieces of data and ten characteristic columns, with "0" replaced with median values. The information has been divided into testing as well as learning information (20% and 70%, respectively). The most common assessment criteria used to evaluate algorithm robustness and efficacy are correctness, accuracy, recollection, and F1score. True positive (tp) signifies when the expected and actual class values are both 1. True negative (tn) means that the expected and actual class values are both 0. False negatives (fn) and false positives (fp) occur when your predicted class contradicts your actual class (fp). The most important measurement is accuracy, which is given as the percentage of total properly forecasted occurrences to total no., of observations. The formulas below are used to determine accuracy, reliability, recollection, and F1score.:

$$accuracy = \frac{tp + tn}{tn + tp + fp + fn}$$

$$precision = \frac{tp}{tp + fp}$$

$$recall = \frac{tp}{tp + fn}$$

$$f1score = \frac{2 \times precision \times recall}{precision + recall}$$

Measurements are compared utilizing 2 states of art systems like Super Learner Algorithm (SLM) [9] and fusion machine learning (FML) [16] with proposed stacked selecting AdaBoost and CatBoost (Sel_stack_AdaCat) system

Table-1 analysis of accuracy

No., of information	SLM	FML	Sel_stack_AdaCat
100	78.5	90.9	98.7
200	77.9	91.4	98
300	78.6	91.5	99
400	78.4	89.7	98.6
500	78	91	98

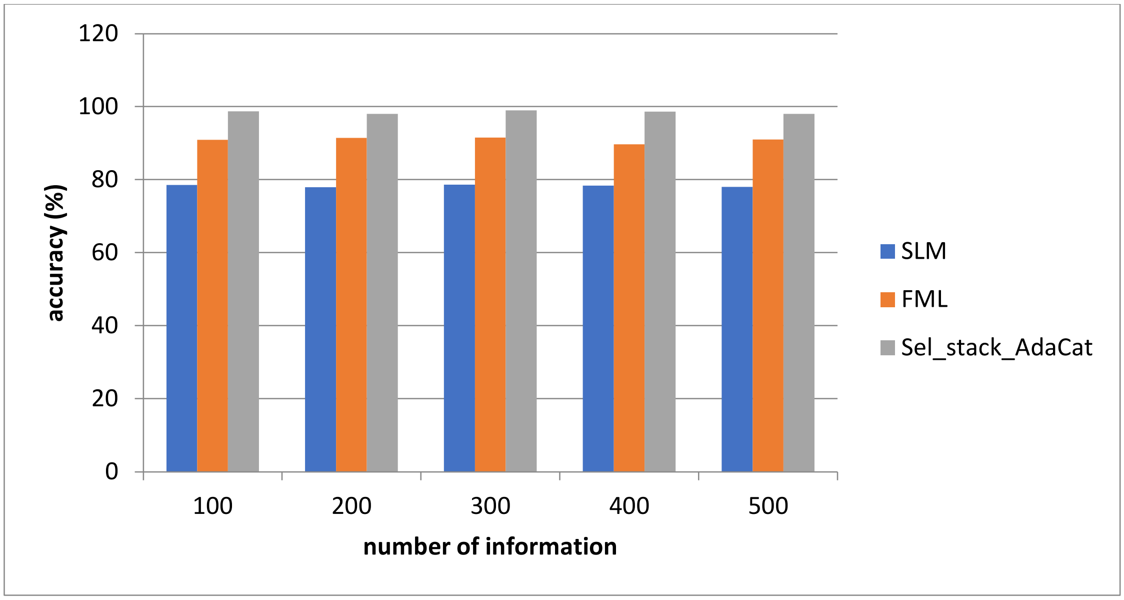

Figure-3 comparison of accuracy

Figure3 depicts a comparison of accuracy between current SLM, FML techniques and the proposed Sel stack AdaCat system, where Xaxis represents no., of information points utilized for analysis and Yaxis represents accuracy values achieved in percent. Existing SLM and FML techniques obtain 78.9 percent and 91.8 percent accuracy, respectively, whereas suggested Sel stack AdaCat approach achieves 98.7 percent accuracy, which is 10.2 percent better than SLM and 7.1 percent better than FML system.

Table-2 analysis of precision

No., of information	SLM	FML	Sel_stack_AdaCat
100	81	90.5	93.5
200	81.4	90.4	92.9
300	81.7	91	93.3
400	81	90.4	93.9
500	81.2	91	93.2

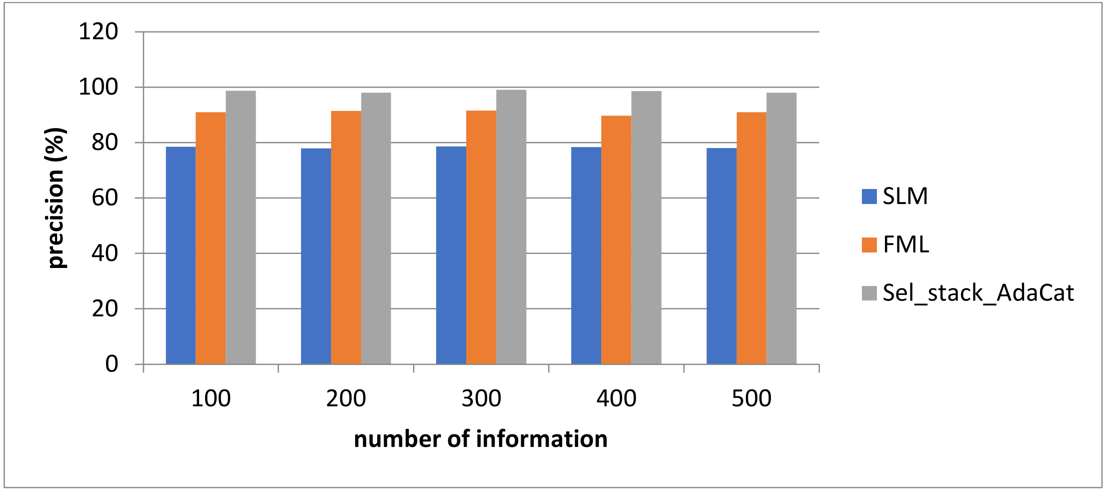

Figure-4 comparison of precision

Figure4 depicts a precision comparison of current SLM, FML techniques and the proposed Sel stack AdaCat system, where Xaxis represents no., of information utilized for analysis and Yaxis represents the accuracy values achieved in percent. Existing SLM and FML techniques produce 81.3 percent and 90.3 percent precision, respectively, whereas the suggested Sel stack AdaCat approach achieves 93.5 percent precision, which is 12.2 percent better than SLM and 3.2 percent better than FML system.

Table-3 analysis of recall

No., of information	SLM	FML	Sel_stack_AdaCat
100	79	85	89
200	78.9	87.8	89.4
300	79	85.7	88.4
400	79.4	86.2	88.4
500	78	85	89.7

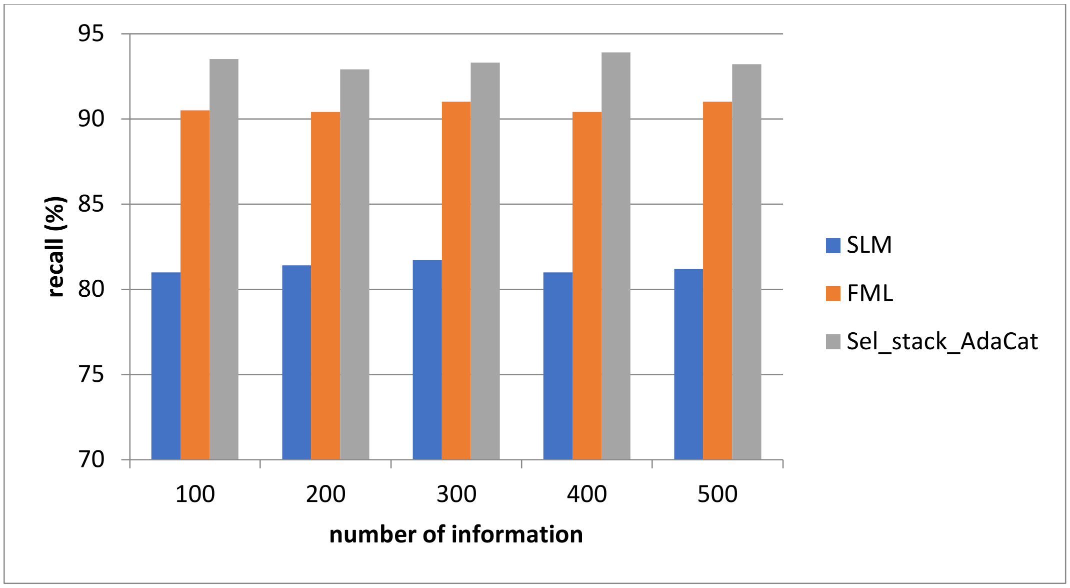

Figure-5 comparison of recall

Figure5 depicts a recall comparison of existing SLM, FML techniques and proposed Sel stack AdaCat system, where the Xaxis represents no., of information utilised for analysis and Yaxis represents recall values achieved in percent. When compared to current SLM and FML systems, suggested Sel stack AdaCat approach obtains 89.5 percent recall, which is 10.1 percent better than SLM and 3.1 percent better than FML system.

Table-4 analysis of F1-score

No., of information	SLM	FML	Sel_stack_AdaCat
100	82	84.3	88.9
200	82.5	83	88.5
300	82.4	83.8	88
400	81.9	84.9	87.4
500	81.8	85	88.3

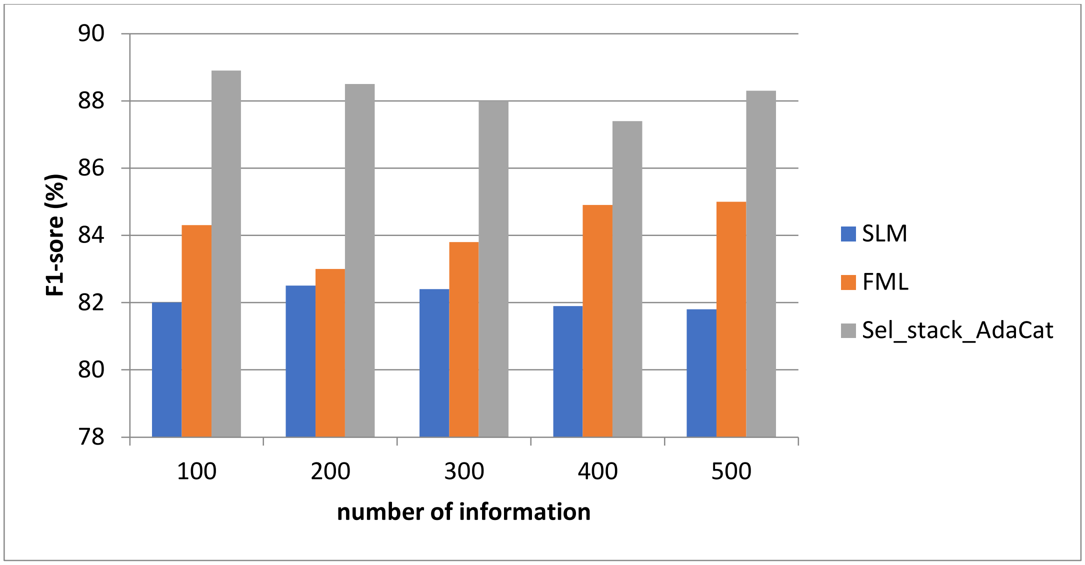

Figure -6 comparison of F1-score

Figure6 depicts a comparison of F1-score values acquired in percent between current SLM, FML techniques and proposed Sel stack AdaCat system, where Xaxis displays no., of information utilised for analysis and the Yaxis shows the F1-score values obtained in percentage. When compared to current SLM and FML techniques, the suggested Sel stack AdaCat approach obtains 84.7 percent of F1-score, which is 6.2 percent better than SLM and 4.5 percent better than FML system.

Table- 5 overall comparative analysis

Systems	**Accuracy (percent)**	**Precision (percent)**	**Recall (percent)**	**F1-score (percent)**
SLM	78.9	81.3	79.4	82.5
FML	91.8	90.3	85.4	84.7
Sel_stack_AdaCat	98.7	93.5	89.5	88.3

Conclusion

Diabetes mellitus is a condition that is becoming more common among individuals nowadays. As a result, detecting this illness early is crucial. The major purpose of this study is to discover the most accurate and systematic way to forecast diabetes patients. The Effectiveness ML methods deployed during the last five years was evaluated. As a consequence, authors created a selective stacking classifier method that is based on a combination of 2 ML algorithms, adaboost as well as catboost. For testing, Pima Indians diabetes information was employed. Sel stack AdaCat ensemble produced superior results. Future work will focus on including a feature selection strategy to lower the complexity of the informationset while increasing the no., of classifiers.

Reference

1. AD Association. Classifcation and diagnosis of diabetes: standards of medical care in diabetes-2020. Diabetes Care. 2019
2. International Diabetes Federation. Diabetes. Brussels: International Diabetes Federation; 2019.
3. Gregg EW, Sattar N, Ali MK. The changing face of diabetes complica- tions. Lancet Diabetes Endocrinol. 2016;4(6):537–47.
4. Maniruzzaman M, Kumar N, Abedin MM, Islam MS, Suri HS, El-Baz AS, Suri JS. Comparative approaches for classifcation of diabetes mellitus information: machine learning paradigm. Comput Systems Programs Biomed. 2017;152:23–34
5. Xie J, Liu Y, Zeng X, Zhang W, Mei Z. A Bayesian ne2rk algorithm for pre- dicting type2diabetes risk based on electronic health records. Modern Phys Lett B. 2017;31(19–21):1740055
6. Xie J, Liu Y, Zeng X, Zhang W, Mei Z. A Bayesian ne2rk algorithm for pre- dicting type2diabetes risk based on electronic health records. Modern Phys Lett B. 2017;31(19–21):1740055
7. K. V. Varma, A. A. Rao, T. S. Lakshmi, and P. N. Rao, "A Computational Intelligence approach for a better diagnosis of diabetic patients," Journal of Computers and Electrical Engineering, vol. 40, no. 5, pp. 1758–1765, 2014.
8. D. K. Choubey, M. Kumar, V. Shukla, S. Tripathi, and V. K. Dhandhania, "Comparative analysis of classification systems with PCA and LDA for diabetes," Current Diabetes Reviews, vol. 16, no. 8, pp. 833–850, 2020.
9. Doğru, A., Buyrukoğlu, S., & Arı, M. (2023). A hybrid super ensemble learning algorithm for the early-stage prediction of diabetes risk. *Medical & Biological Engineering & Computing*, 1-13.
10. Nijaguna, G. S., Babu, J. A., Parameshachari, B. D., de Prado, R. P., & Frnda, J. (2023). Quantum Fruit Fly algorithm and ResNet50-VGG16 for medical diagnosis. *Applied Soft Computing*, 110055.
11. Balasubramaniyan, S., Jeyakumar, V., & Nachimuthu, D. S. (2022). Panoramic tongue imaging and deep convolutional machine learning algorithm for diabetes diagnosis in humans. *Scientific Reports*, *12*(1), 186.
12. Naveena, S., & Bharathi, A. (2022). A new design of diabetes detection and glucose level prediction utilizing moth flame-based crow search deep learning. *Biomedical Signal Processing and Control*, *77*, 103748.
13. Olisah, C. C., Smith, L., & Smith, M. (2022). Diabetes mellitus prediction and diagnosis from a information preprocessing and machine learning perspective. *Computer Systems and Programs in Biomedicine*, *220*, 106773.
14. García-Ordás, M. T., Benavides, C., Benítez-Andrades, J. A., Alaiz-Moretón, H., & García-Rodríguez, I. (2021). Diabetes detection utilizing deep learning techniques with oversampling and feature augmentation. *Computer Systems and Programs in Biomedicine*, *202*, 105968.

15. Kopitar, L., Kocbek, P., Cilar, L., Sheikh, A., & Stiglic, G. (2020). Early detection of type2diabetes mellitus utilizing machine learning-based prediction algorithms. *Scientific reports*, *10*(1), 11981.
16. Ljubic, B., Hai, A. A., Stanojevic, M., Diaz, W., Polimac, D., Pavlovski, M., & Obradovic, Z. (2020). Predicting complications of diabetes mellitus utilizing advanced machine learning algorithms. *Journal of the American Medical Informatics Association*, *27*(9), 1343-1351.
17. Nadeem, M. W., Goh, H. G., Ponnusamy, V., Andonovic, I., Khan, M. A., & Hussain, M. (2021, October). A fusion-based machine learning approach for the prediction of the onset of diabetes. In *Healthcare* (Vol. 9, No. 10, p. 1393). MDPI.
18. Y. Freund and R. E. Schapire, ‘‘Experiments with a new boosting algorithm,’’ in Proc. Int. Conf. Mach. Learn., vol. 96, 1996, pp. 148–156