DIABETES AND TIME PREDICTION USING DIFFERENT MACHINE LEARNING CLASSIFIERS

Drakshayini C R[1], Mrs. Pushpalatha S[2] and Dr. G. F. Ali Ahammed[3]
[1]PG Scholar, [2]Assistant Professor,[3]Associate Professor
[1,2,3]Visvesvaraya Technological University Centre for Post Graduate Studies, Mysuru

Abstract: *Diabetes is also known as diabetes mellitus; it is a group of metabolic disorders which affected millions of people. The detection of diabetes is very importance, due to its severe complications. There are lot of research studies about diabetes identification, many of researches are based on the Pima Indian diabetes data set. The data set is based on the studying women in Pima Indian population which was started from 1965, where diabetes rates is comparatively high. Before many research studies done are mainly focused on one or two particular complex techniques to test the data, but comprehensive research over many common techniques is missing. This work predicts diabetes disease prediction as well as time prediction using machine learning algorithms. In machine algorithms we use two types of algorithms that is Naïve Bayes algorithm for diabetes prediction and K-nearest neighbour algorithm for time prediction. It provides better results on the trained data sets which can be lead to better performance in diabetes prediction. In this paper the source code for diabetes prediction is made by publicly available.*

Keywords: Diabetes prediction machine learning, Pima Indian diabetes data set, K-nearest neighbour(KNN), Naïve Bayes(NB).

I. INTRODUCTION

Diabetes also known as diabetes mellitus, it has a direct signal of high blood sugar, together with some symptoms including frequent urination, increased thirst, increased hunger and also weight loss. Patient of diabetes usually need constant treatment, elsewhere it will possibly lead to many dangerous life-threatening complications. The diabetes disease is diagnosed with the 2-hour post-load plasma glucose being at least 200mg/dL, and the necessity of identifying diabetes timely calls in various studies about diabetes recognition.

Previous research studies have been done in machine learning techniques for diabetes identification. and its result will be done in focused on the diabetes identification through Generalized Discriminant Analysis and Support Vector Machine. and they obtained some inspiring results. Another research was to do the same thing by GRNN (General Regression Neural Network) [3], which is also a very high accuracy. Comparing to the previous work, we make a more comprehensive study containing a number of common techniques used to diabetes identification, intending to compare their performance and find the best one among them.

The types of diabetics can be listed as Blurred vision, Fatigue, Weight Loss, Increased Hunger and Thirst, Frequent Urination, Confusion, Poor Healing, Frequent Infections, Difficulty in concentrating.

Type 1 Diabetics: Type 1 diabetics arises when our immune system obliterates beta cells in your pancreas, this is the cells that construct the insulin in our body. The insulin builds up in our blood and as a result, our cells are in a state of starvation which causes diabetics. It occurs

usually in people less than 30 years and about 5 - 10% of those with diabetics but can occur at any age.

Type 2 Diabetics: People who have type 2 diabetics secrete insulin, but their cells do not consume it as much as they should. The pancreas generates more insulin in order to obtain glucose into the cells since the cells do not make use of it properly sugar builds up in our bloodstream.

Figure 1. causes for diabetes

This experiment, we compare several common and data preprocessors for each of the classifiers we use, and find the best preprocessor respectively. Then we compare these classifiers after we modify the parameters of them to reach their approximate maximum accuracy, and we particularly analyse how to modify the parameters in data science. At last, we also analyze the relevance of each feature with the classification result, and this will help to modify the data set in future studies.

In our proposed system we are going to build this concept as real time application useful for the medical sector. Proposed system also adds an enhancement called “time prediction”. Initially proposed system predicts the diabetes disease where it classifies the new patient either to the class “YES” or “NO”. If the patient is classified to “NO” then system aims at predicting the “Time” of getting the diabetes disease. Here we develop this concept as generic application useful for multiple hospitals and we use more parameters for diabetes disease prediction in order to get more accurate results. In our proposed we use efficient data science algorithms for diabetes disease prediction and Time prediction.

Machine learning algorithms

The dataset used for this research work, The related analysis of our scheme against some different common possible algorithm in ML techniques. In this phase we have implemented Naïve bayes and K-nearest neighbour classification on the data set to classify each patient. Before performing ML algorithms, highly correlated attributes were found which has glucose and age. After implementation of this algorithm will get class labels of each record.

A. Naïve Bayes algorithm

The Naïve Classifier algorithm can be implemented as shown in Algorithm (3). Naïve Classifier may contain to predict class membership probabilities of Diabetes as normal or abnormal such as the probability that a given sample belongs to a particular class. Through Create Likelihood by finding the probabilities based on the Bayes theorem. The most widely used type of Bayesian Network for classification is the Naïve Bayesian's, which has the highest accuracy value of up to 99.51% respectively. The Bayesian Network applies the Naïve Bayes theorem which firmly assumes that the occurrence of any particular attribute in a class is not related to the presence of any other attribute, makes much advantageous, efficient and independent. The Naïve Bayesian is based on the conditional probability (given a set of features, the probability of occurrence of certain results.

The working of the Naïve Bayes algorithm

- Step 1: Scan the dataset, will get required data for mining from the servers such as database, cloud, excel sheet etc.
- Step 2: Calculate the probability of each samples value. [n, nc, m, p] Here for each attribute we calculate the probability of occurrence using the following formula is mentioned in the next step, For each class of disease we should apply the formulae.
- Step 3: Apply the formulae refer Eq.1

$$\frac{p(atributive\ value\ (ai)}{subject\ value\ (vj)} = \frac{nc+mp}{n+m} \quad (1)$$

  Where*:* n = the number of training examples for which v = vj, nc = number of examples for which v = vj and a = ai*,* p = a priori estimate for P(ai/vj) m = the equivalent sample size*.*
- Step 4: Multiplying the probabilities by p for each class, here we multiple the results of each attribute value with p and final results are used for classification.
- Step 5: Compare the values and classify the attribute values to define the predefined set of class.

B. K-nearest neighbour algorithm

KNN is the simplest Machine Learning algorithm based on Supervised Learning technique. KNN takes the same data between the new case data and available cases and put the new case data into the category that is most similar to the available categories. K-NN stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be classified into a well good category by using K- NN classifier. KNN can be used for Regression as well as for Classification but mostly it is used for the Classification problems. KNN is a non-parametric technique, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. KNN at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

The working of the K-NN algorithm

The K-NN working can be explained on the bases of below algorithm

- Step-1: Select the number K of the neighbors
- Step-2: Calculate the adopting postulates distance of K number of neighbors
- Step-3: Take the K nearest neighbors as per the calculated Euclidean distance.
- Step-4: Among these k neighbors, count the number of the data in each category.
- Step-5: Assign the new data to that category for which the number of the neighbor is maximum.
- Step-6: Our model is ready.

II. SYSTEM DESIGN

The purpose of the design is to plan a solution for problem faced by the requirements document. In this phase the first step helps in travelling from the problem phase to the solution phase. In other words, starting with the needs of model; design take us toward how to satisfy the needs. The design of a system is the most difficult factor affecting the quality of the software; it gives the major impact on the later phases especially in testing and maintenance. The design activity often results in three separate outputs –

- Architecture design.
- High level design.
- Detailed design.

According to Software Engineering the approach adopted to develop this project is the Iterative waterfall Model. The iterative waterfall Model is a systematic approach that begins at the feasibility study phase and progress through analysis, design, coding, testing, integration and maintenance. Feedback paths are there in each phase to its preceding phase. and to allow the correction of the errors committed during a phase that are detected in later phase. The context-level data flow diagram as show in the figure 2. first, which shows the interaction between the system and external agents which act as data sources and data sinks. The context diagram (also known as the 'Level 0 DFD') explains the system's interactions with the outside world are modeled purely in terms of data flows across the system boundary. The context diagram shows the entire system as a single process, and gives no clues as to its internal organization.

This context-level DFD is “exploded", to produce a Level 1 DFD that shows some of the detail explanation of system. The Level 1 DFD shows the system is division into sub-systems (processes), each of which deals with one or more of the data flows to or from an external agent, and which provide all of the functionality of the system as a whole. It identifies internal data stores that must be present in the system to do its job, and shows the flow of data between the different parts of the system.

In today’s world, health care industries are providing many benefits like fraud detection in health insurance, availability of medical facilities to patients at inexpensive prices, identification of smarter treatment methodologies, and construction of effective health-care policies, effective hospital resource management, better customer relation, improved patient care and hospital infection control. Stroke type detection is also one of the significant areas of research in medical. There is no self-operating for Stroke disease prediction.

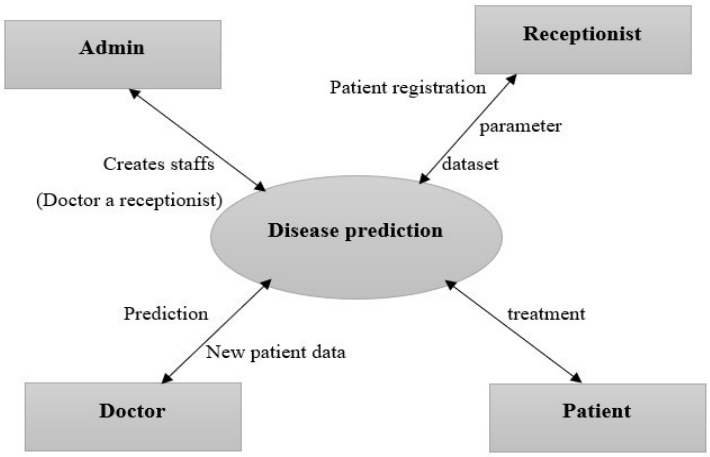

Figure 2. context flow diagram (level 0)

In context dataflow diagram contains some following information.

- **Staff Creation Module (Admin)**: Administrator of the system creates the staffs (specialist, receptionist) and manages the staffs and sets the unique id and password for each staff.
- **Patient Registration Module (Receptionist)**: Receptionist of the hospital registers the patients by collecting the patient details such as name, address, contact no, email id etc… receptionist sets the patient Id and password for each patient for future use.
- **Parameters Module (Receptionist)**: Receptionists of the hospital manages the different constraints required for the prediction of Stroke disease. Basically, there are n number of constraints related to Stroke disease prediction.
- **Data-set Module (Receptionist)**: Receptionist manages the data-set required for the Stroke disease prediction. Here receptionist uploads the old data into server which includes Stroke disease patient's data with related constraints/parameters and results.
- **Input Module - New Patient (Doctor)**: Stroke Disease Specialist uploads the new patient constraints, based on these constraints system will predict the output.
- **Prediction Module (Doctor)**: This is the core module of the project where system accepts the input given by the disease specialist. This module predicts the final output weather patient is classified to "Yes" or "No". We make use of classification rules techniques for the output prediction which is one the efficient technique which works fine for small data-set as well as huge data-set.
- **Treatment Module (Doctors, Patients)**: This module maintained by the Specialist where specialist uploads the treatment details for the patients and patients can view the treatment details.
- **Account Module (Admin, Receptionist, Doctor, and Patient)**: This is a common module of all actors where they can manage their profile by updating, changing passwords etc…

Data flow diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing (structured

design). On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process. A DFD provides no information about the timing of processes, or about whether processes will operate in sequence or in parallel. It is therefore quite different from a flowchart, which shows the flow of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored (all of which are shown on a DFD).

Level 1 (high level diagram):

This level (level 1) consists of all processes at the first level of numbering, data stores, external entities and the data flows between them. The reason of this level is to provide the main and high-level processes of the system and their interrelation. A process model contains only one, level-1 diagram. A level-1 diagram must be equal with its parent context level diagram that is there must be the same external entities and the same data flows, these can be broken down to multiple detail in the level1.

1. **Patient: Patient** is a one who receives the services from the application. Patients can access to treatment details is shown in figure 3.
2. **Doctor (Disease Specialist):** Doctor is a one who specifies the necessary inputs for disease prediction. Doctor is a service receiver. The key service given by the system is "diabetes Disease Prediction" based on the medical data. This is shown in figure 4.
3. **Admin:** Administrator is a one who maintains the entire application. Administrator is a owner of the application is shown in figure 5.
4. **Receptionist:** Receptionist is one who maintains the patient's registration, billing and treatment details is shown in figure 6.

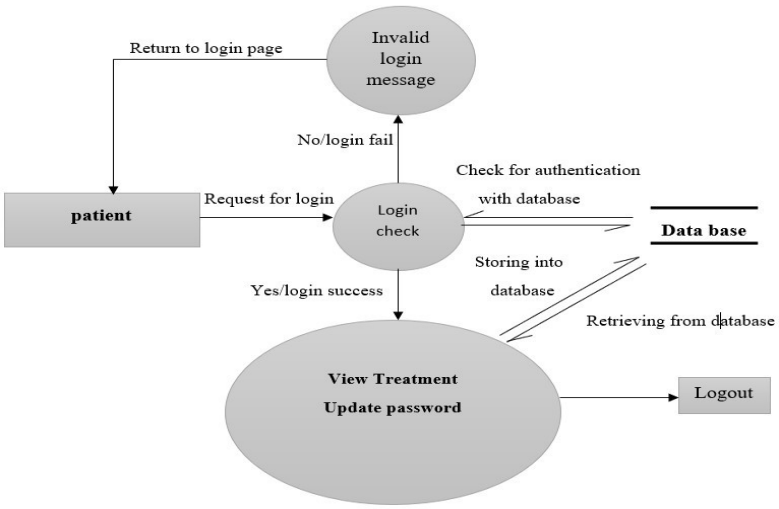

Figure 3. Dataflow diagram of patient

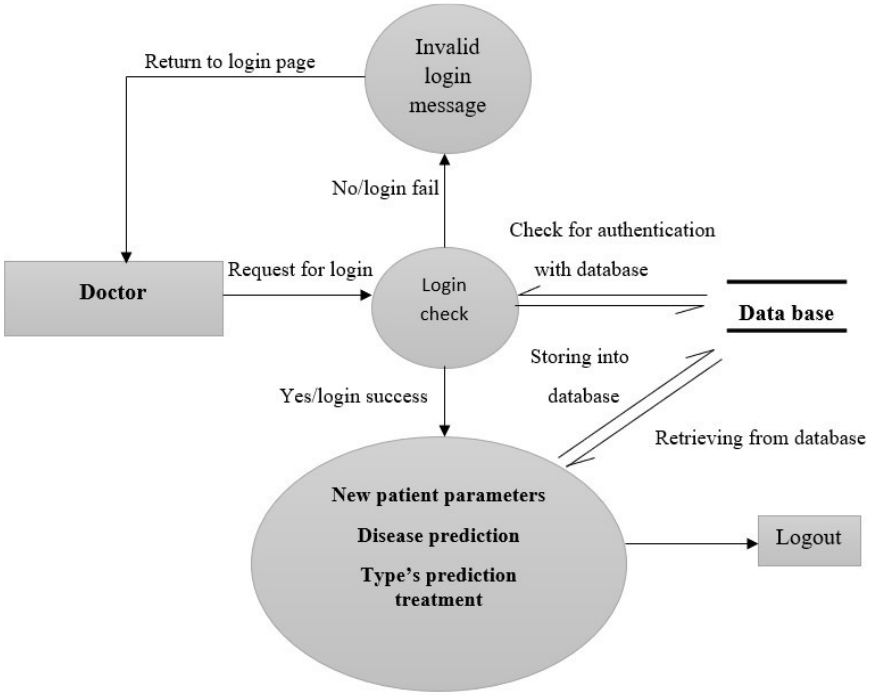

Figure 4. Dataflow diagram of doctor

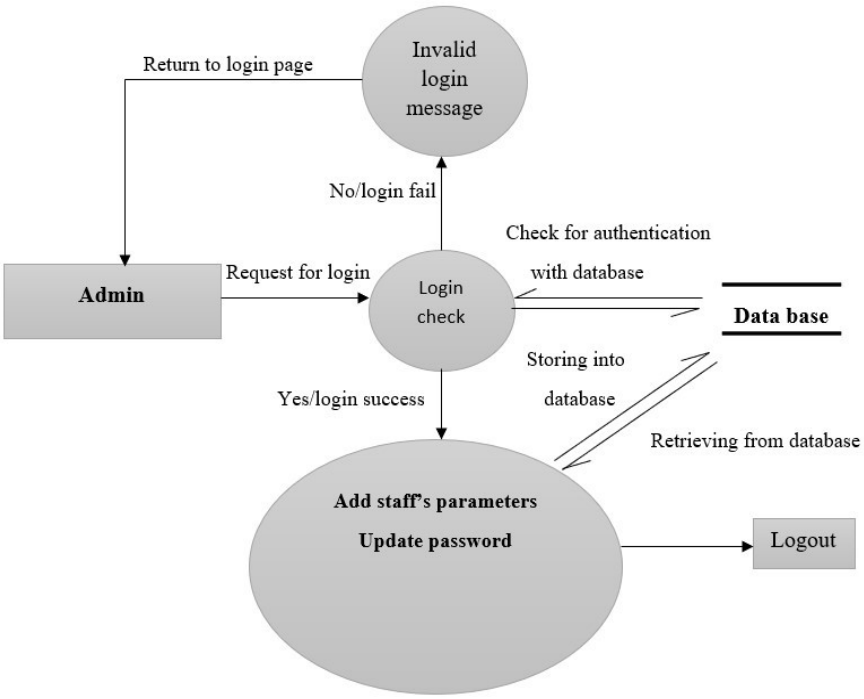

Figure 5. Dataflow diagram of Admin

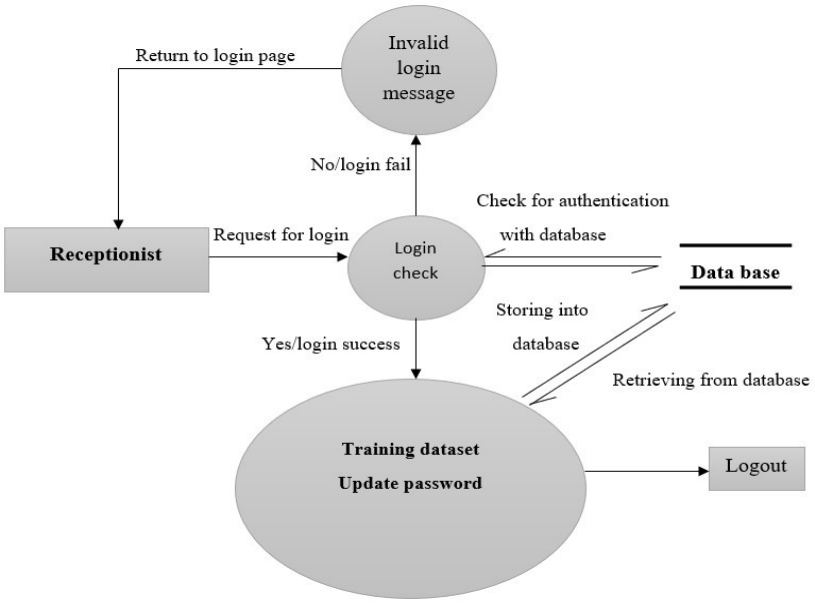

Figure 6. Dataflow diagram of receptionist.

III. RESULTS

This section shows the performance results among benchmark machine learning models and Diabetes level prediction model depending up on the metrics as follows.

Performance evaluation of machine learning techniques: From Table 1, out of these machine learning algorithms, the naïve Bayes was found out to be the best performer with an accuracy of 94.23%, so, we chose the NB classifier gives our prediction algorithm and improves it to increase the prediction accuracy.

Table 1. Performance evaluation of machine learning techniques comparing with previous result

Model	Accuracy
Decision Tree	93.12
LDA	76.77
Logistic Regression	82.5
SVM	83.22
AdaBoost	78.02
Naïve Bayes	94.23

Performance result of Naïve Bayes

We have proposed the Naïve Bayes algorithm which uses three base estimators to improvise and have a much better accuracy of the result to prove the same we have showcased the result of Naïve Bayes in table 1.

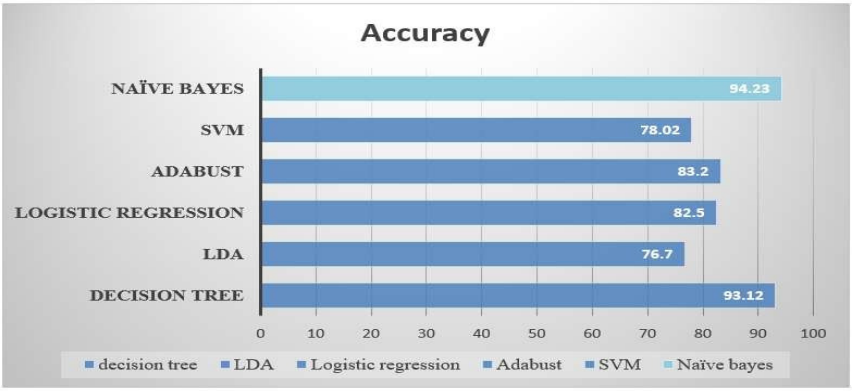

Figure 7. Performance analysis of machine learning model

Table 2. Performance metrics of Naïve Bayes

Performance Metrics Basic	Naïve Bayes
Accuracy	94.23
Precision	91.82
Sensitivity	92.16
Specificity	95.07

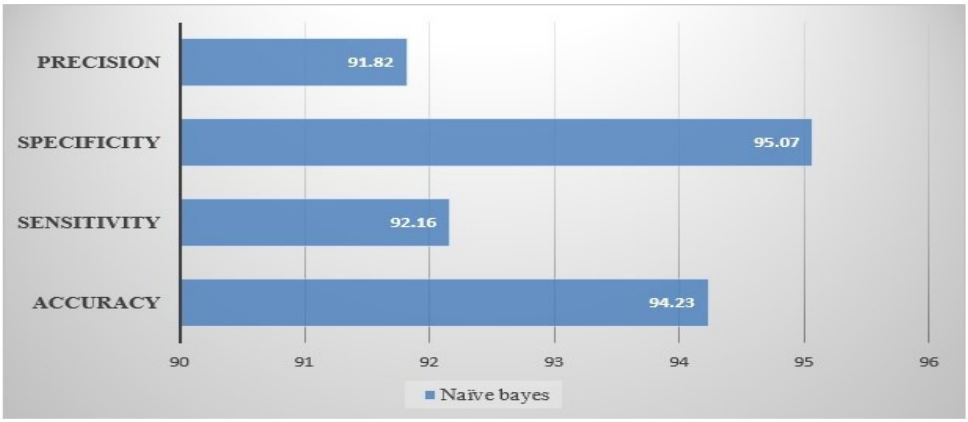

Figure 8. performance analysis of proposed model using Naïve Bayes

By using the proposed system, we can identify the risk level of diabetes prediction along with the classification of disease levels with an accuracy of 94.23%. Table 2 demonstrates the performance Naïve Bayes models. The visual representation of performance metrics is shown in Figure 7.

Performance result of KNN Algorithm

The Sensitivity, Specificity, Precision and Accuracy of the K nearest neighbor, have been shown for different values in Table 3. three important parameters of Accuracy, Sensitivity and Precision were perused. The figures represent the K nearest algorithm, Specificity and Precision decreased after K increment, and then Accuracy decreased a little. In addition, if there is K increment, Sensitivity increased a little and the number of patients who were truly diagnosed increased but simultaneously false positive patients also increased. This problem shows the Precision reduction as K increased. The best results are pertaining to the KNN algorithm that outruns the K nearest neighbor algorithm in the Accuracy, Precision and Specificity criteria by a small difference.

Table 3. Performance metrics of KNN

Algorithm	Sensitivity (%)	Specificity (%)	Precision (%)	Accuracy (%)
KNN with K =1	91.8	89.2	95.8	92
KNN with k=5	93.08	93.45	97.6	94.6
KNN with k=7	96.5	93.1	97.9	96

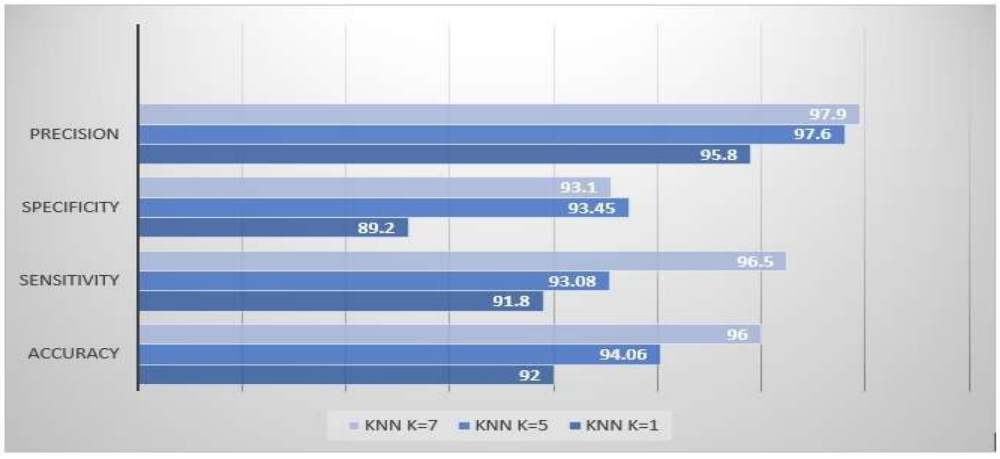

Figure 9. performance analysis of proposed model using KNN

IV. CONCLUSION

The study was made using two machine learning algorithms Naïve Bayes and KNN. An approach for conservation of these two algorithms is necessary such that they continue to provide various services. These algorithms are a home various wild life species and medical plants. This two algorithms NB and KNN can be used in order to predict diabetes and time in high-risk groups. The NB algorithm with diabetes predication is implemented in this work to identify the risk factor. An accuracy of 94.23% is achieved through diabetes predictor. The KNN algorithm with time predication is implemented in this work to identify the risk factor. An accuracy of 96% is achieved through diabetes predictor. As future research, we can derive methods for different types of diabetes along with risk levels using an image dataset.

V. REFERENCES

[1] World Health Organization, "Report of a study group: Diabetes Mellitus," World Health Organization Technical Report Series, Geneva, 727, 1985.

[2] Kemal Polat, Salih Gunes, and Ahmet Arslan, "A cascade learning system for classification of diabetes disease: Generalized Discriminant Analysis and Least Square Support Vector Machine," Expert Systems with Applications, vol. 34. 1, January. 2008, pp. 482-487.

[3] Kayaer K and Yildirim T, "Medical diagnosis on Pima Indian diabetes using general regression neural networks," Proceedings of the international conference on artificial neural networks and neural information processing, 2003, pp. 181-184.

[4] Jack W. Smith, J. E. Everhart, W. C. Dickson, W. C. Knowler, and R. S. Johannes, "Using the ADAP learning algorithm to forecast the onset of diabetes mellitus," Proc. Annu. Symp. Comput. Appl. Med. Care, November 9. 1988, pp. 261-265.

[5] Karegowda A. G., Manjunath A. S. and Jayaram M. A., "Application of genetic algorithm optimized neural network connection weights for medical diagnosis of pima Indians diabetes," International Journal on Soft Computing, vol. 2. 2, 2011, pp. 15-23.

[6] Carpenter G. A. and Markuzon N., "ARTMAP-IC and medical diagnosis: Instance counting and inconsistent cases," Neural Networks, vol. 11. 2, 1998, pp. 323-336.

[7] Wold S., Esbensen K. and Geladi P., "Principal component analysis," Chemometrics and intelligent laboratory systems, vol. 2. 1-3, 1987, pp. 37-52.

[8] Balakrishnama S. and Ganapathiraju A., "Linear discriminant analysis-a brief tutorial," Institute for Signal and information Processing, vol. 18, 1998.

[9] Deng L. and Yu D., "Deep learning: methods and applications," Foundations and Trends in Signal Processing, vol. 7. 3-4, 2014, pp. 197-387.

[10] Lee H., "Tutorial on deep learning and applications," NIPS 2010 Workshop on Deep Learning and Unsupervised Feature Learning, 2010.

[11] Safavian S. R. and Landgrebe D., “A survey of decision tree classifier methodology,” IEEE transactions on systems, man, and cybernetics, vol. 21. 3, 1991, pp. 660-674.

[12] Suykens J. A. K. and Vandewalle J., “Least squares support vector machine classifiers,” Neural processing letters, vol. 9. 3, 1999, pp. 293-300.