Monografico

DOI: http://dx.doi.org/10.25115/eea.v39i4.4475

Estudios de Economía Aplicada
Volumen:39-4 // ISSN: 1133-3197

A comparison of Machine Learning Methods to predict Hospital Readmission of Diabetic Patient

LE DINH PHU CUONG[1,2,*], DONG WANG[1]
[1]College of Computer Science and Electronic Engineering, HUNAN UNIVERSITY, CHINA
[2]YERSIN UNIVERSITY, VIETNAM
[*]E-mail: cuongldp@yersin.edu.vn

ABSTRACT

Diabetes is a chronic disease whereby blood glucose is not metabolized in the body. Electronic health records (EHRs) for each individual or a population have become important to standing developing trends of diseases. Machine/Deep Learning helps provide accurate predictions higher than actual assessments. The main problem that we are trying to apply Machine/Deep learning model and using EHRs that combines the strength of a machine learning model with various features and Hyper-parameter optimization or tuning. The Hyper-parameter optimization uses the random search optimization which minimizes a predefined loss function on given independent data. The evaluation on the method comparisons indicated that Machine/Deep Learning models (Logistic Regression, Artificial Neural Network, Naïve Bayesian Classifier, Support Vector Machine and XGBoost**)** has improved results compared to the majority of previous models increasing the ratio of metrics (Accuracy, Recall, F1 and AUC score) on the same public dataset that is reprocessed. This shows that the proposed XGBoost model implemented in Amazon SageMaker (Amazon SageMaker was a Cloud Computing service) has the best performance evaluation results. This work is also one of the contributions to the global economic recovery in general and the reduction of medical equipment supply for the care and treatment of diabetics in particular during the Covid-19 pandemic.

Keywords: Amazon SageMaker (AWS); Machine/Deep Learning; Diabetes; Cloud Computing service; Economic recovery

JEL Classification: C60

Recibido: 14 de Enero de 2020
Aceptado: 16 de Marzo de 2021

1. Introduction

Diabetes has become a common public health problem in developing and developing countries around the world. The International Diabetes Federation (IDF) estimated that around 425 million people had diabetes in 2017 and this statistic is expected to increase significantly to 629 million globally by 2045 [43], [44]. Currently, during the global Covid-19 pandemic [45], it is also one of the background diseases leading to the highest mortality rate among people infected with Covid-19 virus (*new called SARS-CoV-2 or Coronavirus 2*) during treatment. Moreover, the Covid-19 pandemic has seriously affected all countries of the world and created many challenges and recessions such as Lockdown, social distance, and quarantine conditions have restricted all the business routine [46]. Therefore, we consider diabetes is one of the main factors affecting the global economy by providing the necessary medical support equipment compared to those without or with other background diseases as we know it well.

So, that factor is one of our research subjects covered in this paper. Diabetes has a significant incidence of illness, health care effects and lead to a high mortality rate due to its complications in both developed and developing countries. At present, diabetes treatments are not completely adequate and costly, so we recommend that prevention is an important factor in reducing the burden of diabetes and its complications [37]. Diabetes can be divided into three main types: Type 1 diabetes, Type 2 diabetes and gestational diabetes, the most common type is Type 2 diabetes (T2D). Currently, follow-up treatments for patients who have been diagnosed with diabetes or pre-diabetes to provide a diabetes treatment with an accurate diagnosis of T2D. T2D is a deficiency of insulin from the secretion by the spleen. In the absence of insulin, the un-metabolized sugar leads to a buildup in the bloodstream, causing blood glucose levels are high referred to as hyperglycemia [38].

Machine learning, which is one of the most important branches of artificial intelligence, provides methods and techniques for learning from experience [3]. Researchers often use it for complex statistical analysis tasks [4]. It is a wide multidisciplinary domain which is based on numerous disciplines including, but not limited to, data processing, statistics, algebra, knowledge analytics, information theory, control theory, biology, statistics, cognitive science, philosophy, and complexity of computations. This field plays an important role in term of discovering valuable knowledge from databases which could contain records of supply maintenance, medical records, financial transactions, applications of loans, etc. [5].

Machine learning techniques can be broadly classified into three main categories [3]. Supervised learning techniques involve learning from training data, guided by the data scientist. There are two basic types of learning missions: classification and regression. Models of classification attempt to predict distinguished classes, such as blood groups, while models of regression prognosticate numerical values [3]. In unsupervised learning, on the other hand, the system could attempt to find hidden data patterns, associations among features or variables, or data trends [3], [4]. The main objective of unsupervised learning is the ability to specify hidden structures or data distributions without being subject to supervision or the prior categorization of the training data [6]. Finally, in reinforcement learning the system attempts to learn through interactions (trial and error) with a dynamic environment. During this learning mode, the computer program provides access to a dynamic environment in order to perform a specific objective. It is worth noting that in this case, the system does not have prior knowledge regarding the environments behavior, and the only way to figure it out is through trial and error [3], [7], [8].

The goal of this paper is to apply machine learning techniques, and specifically prediction techniques, for predicting the likelihood of readmission of patients to hospitals. This problem hasn't been adequately addressed in the literature. In fact most research efforts are oriented towards prediction of diseases. Machine learning includes numerous analytic techniques for prediction and the literature lacks adequate comparative studies that assist in selecting a suitable technique for this

purpose. Our research is based on a large data set collected by numerous United States hospitals [11], [12].

2. Background

Table 1 The reviewed results from previous researcher

Reference Used	Applied Method	Achieved Prediction	Accuracy (%)	Limitations
Vijayan, V. V. et al. [47]	Adaptive Boosting (AdaBoost) algorithm with Decision Stump, Support Vector Machine (SVM), Naïve Bayes, Decision Tree	Diabetes, the best performance of AdaBoost with accuracy of 80.72%	80.72	No value results of evaluation metrics (Recall, F1-Score, Precision, AUC). Accuracy value is less than our research value.
Zou, Q. et al. [48]	Decision tree and Random Forest were implemented in WEKA tool and Neural Network is implemented in MATLAB. Feature Selection (used PCA and mRMR)	Diabetes Mellitus, gave a performance comparison these algorithms showed that Random Forest was the best algorithm with accuracy 80%. Besides, features engineering for dataset used PCA and mRMR. Value result of evaluation metrics result (Sensitivity, Specificity, ACC and MCC).	80	No value results of evaluation metrics (Recall, F1-Score, Precision, AUC). Accuracy value is less than our research value.
Dwivedi, A. K. [49]	Artificial Neural Network (ANN), Logistic Regression (LR), Classification Tree, Support Vector Machine (SVM), Naïve Bayes, K-Nearest Neighbor (KNN)	Diabetes, gave a performance comparison these classification performance measurements showed that ANN and LR were the best model with accuracy of 77% and 78%, with F1 measure of 0.83 and 0.84, respectively. Especially, it gave time result to build each model. It gave result of other evaluation metric (RMC) and features engineering.	78	No value results of evaluation metrics (Recall, Precision and AUC). Accuracy value is less than our research value.
Vigneswari, D. et al. [50]	Random Forest (RF), C4.5, Random Tree (RT), REP Tree and Logistic Model Tree (LMT). Preprocessed data and using WEKA tool.	Diabetes Mellitus. The performance of LMT (accuracy of 79.31%) is better than RF (accuracy of 78.54%). Besides features engineering, it shows execution time of the classifiers.	79.31	Accuracy value is less than our research value.
Daanouni, O. et al. [51]	Machine learning algorithms (Decision Tree, KNN, ANN, and Deep Neural Network).	Diabetes Mellitus. The results compared using different similarity metrics like Accuracy, Sensitivity, and Specificity and ROC (Receiver Operating Curve) gives best performance with respect to state of the art.		No value results of evaluation metrics (Recall, Precision and F1). The two different diabetes databases with the aim of evaluating the model are clear but this is considered a complex task. Accuracy value is less than our research value.

This section discusses the dataset and five basic machine learning techniques used in this study. In this work, the application of the XGBoost model is implemented on AWS machine learning platform, besides, a comparison with other model includes Logistic Regression, Artificial Neural Network, Naïve

Bayesian Classifier and Support Vector Machine. We performed each model based on the preprocessed dataset with Hyper-parameter optimization or tuning settings performed numerous times to give the best possible evaluation results. The best evaluation results on each model provided a graph of the model's accuracy for training and validation sets and performance evaluation indicators to test those models. Besides, we also highlight the theory of each model in sections from 2.2 to 2.6. This shows the application of the Machine/Deep learning models has improved results compared to the majority of previous models in Table 1 (Adaptive Boosting, K-Nearest Neighbors, Random Forest, Support Vector Machine, C4.5, Decision Tree, Random Tree, REP Tree, Logistic Model Tree, Neural Network and Artificial Neural Network) or prediction technology (WEKA and MATLAB) [47], [48], [49], [50], [51] increasing the ratio of metrics (Accuracy, Recall, F1 and AUC score) on the same public dataset that is reprocessed and described in detail in section 2.1. Besides, the proposed method shows in Figure 1 below.

In summary, we provide a table and a graph of results in the Hospital Readmission of Diabetic Patient prediction described in detail in section 3.

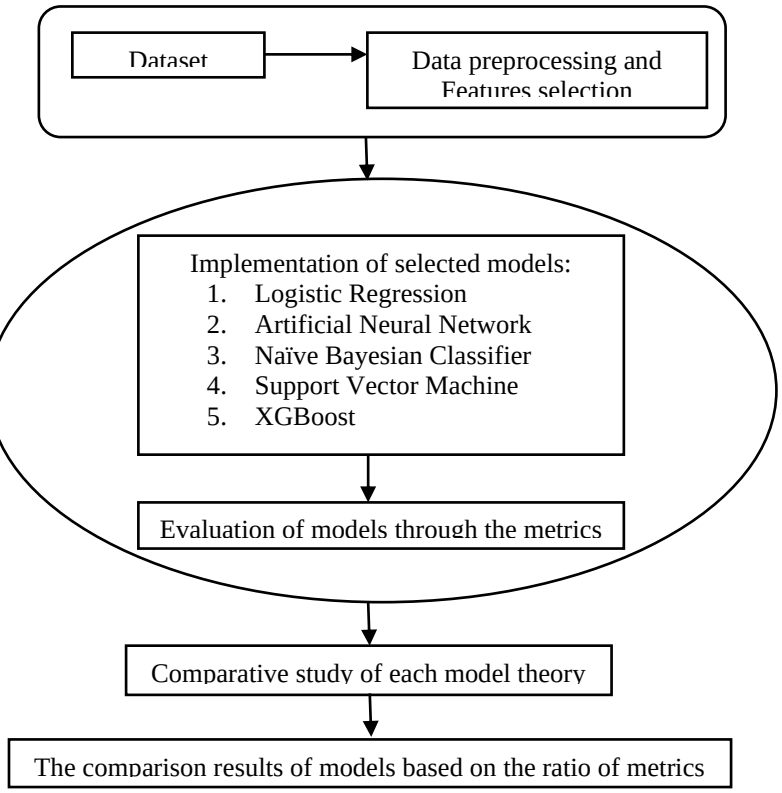


Figure 1 Proposed Methodology

2.1. Diabetes Dataset

The data set represents 10 years (1999-2008) of clinical care at 130 US hospitals and integrated delivery networks. It includes over 50 features representing patient and hospital outcomes. Information was extracted from the database for encounters that satisfied the following criteria. This data is a de-identified abstract of the Health Facts database (Cerner Corporation, Kansas City, MO) [4].

The dataset contains 101,767 records of the patients with eight attributes and one class variable. The attribute information is given in Table 2. The data contains such attributes as encounter, patient number, race, gender, age, weigh, admission type, time in hospital, medical specialty of admitting physician, number of lab test performed, HbA1c test result, diagnosis, number of medication, diabetic medications, a number of outpatient, inpatient, and emergency visits in the year before the hospitalization, etc."

Table 2 The similar as attribute information of Diabetes Dataset

	encounter_id	patient_nbr	race	gender	age	weight	admission_type_id	time_in_hospital	…
1	2278392	8222157	Caucasian	Female	[0-10)	?	6	1	…
2	149190	55629189	Caucasian	Female	[10-20)	?	1	3	…
3	64410	86047875	African American	Female	[20-30)	?	1	2	…
4	500364	82442376	Caucasian	Male	[30-40)	?	1	2	…
5	16680	42519267	Caucasian	Male	[40-50)	?	1	1	…
6	35754	82637451	Caucasian	Male	[50-60)	?	2	3	…
7	55842	84259809	Caucasian	Male	[60-70)	?	3	4	…
8	63768	114882984	Caucasian	Male	[70-80)	?	1	5	…
9	12522	48330783	Caucasian	Female	[80-90)	?	2	13	…
10	15738	63555939	Caucasian	Female	[90-100)	?	3	12	…
…	…	…	…	…	…	…	…	…	…

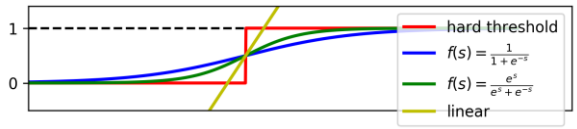
2.2. Logistic Regression

Regression is a statistical notion that can be used to identify the relationship weight between one variable called the dependent variable and a group of other changeable variables denoted as the independent variables. Logistic regression (LR) is a non-linear regression model, used to estimate the likelihood that an event will occur as a function of others [13].

The predicted output of logistic regression is usually written in the form:

$$f(\mathbf{x}) = \theta(\mathbf{w}^T\mathbf{x}) \tag{1}$$

Where ϑ is called the logistic function. Some activation for the linear model is shown in the figure below:



Among the above functions, the sigmoid function most used, as it is blocked in the range (0, 1):

$$f(s) = \frac{1}{1+e^{-s}} \triangleq \sigma(s) \tag{2}$$

With the above model, we can assume that the probability that a data point x falls in class *1* is $f(w^Tx)$ and falls in class *0* is $1 - f(w^Tx)$. With such a model, with training data points (known output y), we could write the following:

$$P(y_i|\mathbf{x}_i;\mathbf{w}) = z_i^{y_i}(1-z_i)^{1-y_i} \tag{3}$$

Where:

$$z_i = f(\mathbf{w}^T\mathbf{x}_i) \tag{4}$$

Considering the entire training set with $X = [x_1, x_2,..., x_N] \in R^{d \times N}$ and $y = [y_1, y_2,..., y_N]$, we need to find w so that the following expression has its maximum value P(**y|X;w**)

Here, we also denote **X, y** just like random variables, In other words:

$$\mathbf{w} = \arg\max_{\mathbf{w}} P(\mathbf{y}|\mathbf{X};\mathbf{w}) \tag{5}$$

The problem of finding the parameters for the model closest to the data above is collectively called the maximum likelihood estimation problem with the function behind *arg max* called the likelihood function.

2.3. Artificial Neural Network

An Artificial Neural Network (ANN) is a computational model which attempts to emulate the human brain parallel processing nature. An ANN is a network of strongly interconnected processing elements (neurons), which operate in parallel [14] inspired by the biological nervous systems [15]. ANNs are broadly used in many researches because they are capable of modeling non-linear systems, where relationships among variables are either unknown of quite complicated [14]. An example of an ANN is the Multi-Layer Perceptron (MLP), which is typically formed of three layers of neurons (input layer, output layer, and hidden layer) and its neurons use nonlinear functions for data processing [16].

Layers: In addition to Input layers and Output layers, a Multi-layer Perceptron (MLP) can have multiple Hidden layers in between. Hidden layers in the order from input layer to output layer are numbered as Hidden layer 1, Hidden layer 2, so on. Figure 2 below is an example with 2 Hidden layers.

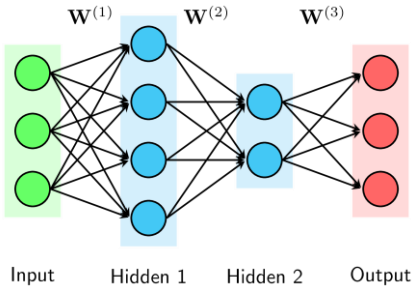


Figure 2 MLP with two hidden layers

Units: A circular node in a layer is called a unit. Units in the input layers, hidden layers, and output layers are called the input unit, hidden unit, and output unit respectively. The inputs of the hidden layers are denoted by *z*, the output of each unit is usually denoted *a* (representing activation, which is the value of each unit after we apply the activation function to *z*). The output of the i^{th} unit in layer *l* is denoted $a^{(l)}_i$. Suppose further that the number of units in the l^{th} layer) (excluding bias) is $d^{(l)}$. The vector represents the output of the first layer, denoted $a^{(l)} \in R^{d(l)}$.

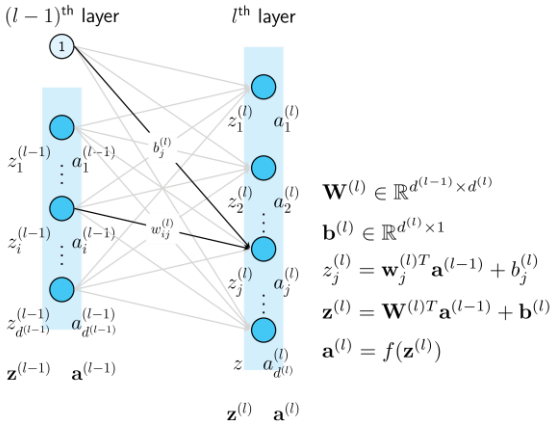


Figure 3 Symbols used in MLP

Weights and Biases: There is *L* weighting matrices for an MLP with *L* layers. These matrices are denoted as $W(l) \in R^{d(l-1)\times d(l)}$, $l=1,2,\ldots, L$ where $W^{(l)}$ represents the connections from the th layer *1* to the l^{th} layer (if we consider the input layer as the secondary layer 0). More specifically, the element $w^{(l)}_{ij}$ represents the connection from the i^{th} node of the th layer *(l−1)* to the node from *j* of the layer *(l)*. The biases of the th layer *(l)* are denoted $b^{(l)} \in R^{d(l)}$. These weights are symbolized as shown in Figure 3. When optimizing an MLP for a given job, we need to find these weights and biases. The set of weights and biases are denoted **W** and **b**

Activation Functions: Each output of a unit (minus input units) is calculated using the formula:

$$a_i^{(l)} = f(\mathbf{w}_i^{(l)T}\mathbf{a}^{(l-1)} + b_i^{(l)}) \tag{6}$$

In which *f(.)* is a (nonlinear) activation function. In vector form, the above expression is written as:

$$\mathbf{a}^{(l)} = f(\mathbf{W}^{(l)T}\mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}) \tag{7}$$

When activation function *f(.)* Is applied to a matrix (or vector), we understand that it is applicable to each component of that matrix. These components are then rearranged in the correct order to be a matrix of the same size as the input matrix.

Back-propagation: The most common method to optimize the MLP is still Gradient Descent (GD). To apply GD, we need to calculate the gradient of the loss function for each weight matrix $W^{(l)}$ and vector bias $b^{(l)}$. First, we need to calculate predicted output *y^* with an input *x*:

$$\begin{aligned}
\mathbf{a}^{(0)} &= \mathbf{x} \\
z_i^{(l)} &= \mathbf{w}_i^{(l)T}\mathbf{a}^{(l-1)} + b_i^{(l)} \\
\mathbf{z}^{(l)} &= \mathbf{W}^{(l)T}\mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}, \;\; l = 1, 2, \ldots, L \\
\mathbf{a}^{(l)} &= f(\mathbf{z}^{(l)}), \;\; l = 1, 2, \ldots, L \\
\hat{\mathbf{y}} &= \mathbf{a}^{(L)}
\end{aligned} \tag{8}$$

This step is wrapped as feed forward because the computation is done from start to finish of the network. MLP is also called.

2.4. Naïve Bayesian Classifier

Naïve Bayesian (NB) classifier relies on applying Bayes‟ theorem to estimate the most probable membership of a given event in one of a set of possible classes. It is described as being naïve, since it assumes independence among variables used in the classification process [15], [17], [18].

Consider classification problem with C classes 1, 2,…, C. Suppose there is one $x \in R^d$ data point. Calculate the probability that this data point falls in class c. In other words, calculate:

$$p(y = c|\mathbf{x}) \tag{9}$$

This expression, if calculated, will help us determine the probability that the data point will fall within each class. From there, you can help determine the class of that data point by selecting the class with the highest probability:

$$c = \arg \max_{c \in \{1,\ldots,C\}} p(c|\mathbf{x}) \tag{10}$$

Expression (7) is often difficult to compute directly. Instead, the Bayes rule is often used:

$$c = \arg\max_c p(c|\mathbf{x}) \tag{11}$$

$$= \arg\max_c \frac{p(\mathbf{x}|c)p(c)}{p(\mathbf{x})} \tag{12}$$

$$= \arg\max_c p(\mathbf{x}|c)p(c) \tag{13}$$

Continuing with the expression (13), *p(c)* can be interpreted as the probability that a point falls in class c. This value can be calculated by MLE, i.e. the ratio of the number of data points in the training set that fall in this class divided by the total amount of data in the training set; or can also be evaluated using MAP estimation. The first is more commonly used.

The remaining component *p(x|c),* i.e. the distribution of data points in class *c*, is often difficult to compute because *x* is a multidimensional random variable, requiring a lot of training data is built to

get that distribution. To facilitate computation, it is often simplest to assume that the components of the random variable *x* are independent of each other, if *c* is known. It means:

$$p(\mathbf{x}|c) = p(x_1, x_2, \ldots, x_d|c) = \prod_{i=1}^{d} p(x_i|c) \tag{14}$$

$$c = \arg \max_{c \in \{1,\ldots,C\}} = \log(p(c)) + \sum_{i=1}^{d} \log(p(x_i|c)) \tag{15}$$

Gaussian Naïve Bayes: For each data dimension *i* and a class *c*, x_i obeys an expected normal distribution of μ_{ci} and the variance of σ^2_{ci}:

$$p(x_i|c) = p(x_i|\mu_{ci}, \sigma^2_{ci}) = \frac{1}{\sqrt{2\pi\sigma^2_{ci}}} \exp\left(-\frac{(x_i - \mu_{ci})^2}{2\sigma^2_{ci}}\right) \tag{16}$$

In which, the parameter set $\vartheta = \{\mu_{ci}, \sigma^2_{ci}\}$ is determined by Maximum Likelihood:

$$(\mu_{ci}, \sigma^2_{ci}) = \arg \max_{\mu_{ci}, \sigma^2_{ci}} \prod_{n=1}^{N} p(x_i^{(n)}|\mu_{ci}, \sigma^2_{ci}) \tag{17}$$

2.5. Support Vector Machine

Support Vector Machines (SVMs) are supervised learning models, which can be applied for classification analysis and regression analysis. They have been proposed by Vapnik in 1995. They can perform both linear and non-linear classification tasks [5], [12], [17], [19].

Consider the duality problem in Soft Margin SVM for nearly linearly differentiated data:

$$\begin{aligned} \lambda &= \arg \max_{\lambda} \sum_{n=1}^{N} \lambda_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \mathbf{x}_n^T \mathbf{x}_m \\ \text{subject to:} \quad & \sum_{n=1}^{N} \lambda_n y_n = 0 \\ & 0 \le \lambda_n \le C, \ \forall n = 1, 2, \ldots, N \end{aligned} \tag{18}$$

Inside:

N: number of data point pairs in training set.

x_n: feature vector of nth data in training set.

y_n: label of nth data, equal to 1 or -1.

λ_n: Lagrange factor corresponding to n data point.

C: a positive constant helps to balance the magnitude of the margin and the sacrifice of points in the insecure region. When $C = \infty$ or very large, Soft Margin SVM becomes Hard Margin SVM.

After solving λ for problem (18), the label of a new data point will be identified by the sign of the expression:

$$\sum_{m \in \mathcal{S}} \lambda_m y_m \mathbf{x}_m^T \mathbf{x} + \frac{1}{N_{\mathcal{M}}} \sum_{n \in \mathcal{M}} \left(y_n - \sum_{m \in \mathcal{S}} \lambda_m y_m \mathbf{x}_m^T \mathbf{x}_n\right) \tag{19}$$

Inside:

$M = \{n: 0 < \lambda_n < C\}$ is a set of points on margin.

$S = \{n: 0 < \lambda_n\}$ is a set of support points.

N_M is the number of elements in *M*

Suppose we can find a function Φ() such that, after being converted to a new space, each data point x becomes Φ(x), and in this new space, the data becomes close to linear difference. Now, hopefully the solution of the Soft Margin SVM problem will give us a better classifier.

In the new space, the problem (18) becomes:

$$\begin{aligned} \lambda = \arg\max_{\lambda} & \sum_{n=1}^{N} \lambda_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \Phi(\mathbf{x}_n)^T \Phi(\mathbf{x}_m) \\ \text{subject to:} \quad & \sum_{n=1}^{N} \lambda_n y_n = 0 \\ & 0 \leq \lambda_n \leq C, \ \forall n = 1, 2, \ldots, N \end{aligned} \tag{20}$$

And the label of a new data point is identified by the sign of the expression:

$$\mathbf{w}^T \Phi(\mathbf{x}) + b = \sum_{m \in \mathcal{S}} \lambda_m y_m \Phi(\mathbf{x}_m)^T \Phi(\mathbf{x}) + \frac{1}{N_{\mathcal{M}}} \sum_{n \in \mathcal{M}} \left(y_n - \sum_{m \in \mathcal{S}} \lambda_m y_m \Phi(\mathbf{x}_m)^T \Phi(\mathbf{x}_n) \right) \tag{21}$$

In problem (20) and expression (21), we do not need to calculate Φ(x) directly for all data points. We simply need to compute $\Phi(x)^T\Phi(z)$ based on any two data points x, z. This technique is also known as a kernel trick. Methods based on this technique, that is, instead of directly calculating the coordinates of a point in a new space, we compute the scalar product between two points in the new space, collectively called the kernel method.

Now, by defining the kernel function $k(x,z) = \Phi(x)^T\Phi(z)$, we can rewrite problem (20) and expression (21) as follows:

$$\begin{aligned} \lambda = \arg\max_{\lambda} & \sum_{n=1}^{N} \lambda_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m k(\mathbf{x}_n, \mathbf{x}_m) \\ \text{subject to:} \quad & \sum_{n=1}^{N} \lambda_n y_n = 0 \\ & 0 \leq \lambda_n \leq C, \ \forall n = 1, 2, \ldots, N \end{aligned} \tag{22}$$

$$\sum_{m \in \mathcal{S}} \lambda_m y_m k(\mathbf{x}_m, \mathbf{x}) + \frac{1}{N_{\mathcal{M}}} \sum_{n \in \mathcal{M}} \left(y_n - \sum_{m \in \mathcal{S}} \lambda_m y_m k(\mathbf{x}_m, \mathbf{x}_n) \right) \tag{23}$$

Below is a summary of common kernels and their usage:

Table 3 Common Kernels

Name	Math
linear	x^Tz
Polynomial	$(r+\gamma x^Tz)^d$
Sigmoid	$\tanh(\gamma x^Tz+r)$
rbf	$\exp(-\gamma\|\|x - z\|\|^2_2)$

2.6. XGBoost (Extreme Gradient Boosting)

XGBoost (Extreme Gradient Boosting) is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. In prediction problems involving unstructured data (images, text, etc.) artificial neural networks tend to outperform all other algorithms or frameworks. However, when it comes to small-to-medium structured/tabular data, decision tree based algorithms are considered best-in-class right now.

Now, XGBoost is also one of the most commonly used implementations of boosted decision trees in the word. Now, it is available in Amazon SageMaker.

Among the machine learning methods used in experiment, gradient tree boosting [21] is one technique that shines in many applications. Gradient tree boosting is also known as gradient boosting machine or gradient boosted tree.

Gradient Boosting Decision Tree (GBDT) [22] is an immortal model in machine learning. In fact:

GBDT = Gradient Boosting + Decision Tree

For many years, [32] reported XGBoost algorithm that is good at dealing with high dimensional data is very consistent with the research in this paper. Based on cleaning for excessive number of variables and missing values of high dimensional data, therefore, we apply XGBoost algorithm in detail in Algorithm 1 which is using multi-observation data cleaning has high accuracy in prediction. It has also been considered a recent phenomenon of excellence in various cases in which the concept originated from the construction of additive models [33].

Algorithm 1 XGBoost algorithm

1: Data: Dataset and Hyperparameters

2: Initialize $f_0(x)$;

3: For i= 1,2, …, M do

Calculate $g_i = \frac{\partial L(y,f)}{\partial f}$;

Calculate $h_i = \frac{\partial^2 L(y,f)}{\partial f^2}$;

Determine the structure by choosing splits with maximized gain

$$Gain = \frac{1}{2}\left[\frac{G_L^2}{H_L} + \frac{G_R^2}{H_R} - \frac{G_L^2}{H}\right];$$

Determine the leaf weights $w^* = -\frac{G}{H}$;

3. Experiments and Results

In this comparative study, the selected models included one output/target with two values (True or False) regarding hospital readmission during a period of 30 days. In other words, the value of the readmission parameter is true if readmission is done during a period of 30 days. Otherwise, in case of no readmission or in case readmission is done after 30 days, its value if false. The set of drivers for the prediction was comprised of the selected features as discussed above. The training dataset and the testing dataset were selected randomly. Additionally, 10-fold cross validation was applied by selecting 40% of the data for testing and the rest for training. The settings of the various models are discussed below.

$$\text{Accuracy} = (TP+TN)/(TP+FP+FN+TN) \quad (24)$$

$$\text{Recall} = TP/(TP+FN) \quad (25)$$

$$\text{Precision} = TP/(TP+FP) \quad (26)$$

$$F1 = 2^* (\text{Recall}^*\text{Precision}) / (\text{Recall} + \text{Precision}) \quad (27)$$

Accuracy indicates how often the classifier is correct. The recall is a sensitivity measure (ratio of TPs to the sum of TPs and FNs). It indicates the rate of cases the model predicted the patient will be readmitted in a month period (relative to the number of cases the patient was actually readmitted). The precision measures the rate of cases that the model predicts the patient will be readmitted in a month period correctly compared to total number of cases in which the model predicts the patients will be readmitted. Table 4 depicts the values of the performance measures.

Table 4 Results of model evaluation

	Model	Accuracy	Recall	Precision	F1
1	Logistic Regression	0.655274	0.621895	0.680601	0.625131
2	Artificial Neural Network	0.8163147	0.7658391	0.827283	0.727114
3	Naïve Bayesian Classifier	0.6148718	0.5558375	0.6371715	0.5811649
4	Support Vector Machine	0.9226865	0.9178281	0.9245068	0.8998967
5	XGBoost	0.9638311	0.9245236	0.9851764	0.9046265

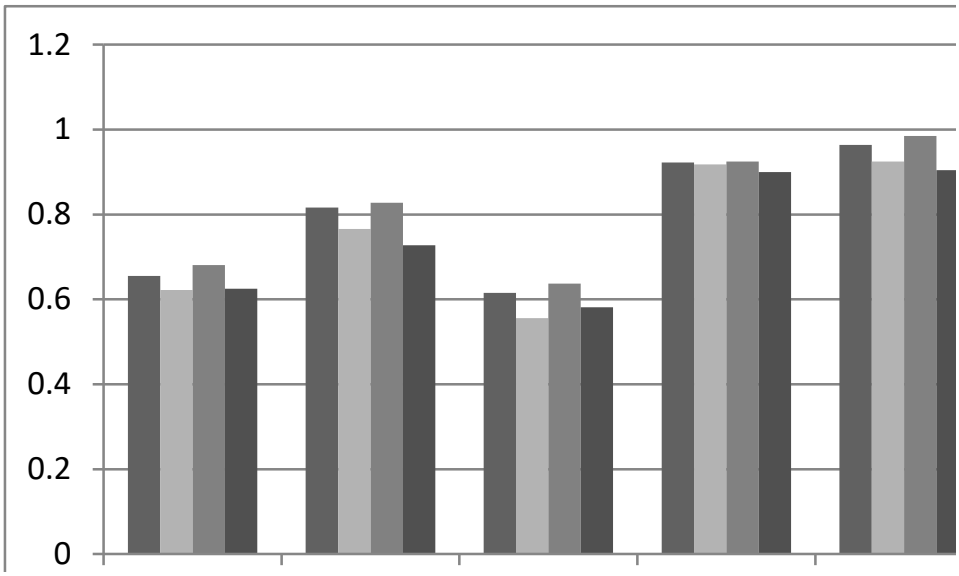


Figure 4 The results of model evaluation

In both Table 4 and Figure 4 are a summary of results showing the comparison of measured values to evaluate the predictive performance of hospital readmission of diabetic patients by the models. It also shows that the XGBoost model gives the highest results with accuracy of 0.96, recall measure of 0.92, precision measure of 0.98 and F1 measure of 0.9. This is better than SVM with accuracy of 0.92, recall measure of 0.91, precision measure of 0.92 and F1 measure of 0.89. Therefore, the proposed XGBoost model implemented in Amazon SageMaker.

4. Conclusion and Future work

According to the above experiments, this work uses five computational intelligence techniques for diabetes mellitus prediction namely Logistic Regression, Artificial Neural Network, Naïve Bayes, Support Vector Machine and XGBoost. The performance of these techniques was evaluated on various performance measurements. We found the results of using the all features, Tuning Hyper-parameter

and using the proposed XGBoost model through a proposed methodology in Figure 1 have better results. The result, which gives the best predictive results compared to other models by the ratio of metrics (Accuracy, Recall, F1 and AUC score) is almost 0.9 or higher than the ratio of metrics of other models in the range of 0.6 to 0.9. Besides, XGBoost model is implemented in Amazon's SageMaker machine learning platform is slightly higher in the Train Accuracy and the Test Accuracy because the Tuning Hyper-parameter focuses intensively on minimizing the Validation Error. It is better for avoiding 'overfitting' and applying in real-life problems. SageMaker configures tuning job quickly thanks to its Tuning Hyper-parameter built-in function. Moreover, it also provides Python SDK to manually develop by developers.

So, readmissions increased costs of health care and negatively affect the reputation of the hospital. Therefore, it is important to predict the number of hospitalizations for diabetics. This article presents Machine/Deep learning as an effective approach to predict hospitalization in diabetics.

In future work, in the case of XGBoost model used in the other method whose has model is trained successfully, it can be copied to deploy to any machine to use if this machine is set up Python environment and installed XGBoost package. However, on the large scale, for example, there are thousands of machine using this model, it will be very struggle when the model is updated and modified. In this case, the only option is to set up a server and creates an API to synchronize all the machines. Also, for the XGBoost model implemented in Amazon SageMaker, after there is a trained model, it can be deployed as an endpoint API which can be use easily by many machines and systems. So, we can do it quickly and easily without much effort in the future.

In addition, we will come up with methods of prevention of complications and several other relevant risk factors of Type 2 Diabetes (T2D) [39] to reduce the reduction of underlying diseases for global citizens and to raise awareness of better health education and to limit investment costs of medical equipment to the global economic recovery in general [52]. In addition, the social, economic, and financial impacts around the world are enormous, especially with many stock exchanges plummeting due to the spread of *Coronavirus* [53]. Therefore, we analyze the important of Cloud computing service and the obstacles to Cloud computing services development [40], [41], [42] in Vietnam. It is easily to recognize that, there are others important problem, such as, the development of network infrastructure, improve the education, training and research campaign donations in developing countries. Through this development, the application of modern information technology is one of the most important issues in restoring economies in developing countries with the most modern and asynchronous information technology infrastructure. The goal of this economic recovery is likely to identify the key trends of marketing integration and innovation, which will help business organizations including industrial enterprises, commercial firms, public companies, transportation companies and especially travel companies, taking into account the characteristics of the modern world, moving towards a digital society in order to develop sustainable innovation and innovation of business organizations [54], as well as identify key areas of marketing integration and innovation in the future.

References

1. Roglic, G. (2016). Global report on diabetes. *World Health Organization*, 58(12), 1-88.
2. Rubin, D., Donnell-Jackson, K., Jhingan, R., Golden, S. & Paranjape, A. (2014). Early readmission among patients with diabetes: A qualitative assessment of contributing factors. *J. Diabetes Complications*, 28(6), 869-873.
3. Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I. & Chouvarda, I. (2017). Machine Learning and Data Mining Methods in Diabetes Research. *Comput. Struct. Biotechnol. J.*, 15, 104-116.
4. Chowriappa, P., Dua, S. & Todorov, Y. (2014). Machine Learning in Healthcare Informatics. *Berlin: Spinger*, 56, 1-23.
5. Mitchell, T. (1997). Machine learning (mcgraw-hill international editions computer science series).

6. Bose, E. & Radhakrishnan, K. (2018). Using Unsupervised Machine Learning to Identify Subgroups among Home Health Patients with Heart Failure Using Telehealth. *CIN - Comput. Informatics Nurs.*, 36 (5), 242-248.
7. Kaelbling, L., Littman, A. & Moore, A. (1996). Reinforcement learning: A survey. *J. Artif. Intell. Res.*, 4, 237-285.
8. Shailaja, K., Seetharamulu, B., & Jabbar, M. A. (2018). Machine Learning in Healthcare: A Review. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, *IEEE*, 910-914.
9. Davies, J. & Gibbons, J. (2012). Machine Learning and Software Engineering in Health Informatics. *In Proceedings of the First International Workshop on Realizing AI Synergies in Software Engineering*, 37-41.
10. Bhardwaj, R., Nambiar, A. & Dutta, D. (2017). A Study of Machine Learning in Healthcare. *Proc. - Int. Comput. Softw. Appl. Conf.*, 2, 236- 241.
11. Asuncion, A. & Newman, D. (2007). 'UCI Machine Learning Repository [Online]'. Available: *https://archive.ics.uci.edu/ml/index.php*.
12. Strack. B. et al. (2014). Impact of HbA1c measurement on hospital readmission rates: Analysis of 70,000 clinical database patient records. B*iomed Res. Int.*
13. Karp, A. H. (1998). Using logistic regression to predict customer retention. *Proc. Elev. Northeast SAS Users Gr. Conf.* Available: *https://www. lexjansen. com/nesug/nesug98/solu/p095.pdf*.
14. Amato, F., López, A., Peña-Méndez, E. M., Vaňhara, P., Hampl, A. & Havel, J. (2013). Artificial neural networks in medical diagnosis. *J. Appl. Biomed.*, 11(2), 47-58.
15. Sabbeh, S. F. (2018). Machine-learning techniques for customer retention: A comparative study. *International Journal of Advanced Computer Science and Applications*, *9*(2).
16. Jothi, N., Rashid, N. & Husain, W. (2015). Data Mining in Healthcare - A Review. *Procedia Comput. Sci*., 72, 306-313.
17. Sisodia, D. and Sisodia, D. (2018). Prediction of Diabetes using Classification Algorithms. *Procedia Comput. Sci.*, 132, 1578-1585.
18. Hazra, A., Kumar, S. & Gupta, A. (2016). Study and Analysis of Breast Cancer Cell Detection using Naïve Bayes, SVM and Ensemble Algorithms. *Int. J. Comput. Appl.*, 145(2), 39-45.
19. Holzschuh, E. (1992). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Reports Prog. Phys.*, 55(7), 1035-1091.
20. Sharma, R., Sugumaran, V., Kumar, H. & Amarnath, M. (2015). A comparative study of naive Bayes classifier and Bayes net classifier for fault diagnosis of roller bearing using sound signal. *Int. J. Decis. Support Syst.*, 1(1), 115.
21. Hazra, A., Mandal, S. K., Gupta, A., Mukherjee, A. & Mukherjee, A. (2017). Heart disease diagnosis and prediction using machine learning and data mining techniques: a review. *Advances in Computational Sciences and Technology*, *10*(7), 2137-2159.
22. Nai-Arun, N. & Sittidech, P. (2014). Ensemble Learning Model for Diabetes Classification. *In Adv. Mater. Res.*, 931, 1427-1431.
23. Perveen, S., Shahbaz, M., Guergachi, A. & Keshavjee, K. (2016). Performance Analysis of Data Mining Classification Techniques to Predict Diabetes. *Procedia Comput. Sci.*, 82(3), 115-121.
24. Orabi, K. M., Kamal, Y. M., & Rabah, T. M. (2016). Early predictive system for diabetes mellitus disease. *In Industrial Conference on Data Mining*, 420-427.
25. Paul, D. (2018). Analysing Feature Importances for Diabetes Prediction using Machine Learning Debadri. *In 2018 IEEE 9th Annu. Inf. Technol. Electron. Mob. Commun. Conf.*, 924-928.
26. Kerexeta, J., Artetxe, A., Escolar, V., Lozano, A. & Larburu, N. (2018). Predicting 30-day Readmission in Heart Failure using Machine Learning Techniques. *In HEALTHINF*, 308-315.

27. Sabbeh, S. F. (2018). Machine-learning techniques for customer retention: A comparative study. *International Journal of Advanced Computer Science and Applications*, *9*(2).

28. Kira, K. & Rendell, L. A. (1992). A practical approach to feature selection. *In Machine Learning Proceedings 1992, Morgan Kaufmann*, 249-256.

29. Opitz, D. W. (1999). Feature selection for ensembles. *Proc. 16th Natl. Conf. Artif. Intell. AAAI*, 16(3), 379–384.

30. Guyon, I. & Elisseeff, A. (2003). An Introduction to Variable and Feature Selection 1 Introduction. *An Introd. to Var. Featur. Sel*., 3, 1157–1182.

31. Sokolova M. & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing and Managemet*, 45, 427-437.

32. Ma, X., et al. (2018). Study on a prediction of P2P network loan default based on the machine learning LightGBM and XGboost algorithms according to different high dimensional data cleaning. *Electronic Commerce Research and Applications*, 31, 24-39.

33. Choi, D. K. (2019). Data-Driven Materials Modeling with XGBoost Algorithm and Statistical Inference Analysis for Prediction of Fatigue Strength of Steels. *International Journal of Precision Engineering and Manufacturing*, 20(1), 129-138.

34. Hersh, W. R. (2007). Adding value to the electronic health record through secondary use of data for quality assurance, research, and surveillance, *Clin Pharmacol Ther*, 81.

35. Swapna, G., Vinayakumar, R. & Soman, K. (2018). Diabetes detection using deep learning algorithms*. ICT Express*, 4(4), 243-246.

36. Ramírez, J. C. & Herrera, D. (2019). Prediction of diabetic patient readmission using machine learning. *IEEE Colombian Conference on Applications in Computational Intelligence*.

37. Nguyen, B. P. et al. (2019). Predicting the onset of type 2 diabetes using wide and deep learning with electronic health records. *Computer methods and programs in biomedicine*, 182.

38. Jovanovič, L. (2019). The quest to conquer maternal hyperglycemia-a personal tryst. *The journal of maternal-fetal & neonatal medicine: the official journal of the European Association of Perinatal Medicine, the Federation of Asia and Oceania Perinatal Societies, the International Society of Perinatal Obstetricians*, *32*(23).

39. Chinh, N. V. et al. (2020). Prevention of complication and several other relevant risk factors of Type 2 Diabetes (T2D) at a hospital in Vietnam. *European Journal of Molecular & Clinical Medicine*, 7(10), 266-279.

40. Le Dinh Phu Cuong & Dong Wang et al. (2020). Advanced Cloud Computing Services: The evaluation of a development roadmap for emerging fields in Vietnam. *Journal of Critical Reviews*, 7(17), 3085-3105.

41. Cuong, L. D. P. & Wang Dong et al. (2020). Breast Cancer prediction based on Deep Neural Network model implemented AWS Machine Learning Platform. *International Journal of Recent Technology and Engineering*, 9(2), 868-873.

42. Hoang, D. T. et al. (2020). Weather prediction based on LSTM model implemented AWS Machine Learning Platform. *International Journal for Research in Applied Science & Engineering Technology*, 8(5), 283-290.

43. Peters, S. A. & Woodward, M. (2018). Sex differences in the burden and complications of diabetes. *Current diabetes reports*, 18(6), 1-8.

44. Federation, I. D. (2017). IDF diabetes atlas 8th edition. *International Diabetes Federation*, 905-911.

45. Maital, S. & Barzani, E. (2020). The global economic impact of COVID-19: A summary of research. *Samuel Neaman Institute for National Policy Research*, *2020*, 1-12.

46. Tunio, M. N., Shaikh, E. & Lighari, S. (2021). Multifaceted perils of the Covid-19 and implications: A Review. *Studies of Applied Economics*, *39*(1).

47. Vijayan, V. V. & Anjali, C. (2015). Prediction and diagnosis of diabetes mellitus - A machine learning approach. In *2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*, 122-127.

48. Zou, Q., Qu, K., Luo, Y., Yin, D., Ju, Y. & Tang, H. (2018). Predicting diabetes mellitus with machine learning techniques. *Frontiers in genetics*, *9*.

49. Dwivedi, A. K. (2018). Analysis of computational intelligence techniques for diabetes mellitus prediction. *Neural Computing and Applications*, *30*(12), 3837-3845.

50. Vigneswari, D., Kumar, N. K., Raj, V. G., Gugan, A. & Vikash, S. R. (2019). Machine learning tree classifiers in predicting diabetes mellitus. In *2019 5th international conference on advanced computing & communication systems (ICACCS),* 84-87.

51. Daanouni, O., Cherradi, B. & Tmiri, A. (2019). Type 2 diabetes mellitus prediction model based on machine learning approach. In *The Proceedings of the Third International Conference on Smart City Applications,* 454-469.

52. Furman, J., Geithner, T., Hubbard, G. & Kearney, M. S. (2020). Promoting Economic Recovery After COVID-19. *Washington: Economic Strategy Group, The Aspen Institute*, 16.

53. Filipe, J. A. (2020). Covid-19, Chaos theory and the "drop of honey effect". Viruses and human behavior. *Estudios de Economia Aplicada*, *38*(3).

54. Popova, N., Kataiev, A., Nevertii, A., Kryvoruchko, O. & Skrynkovskyi, R. (2020). Marketing aspects of innovative development of business organizations in the sphere of production, trade, transport, and logistics in VUCA conditions. *Studies of Applied Economics*, *38*(3 (1)).