

Automatic missing value imputation for cleaning phase of diabetic's readmission prediction model

Jesmeen Mohd Zebaral Hoque¹, Jakir Hossen¹, Shohel Sayeed², Chy. Mohammed Tawsif K.¹,
Jaya Ganesan³, J. Emerson Raja¹

¹Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia

²Faculty of Information Science and Technology, Multimedia University, Melaka, Malaysia

³Alliance School of Business, Alliance University, Bangalore, Karnataka, India

Article Info

Article history:

Received Dec 30, 2020

Revised Aug 3, 2021

Accepted Sep 1, 2021

Keywords:

Classification algorithms

Cleaning

Data analytics

Data preprocessing

Feature selection

ABSTRACT

Recently, the industry of healthcare started generating a large volume of datasets. If hospitals can employ the data, they could easily predict the outcomes and provide better treatments at early stages with low cost. Here, data analytics (DA) was used to make correct decisions through proper analysis and prediction. However, inappropriate data may lead to flawed analysis and thus yield unacceptable conclusions. Hence, transforming the improper data from the entire data set into useful data is essential. Machine learning (ML) technique was used to overcome the issues due to incomplete data. A new architecture, automatic missing value imputation (AMVI) was developed to predict missing values in the dataset, including data sampling and feature selection. Four prediction models (i.e., logistic regression, support vector machine (SVM), AdaBoost, and random forest algorithms) were selected from the well-known classification. The complete AMVI architecture performance was evaluated using a structured data set obtained from the UCI repository. Accuracy of around 90% was achieved. It was also confirmed from cross-validation that the trained ML model is suitable and not over-fitted. This trained model is developed based on the dataset, which is not dependent on a specific environment. It will train and obtain the outperformed model depending on the data available.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Jesmeen Mohd Zebaral Hoque, Jakir Hossen
Faculty of Engineering and Technology, Multimedia University
Melaka, Malaysia
Email: jesmeen.online@gmail.com, jakir.hossen@mmu.edu.my

1. INTRODUCTION

Data analytics (DA) is a new technology used to make correct decisions through proper analysis and prediction. The DA model will help to reduce cost of health care and enhance patient care process. However, the main concern is to have clean data to get valuable and accurate outcomes. There could be a chance for risk if data quality (DQ) is low, as it will lead to incorrect or unwanted decisions and actions. Consequently, this risk may affect the company's data processing time and cost over billions of dollars every year [1]. Data issues are becoming more problematic, as around 60% of organizations face critical issues from bad DQ, and every individual organization may contain 10-30% of inaccurate data in their databases. As stated by [2] "DQ is generally described as the capability of data to satisfy stated and implied needs when used under specified conditions". Low-level DQ can cause inaccurate or missing data. It may lead to incorrect or misleading decisions, predictions, or instruction. In 2010, Dey and Kumar [1] stated that dirty data could slow down any processing depending on DA and even affect the organization's total cost; the cost can be over billions of dollars

per year. Around 60% of the data in an organization contains data issues. Hence organizations are now worried about those dirty data. When it comes to the medical environment, these dirty data may kill patients or take towards the patient's long-lasting health issue. An organization of medicine [3] reported in 1999 that approximately 44,000 to 98,000 patients died every year due to errors present in medical records. It also costs more than 17 to 29 billion dollars annually. Other than wasting money and health, these insufficient data can affect the patient's privacy. However, it is still challenging to handle this type of incomplete data in healthcare system [4].

Some researches [5]–[8] focused on one specific issue of DQ, i.e., duplicate identification and elimination. It is only one aspect of data issue, hence the complete process of cleaning data had little attention in the research community. Before training final DA process to obtain meaningful outcome, it is important to concern almost all the dirty data issues, especially the common dimensions: Incomplete data, Inaccurate data, Duplicate data and Inconsistent data [9]. There are other dirty data dimensions found by different organization's dataset [10], [11], i.e., incomplete, ambiguous, inconsistent, and inaccurate data. It is essential to clean at least the common four dimensions, hence there was scope of obtaining enhance process of cleaning data. To make this process automatic a system is introduced using machine learning (ML) model [12] had modified the random forest (RF) algorithm to handle missing data; similarly [13] had developed a model to address the support vector machine (SVM). However, in this paper, ML is used to predict the missing value instead. Many of them are currently using Hadoop file system (HDFS) to reduce the data storing and retrieving cost [14], [15]. In this study, sampling method is introduced to make sure the model works for different data size after obtaining data from HDFS. Sampling helps to train model with data sample until ML model is trained to get it maximum performance. This ML model will then use to predict the missing values for the proposed framework automatic missing value imputation (AMVI). Before training the model, the system will also automatically select the important features. The sampling technique, feature selection methods, and automatically ML model selection will help to blend with different domain without involving human.

Contributions. This research focuses on improving the performance of DA by considering data errors that cause domain value violations in the context of supervised classification models. The system is developed using python language. The proposed system's main contribution will make the cleaning phase automatic by using appropriate predictive methods for different domains. The model is trained in such a way it will be able to predict the missing class. It may also reduce the high computational cost required to process the massive amount of data in ML during the cleaning phase of DA by introducing the proposed system. The cost is reduced by allowing only the sampled data to be used for training. Sampling is achieved by the divide-and-conquer technique in this proposed system. Moreover, human error could be avoided by introducing the proposed approach in life-critical applications such as healthcare management.

2. IMPLEMENTATION OF DATA ANALYTICS

Due to changes in food habits in this modern time, humans are facing different health issues. With the increase in health issues, hospital readmission has resulted in unaffordable, and it is essential to prevent this by taking the required measurements [16]. Here, if a patient is frequently admitted into the hospital in a short duration (say, within 30 days), he/she is considered to be readmitted. Readmission may happen due to different reasons, such as improper medication, patient diagnosis, follow-up surgery, transferring to another and hospital. In this use case, the essential purpose is to reduce the risk of readmission. To complete this intention, it is necessary to build a system providing an accurate tool for prediction analyzing patterns for hospital readmission. For developing this model, it was challenging to deal with data issues. Depending on hospital data, a DA model is designed and developed to predict according to the patient condition will readmit within 30 days to the hospital. This use case is selected to test whether the proposed cleaning phase in the system enhances the prediction or not. The DA process was implemented by steps presented in Figure 1.

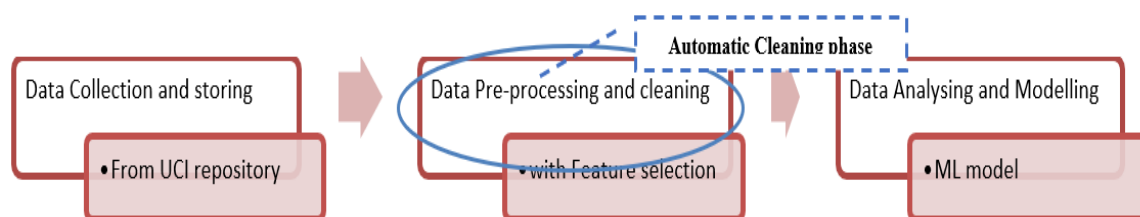


Figure 1. DA process overviews

2.1. Data collection and storing

Collecting data from a reliable source is very important to make sure data in real-world data. Typically, collected data contains data issues. It is essential to examine the dataset to obtain accurate results. For this research, a dataset was selected from the UCI repository [17]. The selected data set is initially stored in Hadoop file system (HDFS) in files to keep and read data with a lower computational cost. Then, data is retrieved from the file using 'PANDAS' library. 'InsecureClient' function in 'hdfs' library is used to connect (write and read) with HDFS. For testing purpose Hadoop is installed in the client desktop. 'hdfs' library act as a bridge between Hadoop and Python language used for scripting.

2.2. Data preprocessing

Data combined from heterogeneity sources can contain data inconsistencies and missing values. Few steps of data preprocessing and cleaning is done as follows:

- Dropping features that contain a high percentage (i.e. >50%) of data missing values
- Dropping features that do not help this process is completed by feature selection
- Unique IDs or unique values in every row of a feature is not useful for classification. Hence these columns also dropped (such as *patient_nbr* and *encounter_id*)
- Substitute missing values
- Adding a class label is required to set to indicate that the patient is readmitted or not. Here value 0 means readmitted within 30 days, and 1 means not readmitted within 30 days.
- Encode categorical values into numerical values, as selected ML can train using numerical values.

A new architecture is developed and able to input any structured dataset. Using the data, the ML models are trained for predicting missing values and then impute missing value in the dataset, known as AMVI, is presented in the following section. The automatic cleaning phase also can encode categorical values into numerical values, which is very important to training ML algorithms. Not only missing values, it also solves inconsistent data and duplicate data. In the case of massive data, to develop the model, a faster sampling technique is involved. The model is trained using sample data until the model prediction accuracy reaches a stable level.

2.3. Data analysis and modelling

Predictive modelling has been executed using fully scripted Python language. Once an ML model is trained for prediction purposes, in this stage, the accurate classification predictive algorithm is selected for further use for predicting readmission status. The classification algorithm in the DA stage applied for the experiment is logistic regression (LR), decision tree-based method (CART), and RF classifier. Finally, to evaluate dataset is classified into train and test data set into 70% and 30% respectively and followed by model effectiveness using model accuracy and receiver operating characteristic (ROC) Curve.

3. DESIGN OF AUTOMATIC CLEANING PHASE

When humans enter data into a system, then it is usual to get incomplete and inconsistent data. Even data from sensors can provide data error due to failure. However, analysis processed by ML cannot accept these data errors. The dataset was selected to make sure it has missing and inconsistent values. If the dataset has only a missing value (means the value is unknown or '?'), it is inconsistent (such as changing date format). Therefore, to avoid manual suggestions, a system was developed to overcome missing values and inconsistent data issues in this research. Hence, a method is designed aiming at auto-cleaning for data analytics that will improve DQ. The developed cleaning phase is presented in Figure 2. The complete framework contains list of function $F = \{f_1, f_n\}$ to solve dirty data dimensions. The cleaner function does the following: i) selects the best solving missing values from predictive (AMVI) or drops or mode (categorical variables) or median (continuous values) or set a unique value, ii) encodes non-numerical variables, iii) reformats any date-time inconsistencies, and iv) removes duplicates

The easiest and most popular solution to the missing data problem (i.e., f_{NAN}) is removing all the rows with any missing data. Unfortunately, deleting rows with missing data could add a significant bias to the data set. If all the rows that have been deleted are for males, that data is biased against males. Another example is when different columns have missing values in other rows. By removing all the rows with missing values, one could potentially remove a significant portion of the usable data, reducing accuracy. The second standard treatment for missing value is to add a constant. That approach only works in some cases and is best made by a person with a strong background in the problem the data represents. The third common approach is to replace the missing values with a median, mean or mode. This approach works if the data is well-balanced/normalized. It is also best processed by someone with a good understanding of the information they are dealing with. One exciting and straightforward way to enhance missing data using the mean is to calculate the mean using a specific condition. However, the approach is not dependent on its features. The solution is using ML

techniques. First, the collection of missing values needs to be separated from the actual set. Next, by using another dataset, the ML model must train and finally predict the values missing. Here, the ML model learns from the data patterns available between the features and output (by using rows that do not contain any missing data) and uses those patterns to predict the missing class of all affected rows (details described in the following section).

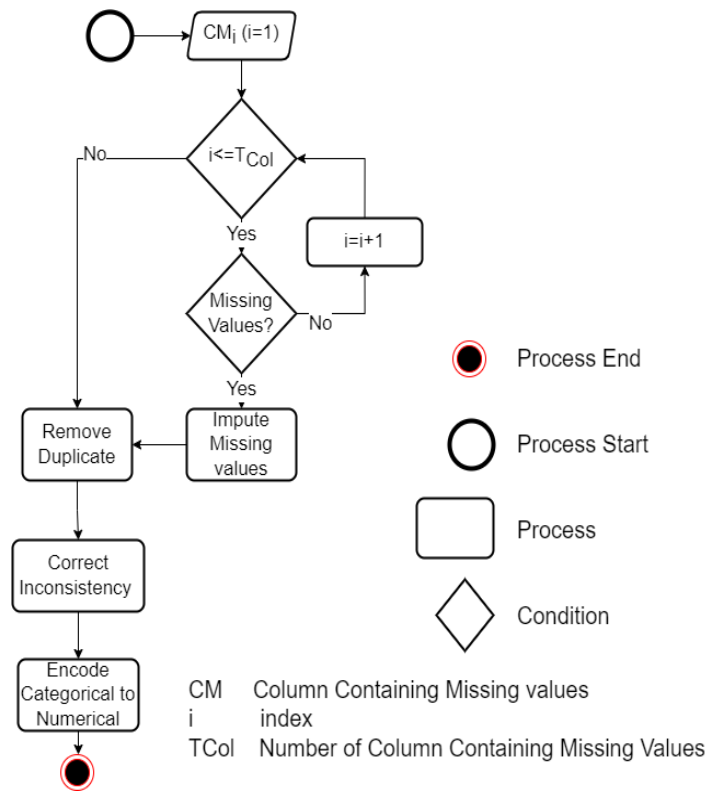


Figure 2. Data cleaning approach in da phase cleaning missing data

3.1. Design of automatic missing value imputation (AMVI) approach

The few challenges faced after implementing DA process are stated [18]. An automatic cleaning phase was developed to overcome a few of the challenges (i.e., volume and incompleteness). The auto-cleaning will allow the users to train predictive models while progressively cleaning data and preserve convergences guarantees. The proposed architecture AMVI for the auto-cleaning tool is shown in Figure 3. The algorithm is presented in the following section.

- Data sample. The ML models will be trained on the previously formed subset. In this way, data analysis algorithms performance with minimum computing and stock resources can be maintained and improved. Every sample was tested until a stable ML accuracy was obtained to find the minimum consistent subset. For data sampling, the divide-and-conquer strategy is implemented, which has a good effect in practical application.
- Removing rows with issues. For training purposes, it is essential to separate the rows containing missing values. These rows may cause the problem to get a better predictive model.
- Splitting data. Considering, the dataset contains clean data. This data set is spat into the training and testing data set. The percentage is usually around 80/20 or 70/30. Here, 70/30 selected to split data for training purposes. It is implemented using the Scikit-learn library and precisely the *train_test_split* method.
- Feature selection. Firstly, before selecting features, it is essential to find out which column contains issues. Considering that column ‘C’ to be class/output and other columns are features. The best parts are then selected using Gini values obtain from the trained random forest model. It is an important step, as it affects the outcome of the final trained model [19]
- Train/Test model. The selected models were trained and tested using a training and testing data set where features were obtained from the feature selection method.

- Evaluation method. By using the evaluation technique (presented in Section VII), the best model is selected. Different datasets will perform differently due to data structure. Finally, the best model is stored and used for future prediction.
- Prediction phase. The prediction phase can be integrated into the DA system's cleaning phase, identifying and imputing the missing class.

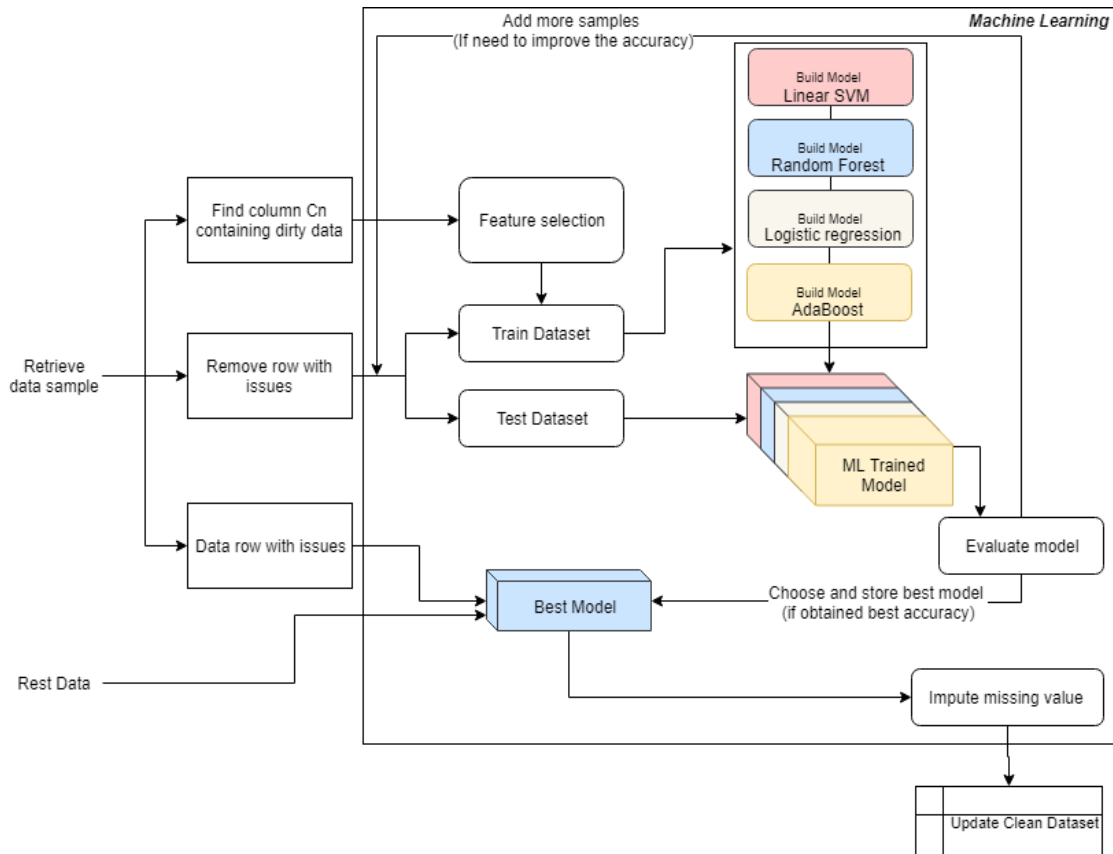


Figure 3. AMVI auto-cleaning architecture

3.2. Implemented algorithm for AMVI

Here, four best trained ML model training approach is integrated to detect numerical and categorical data. The missing value column is considered as class and rest data are features. Initially, a clean dataset (D_{clean}) is used to insert into the system. If the dataset is not cleaned, it will be cleaned manually. Next, consider D_{NAN} to be set of containing data with missing value and can be solved using f_{NAN} function, as mentioned earlier. The system developed for f_{NAN} function is as:

- Step 1. Let C_{null} be the missing value columns and considered missing class, Detect C_{null} containing missing values using F_{NAN} (Function to detect is column contains null/NaN/?)
- Step 2. Obtaining, D_{clean} (the sample dataset cleaned previously) and let C_{clean} is the list of features for each record containing clean data
- Step 3. Split D_{clean} into $D_{\text{clean_train}}$ (Train input data) and $D_{\text{clean_test}}$ (Test input data), and split C_{clean} into $C_{\text{clean_train}}$ (Train output data) and $C_{\text{clean_test}}$ (Test output data)
- Step 4. Train selected model (models represented in Section VI) with selected data set and labels
Train ($D_{\text{clean_train}}$, C_{clean})
if obtained best features
go to step 6
else
go to next step
- Step 5. Important feature selection using Gini index value obtained from RF. Gini index is defined as in (1). Using the algorithm presented in section VI.

$$gini(C) = \sum_{i=1}^{n_i} p_i(1 - p_i) \quad (1)$$

where n_i is the number of classes in set C (the target variable) and p_i refers ratio of this class i .

Step 6. Repeat step 3

Step 7. Test the trained classifier by using test data to obtain accuracy

$$P_{\text{test}} = \text{Predict}(D_{\text{clean_test}})$$

$$\text{accuracy_score}(C_{\text{clean_test}}, P_{\text{test}})$$

Step 8. Repeat the process from step 3 until Four selected model accuracy is obtained

Step 9. Store the best-trained ML model

Step 10. Applying the best-trained ML model used to predict the missing class for the set of rows D_{NAN}

$$P = \text{Predict}(D_{\text{NAN}})$$

Step 11. Predicted P data are appended to the D_{clean} data, Goto step 2.

$$D_{\text{clean}} = D_{\text{clean}} \cup P$$

The script was implemented using Python Language. Moreover, for data retrieval and presentation using pandas and Matplotlib library was integrated.

3.3. Feature selection using Gini index

Once the system has all the features, it is crucial to select the essential features for training and testing the ML model. All features are not helpful to train the ML model for accurate prediction. Therefore, in this stage, the RF Gini index values can be used for feature selection. The outcome of feature importance values is plotted in the graph shown in the result and discussion section, containing the calculation of the best features selected for training ML algorithms. Here, in Figure 4 (step 5), the threshold value 0.95 was chosen by testing the system with different threshold values. The best features will be selected for each column containing the missing value.

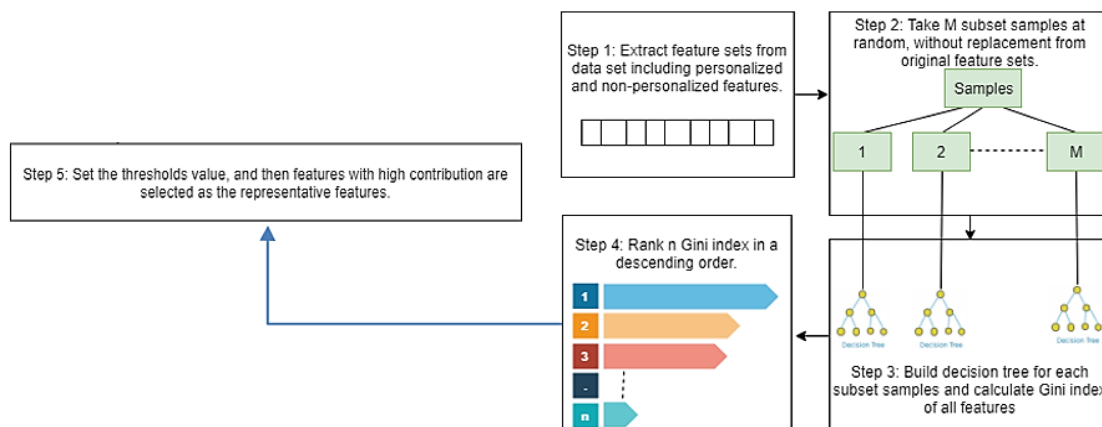


Figure 4. Basic steps of feature selection

4. TRAINING ML CLASSIFIERS

The supervised ML models are known approach to train the model using labelled dataset. Four selected ML techniques were used for training the model, i.e., RF, Linear SVM, Adaboost, and LR. These are selected initially by comparing eight well-known technologies (i.e., LR, linear regression, Linear SVM, AdaBoost, K-nearest neighbour, SGDClassifier, gradient boosting, and RF). The selection is processed by training and testing the model and comparing its accuracy.

4.1. Random forest algorithm

RF contains more than one decision trees. RF is one of the excellent choices; it is tolerable in data noise and trains with reasonable accuracy. The decision trees are independent of each other [20]. The tree rules can be generated using two techniques. The techniques are i) Sampling indiscriminately by negating loop replacement and ii) broadly selecting the best subset of features from a complete set of features for the splitting node. The splitting node process helps to acquire the optimal set of features at each node, depending on its

impurity. These impurities of parent and children's nodes are calculated by using the Gini index. Consider the classes C for samples from set l ; then the Gini index values are calculated using (2).

$$gini(C) = \sum_{i=1}^{n_l} p_i(1 - p_i) \quad (2)$$

where n_l are the number of classes in set C (the target variable), and p_i refers ratio of this class i .

Therefore, for two classes with a quantity of data split into N_1 and N_2 for each class, the Gini index for C is calculated using (3).

$$gini_{split}(C) = \frac{N_1}{N} Gini(C_1) + \frac{N_2}{N} Gini(C_2) \quad (3)$$

The smallest split Gini (C) is selected to split the node, as it has lower impurities. The Gini index value of one class node will be 0. Feature importance can be classified using these Gini Index. The outcome is selected from checking the decision trees results. The final classifier gathers the majority votes for class C , and then provides the final prediction.

4.2. Support vector machine (SVM) algorithm

The second method selected is SVM, due to its capability of single and multiclass classification. The main task to train SVM algorithm is to find a hyperplane in an N -dimensional space, where N is the number of features/fields. The line will discriminate different data points in the group. The output is obtained in the linear function. To get an accurate trained model, it is important to obtain the maximum distance between data points and the hyperplane. The cost function calculated by using (4) to maximize the margin,

$$J(\theta) = \sum_{i=1}^n y^{(i)} Cost_1(\theta^T x^{(i)}) + (1 - y^{(i)}) Cost_0(\theta^T x^{(i)}) \quad (4)$$

$$\text{where } Cost(h_\theta(x), y) = \begin{cases} \max(0, 1 - \theta^T x) & \text{if } y = 1 \\ \max(0, 1 + \theta^T x) & \text{if } y = 0 \end{cases}$$

where n is the total number of data. $h_\theta(x)$ is the SVM hypothesis from the raw model output $\theta^T x$. Here y is the predicted value. If $\theta^T x \geq 0$, then $y=1$, else $y=0$. For further regulation, (7) is calculated and added with (5).

$$reg = \frac{1}{2} \sum_{j=1}^m \theta_j^2 \quad (5)$$

Here, m is the total number of features/fields used to train the SVM model

4.3. Logistic regression algorithm

One of the most common classification ML algorithms is LR. It is one of the probabilistic classification models. Here, LR contains a sigmoidal curve, which helps plot the data patterns to get the label as output. The sigmoid function graph is plotted using (6):

$$S(x) = \frac{1}{1 + e^{-z}} \quad (6)$$

Equation (7) is required to cast the problem to obtain the LR model in the generalized form structure.

$$\hat{y} = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n \quad (7)$$

where predicted value is indicated by \hat{y} , is independent variables indicated by x and the β are coefficients to be trained. Finally, (10) is compacted to vector form. Thus, the logistic link function can be used to cast LR into the Generalized Linear Model. To able to work with Multiclass classification with LR, the one-vs-rest (OvR) scheme was used. In OvR scheme for every class, a binary classification process is executed whether the data contains belonging or not. The loss function also does it to cross-entropy loss.

4.4. Adaboost classifier

AdaBoost, a short form of Adaptive Boosting, is an ML meta-algorithm that is the first practical boosting algorithm proposed by [21]. It helps to convert weak classifiers into the string. For final classification is presented in (8).

$$F(x) = \text{sign}(\sum_{m=1}^M \theta_m f_m(x)) \quad (8)$$

where f_m contains a value for m^{th} weak classifier and θ_m holds corresponding weight. The equation includes a summation of a combination of weight and M weak classifiers. The process of the AdaBoost algorithm is as:

For n points of the inputted dataset, and $x_i \in \mathbb{R}^d, y_i \in \{-1,1\}$. negative class is indicated by -1, and a positive class is characterized by 1. The weight for every data point is initialized by using (9).

$$w(x_i, y_i) = \frac{1}{n}, i = 1, \dots, n \tag{9}$$

For every iteration $m=1, \dots, M$: Find the classifier and fit it to the data set. Concerning distribution, select one of the lowest weighted classification errors by using (10).

$$\epsilon_m = E_{w_m} [1_{y \neq f(x)}] \tag{10}$$

Here, E is exponential loss function for weight w of m^{th} classifier. If y is not equal to $f(x)$ it is considered as misclassification (i.e., 1)

Prerequisite <0.5 ; otherwise stop

Calculate the weight for the m^{th} weak classifier by using (11).

$$\theta_m = \frac{1}{2} \ln \left(\frac{1 - \epsilon_m}{\epsilon_m} \right) \tag{11}$$

The weight is considered positive; if the classifier accuracy is greater than 50%, the weight is deemed to be negative. The combination of prediction is processed by flipping the sign. By using (12), the weight was updated for all points.

$$w_{m+1}(x_i, y_i) = \frac{w_m(x_i, y_i) \exp[-\theta_m y_i f_m(x_i)]}{Z_m} \tag{12}$$

To ensure the total weight instances are equal to 1, Z_m is a normalization factor. Here, the term ‘exp’ in the numerator would be greater than 1 ($y_i * f_i$ is always -1, the θ_m is positive) if any misclassified case occurs from a positive weighted. Hence, after an iteration, the misclassified values will be updated with larger weights. Once, iteration is completed, the predictive weight of all classifiers is summarized for the final prediction.

5. SYSTEM EVALUATION TECHNIQUES

Accuracy, confusion matrix, and cross-validation techniques are used to evaluate the performance of the prediction model. Classification accuracy method used for evaluation is by retrieving predictive outcomes. By using confusion matrix, the terms true positive (TP), true negative (TN), false negative (FN), and false positive (FP) can be presented as shown in Table 1. A confusion matrix clarifies that these terms are about actual and classified outcomes given by a trained ML model.

Table 1. Confusion matrix of a classifier [20]

	Classified Positive Outcome	Classified Negative Outcome
Actual Positive Outcome	The amount of readmitted patients that are correctly identified as readmitted TP	The amount of readmitted patients that are incorrectly identified as not readmitted FN
Actual Negative Outcome	The amount of not readmitted patients that are incorrectly identified as readmitted FP	The amount of not readmitted patients that are correctly identified as not readmitted TN

The essential measurements can be calculated once TP, TN, FN, and FP values are obtained. These accuracy measurements (The probability of correct classification) are obtained using (13) for evaluating the trained model.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \tag{16}$$

Next, the K-Folds cross-validation technique is also used and confirmed that the trained model does not have unreliable issue. Here, the complete dataset was broken into k section (where, $k=5$). In this K-Folds Cross-Validation, the sample data was split into k different subsets/folds. At first, fold-2 to fold-5 is used for training

purposes, leaving the fold-1 for testing purposes. Next, fold-1, fold-3, fold-4, and fold-5 were used for training and fold-2 used to test the model. This process goes on until every fold is used for testing purposes. Finally, the average accuracy (Cross-validation Score) is compared, and the final model was selected. This testing helps overcome the over-fitting model issue.

6. RESULT AND DISCUSSION

6.1. Feature importance

The first set of data used to find its feature importance is diabetics data [18]. This data set is used to predict whether a patient is getting readmitted or not. The features to be selected to train AMVI to predict the missing values of a column labeled as 'rosiglitazone' are shown in Figure 5. Best features are obtained based on cumulative values of features' importance. This technique also helps to reduce fluctuation [22]. It is evident from the graph that features are ranked from most important to least necessary according to their Gini values. They are *diag_1*, *medical_specialty*, *DiabetesMed*, *metformin*, *age*, and so on.

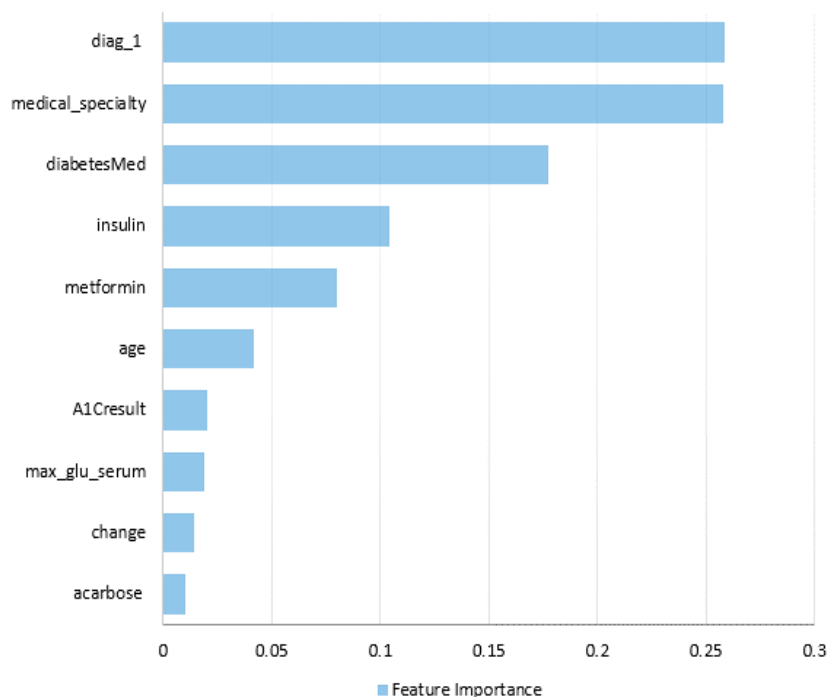


Figure 5. Calculated feature importance using Gini index

6.2. Performance AMVI architecture with selected classifiers

Classifier's performance depends on the type of dataset selected for training. The inclusion of four different ML algorithms in the AMVI gives the ability to choose the best ML algorithm according to the type of dataset presented. The results obtained after training and testing AMVI with the four different datasets separately are shown in Figure 6.

This accuracy refers to the amount of correctly predicted missing values for each column; for example, in the graph in Figure 6(a), prediction accuracy values plotted for 'rosiglitazone' column are presented. The results obtained while training and testing RF in AMVI with Diabetics Data are shown in Figure 6(a), where the trained RF Algorithm recorded more than 90% accuracy. The Trained LinearSVM model and Adaboost model results are shown in Figures 6(b) and 6(c), respectively. It indicates that they are unstable and low in prediction accuracy. LR trained algorithm recorded more than 85% accuracy Figure 6(d).

6.3. Cross-validation analysis for AMVI

Cross-validation testing is done to measure the performance of AMVI by giving training and testing to ML models of AMVI. The entire dataset was divided into 5-folds; and four of them were used for training, and the remaining one was used for testing. It is repeated for all the folds, and total accuracy is calculated to get a Cross-validation score. The results obtained are shown in Table 2 along with classification accuracy

obtained for predicting the missing value(s) of the column rosiglitazone. It is proved from the table that the trained model is not over-fitted because of the closeness between model accuracy and cross-validation accuracy. It also indicates that the model is more dependable.

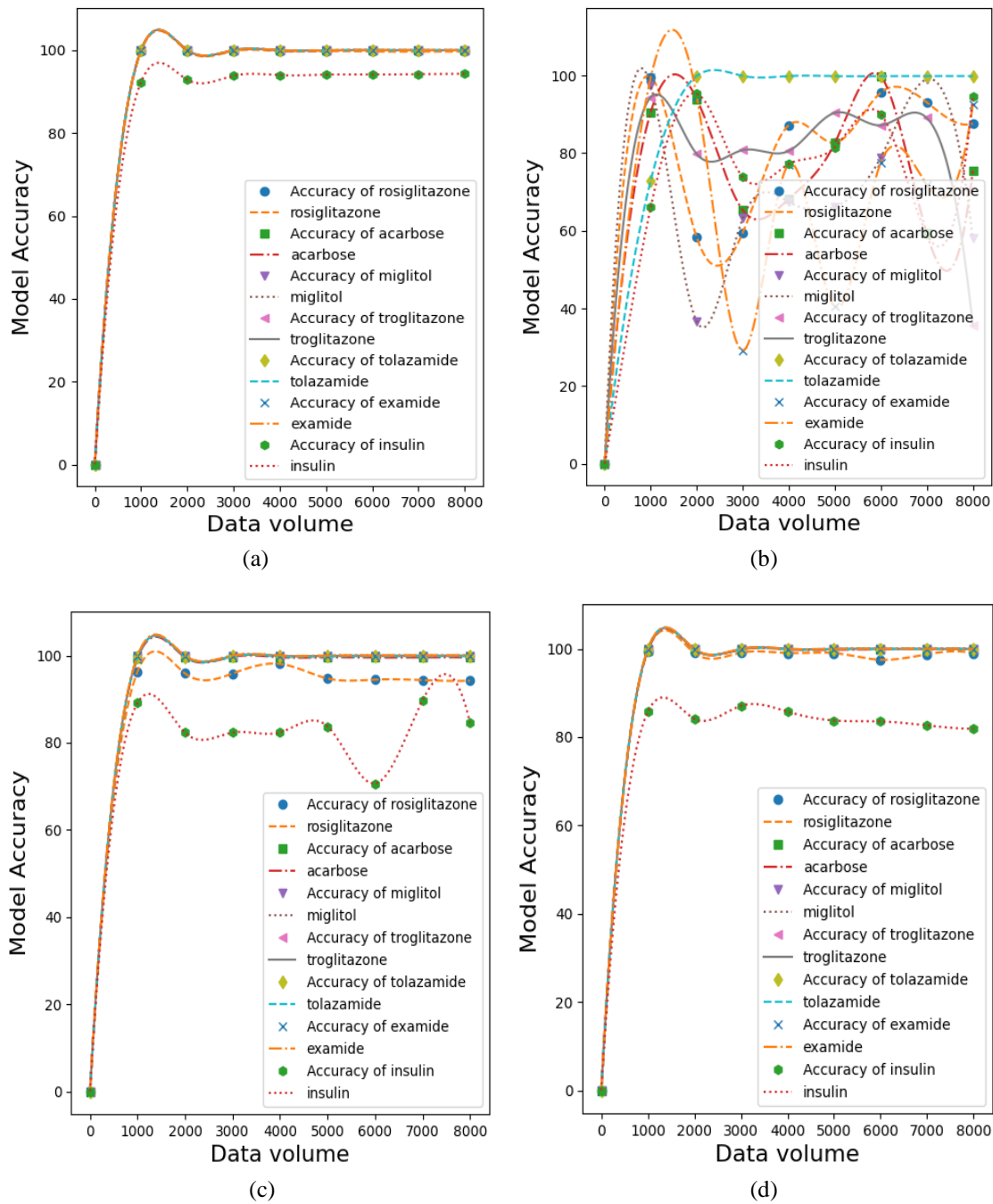


Figure 6. Accuracy percentage vs data volume for trained ML: (a) RF, (b) Linear SVM, (c) Adaboost, and (d) LR

Table 2. Cross-validation comparison for ‘rosiglitazone’

Number of data row	Accuracy	Cross-Validation percentage accuracy
20000	88.10%	86.082%
40000	92.65%	88.233%
60000	90.05	87.285%
80000	90.84	87.043%

6.4. Comparing with existing work

The implemented DA's performance is compared with the results obtained from different published papers used the same dataset. The obtained outcome from these papers is presented in Table 3 for comparison with developed AMVI framework. In Table 3, the column titled 'Paper' was used to refer to the research paper from which the result was obtained. The column titled 'results' presents the results obtained from their papers. The last column represents the information used to compare with existing work and the proposed system. However, the techniques used are different from the methods used in this research, but the dataset and use case is similar. It clearly shows that the final DA outcome for the proposed system performs better than the existing ones. The main reason for the difference is the cleaning phase. An enhanced and automatic cleaning phase tends to improve the DA process.

Table 3. Comparison with existing readmission prediction research

Paper	Accuracy	Compared with AMVI
[23]	Accuracy using SVM model 82.70%	Final DA prediction ~91% accuracy for RF, SVM and LR
[24]	Overall AUC: 0.56 using Tree Classifiers [0-30): Precision 0.3651 Accuracy 84.81% [30-70): Precision 0.2288, Accuracy 78.5% [30-70): Precision 0.1857, Accuracy 68.49%	The developed system for overall data set is AUC:0.64 using the tree classifier (Random Forest)
[25]	the accuracy obtained by the model is 63.38%	Used SVM technique. However, ignored the cleaning phase

In paper [23], they had predicted using SVM and found less accuracy then the model developed in this paper. For further comparison [26] is compared by finding AUC value. Here AUC value is Area under curve of receiver operating characteristic plotted graph, i.e., true positive rate (TPR) vs false positive rate (FPR). TPR is rate of output positive correctly identified, by calculating $TP/(TP+FN)$. Whereas FPR is rate of wrongly identifying positive output by using equation $FP/(FP+TN)$.

7. CONCLUSION AND FUTURE WORK

An approach was designed and developed for the automatic cleaning phase to enhance DA' overall performance. How to extract knowledge is one of the most frequently discussed issues. Considering an important use case from healthcare, the comprehensive system was evaluated. This research was initiated to identify the critical phase in DA processing. It was derived with a conclusion that data cleaning is the most potential phase, which many researchers ignore. A ML technique is implemented in the cleaning phase of Data analytics to achieve the objective. ML is an advanced artificial intelligence technique used in areas where data needs to be analyzed for accurate prediction. It can learn from the data analyzed and determine or predict something about the researched subject. Even though data cleaning is possible by software implementation, it isn't easy to make it adaptive without ML ability. Software data cleaning techniques are usually based on rules applicable to a specific domain. Hence, any changes in the environment might fail the data cleaning process, affecting DA performance drastically. The data cleaning process with ML can withstand any domain changes because of learning from the data. Hence, the automatic cleaning of dirty data is possible with the ML-based data cleaning process. Firstly, if the cleaning process is automated, it is also expected to reduce human fault occurrences and improve the analysis outcomes with more meaningful visions. Secondly, the trained model will select features and get the best-trained model automatically according to the dataset. Moreover, other researcher had already proved the importance of feature selection. Accuracy and cross-validation techniques were used for evaluating the cleaning stage. Around 90% accuracy is achieved for missing values prediction in the cleaning phase alone. While comparing Adaboost Classifier, linear SVM, LR, and random forest with the same data set, it was found that Adaboost Classifier and linear SVM performance far better than the other two. The reason may be due to the non-linearity in the decision boundaries and the complexity in the time-sequence dependent interactions. It was also confirmed from the results of cross-validation that the trained model is not over-fitted. Even though the proposed system had been tested in all essential aspects and proved useful in performance after comparing with existing techniques, there could be some space for further improvement. It could be achieved by introducing an auto-tuning facility to the parameters of the ML model. The proposed system is designed to work with only structured data containing data issues. Hence, to make it suitable for unstructured data, an appropriate module could be added in the future for converting the unstructured data to structured data, if required. Future enhancement is also necessary to this approach to have the ability to auto-detect and correct the inaccurate data, if any, stored in the data set. Furthermore, the method can be evaluated with other common method of imputing missing values, such as mean, median and unique value.

ACKNOWLEDGEMENTS




Funding: This work was supported in part by the Telekom Malaysia under Grant TMRND.

REFERENCES




- [1] D. Dey and S. Kumar, "Reassessing data quality for information products," *Management science*, vol. 56, no. 12, pp. 2316–2322, Dec. 2010, doi: 10.1287/mnsc.1100.1261.
- [2] F. Sidi, P. H. S. Panahy, L. S. Affendey, M. A. Jabar, H. Ibrahim, and A. Mustapha, "Data quality: A survey of data quality dimensions," *Int. Conference on Information Retrieval & Knowledge Management*, Mar. 2012, doi: 10.1109/infrkm.2012.6204995.
- [3] "To err is human: building a safer health system," Institute Of Medicine, 1999. Accessed May 14, 2020 [Online]. Available: <https://www.nap.edu/resource/9728/To-Err-is-Human-1999--report-brief.pdf>
- [4] M. Adibuzzaman, P. DeLaurentis, J. Hill, and B. D. Benneyworth, "Big data in healthcare-the promises, challenges and opportunities from a research perspective: A case study with a model database," *AMIA Annual Symposium Proceedings*, vol. 2017, pp. 384–392, Apr. 2018.
- [5] Y. Chen, W. He, Y. Hua, and W. Wang, "CompoundEyes: Near-duplicate detection in large scale online video systems in the cloud," in *Proceedings - IEEE INFOCOM*, Apr. 2016, vol. 2016-July, doi: 10.1109/INFOCOM.2016.7524429.
- [6] J. M. Dupare and N. U. Sambhe, "A novel data cleaning algorithm using RFID and WSN integration," *International Conference on Innovations in Information, Embedded and Communication Systems (ICIECS)*, Mar. 2015, doi: 10.1109/ICIECS.2015.7193201.
- [7] B. A. Hoverstad, A. Tidemann, and H. Langseth, "Effects of data cleansing on load prediction algorithms," *IEEE Computational Intelligence Applications in Smart Grid (CIASG)*, Apr. 2013, doi: 10.1109/CIASG.2013.6611504.
- [8] W. S. Ku, H. Chen, H. Wang, and M. Te Sun, "A Bayesian inference-based framework for RFID data cleansing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2177–2191, Oct. 2013, doi: 10.1109/TKDE.2012.116.
- [9] J. Archana and E. A. M. Anita, "A survey of big data analytics in healthcare and government," *Procedia Comput. Sci.*, vol. 50, pp. 408–413, 2015, doi: 10.1016/j.procs.2015.04.021.
- [10] J. Han, K. Chen, and J. Wang, "Web article quality ranking based on web community knowledge," *Computing*, vol. 97, no. 5, pp. 509–537, Nov. 2014, doi: 10.1007/s00607-014-0435-4.
- [11] J. R. C. Nurse, S. S. Rahman, S. Creese, M. Goldsmith, and K. Lamberts, "Information quality and trustworthiness: A topical state-of-the-art review," *International Conference on Computer Applications and Network Security (ICCANS 2011)*, 2011.
- [12] J. Xia *et al.*, "Adjusted weight voting algorithm for random forests in handling missing values," *Pattern Recognit.*, vol. 69, pp. 52–60, Sep. 2017, doi: 10.1016/j.patcog.2017.04.005.
- [13] R. K. Nowicki, K. Grzanek, and Y. Hayashi, "Rough support vector machine for classification with interval and incomplete data," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 10, no. 1, pp. 47–56, Dec. 2019, doi: 10.2478/jaiscr-2020-0004.
- [14] A. L. and D. V. Kumar, "Amodel to predict and pre-treat diabetes mellitus," in *International Conference on Advances in computer Science and Technology (IC-ACT'18) - 2018*, 2018, pp. 1–6.
- [15] R. Shantha, R. S. Joshitta, and L. Arockiam, "A predictive model to forecast and pre-treat diabetes mellitus using clinical big data in cloud," *International Journal of Applied Engineering Research*, vol. 10, pp. 55–59, 2015.
- [16] M. K. N and R. Manjula, "Role of big data analytics in rural health care - a step towards svasth bharath," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 6, pp. 7172–7178, 2014.
- [17] J. Donzé, D. Aujesky, D. Williams, and J. L. Schnipper, "Potentially avoidable 30-day hospital readmissions in medical patients," vol. 173, no. 8, Apr. 2013, Art. no. 632, doi: 10.1001/jamainternmed.2013.3023.
- [18] UCI, "Diabetes 130-US hospitals for years 1999-2008 Data Set," *Clinical and Translational Research, Virginia Commonwealth University*, 2014.
- [19] C. L. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: a survey on big data," *Information sciences*, vol. 275, pp. 314–347, Aug. 2014, doi: 10.1016/j.ins.2014.01.015.
- [20] C. Liu, C. Tsai, K. Sue, and M. Huang, "The feature selection effect on missing value imputation of medical datasets," *applied sciences*, vol. 10, no. 2344, pp. 1–12, 2021.
- [21] L. Breiman, "Random forests," *Machine Learning*, 2001.
- [22] Y. Freund, R. E. Schapire, and M. Hill, "Experiments with a new boosting algorithm yoav," in *Machine Learning: Proceedings of the Thirteenth International Conference, 1996.*, 1996, pp. 148–156, doi: 10.1.1.133.1040.
- [23] L. Duan, F. Zhao, J. Wang, N. Wang, and J. Zhang, "An integrated cumulative transformation and feature fusion approach for bearing degradation prognostics," *Shock and Vibration*, vol. 2018, pp. 1–15, 2018, doi: 10.1155/2018/9067184.
- [24] S. Cui, D. Wang, Y. Wang, P.-W. Yu, and Y. Jin, "An improved support vector machine-based diabetic readmission prediction," *Computer methods and programs in biomedicine*, vol. 166, pp. 123–135, Nov. 2018, doi: 10.1016/j.cmpb.2018.10.012.
- [25] D. Mingle, "Predicting diabetic readmission rates: moving beyond Hba1c," *Current Trends in Biomedical Engineering & Biosciences*, vol. 7, no. 3, Aug. 2017, doi: 10.19080/ctbeb.2017.07.555715.
- [26] H. Munnangi and G. Chakraborty, "Predicting readmission of diabetic patients using the high-performance support vector machine algorithm of SAS @ Enterprise Miner™," in *SAS Global Forum, Dallas, TX 2015*, 2015, pp. 1–10.

BIOGRAPHIES OF AUTHORS






Jesmeen Mohd Zebaral Hoque    is currently a PhD student in engineering and specializing in artificial intelligence from Multimedia University, Malaysia. She completed Master's in engineering (Multimedia University, Malaysia) and a bachelor's degree in computer science and engineering (International Islamic University Chittagong, Bangladesh). Her research interest is artificial intelligence, big data, smart homes, and Machine learning. She can be contacted at email: jesmeen.online@gmail.com.






Jakir Hossen    is graduated in Mechanical Engineering from the Dhaka University of Engineering and Technology (1997), Masters in Communication and Network Engineering from Universiti Putra Malaysia (2003) and PhD in Smart Technology and Robotic Engineering from Universiti Putra Malaysia (2012). He is currently a Senior Lecturer at the Faculty of Engineering and Technology, Multimedia University, Malaysia. His research interests are in the area of Artificial Intelligence (Fuzzy Logic, Neural Network), Inference Systems, Pattern Classification, Mobile Robot Navigation and Intelligent Control. He can be contacted at email: jakir.hossen@mmu.edu.my.






Shohel Sayeed    obtained the B. Sc. Ag. (Hons) from Bangladesh Agricultural University. He completed his M.Sc. (IT) from Universiti Kebangsaan Malaysia (UKM) and Ph.D. in Engineering from Multimedia University, Malaysia. At present, he is holding a position of Associate Professor at the Faculty Information Science and Technology, Multimedia University, Malaysia. His main research interests are Biometrics, Pattern Recognition, Signal and Image Processing, Big Data and Data Mining. He can be contacted at email: shohel.sayeed@mmu.edu.my.






Chy. Mohammed Tawsif K.    is currently a PhD student in Engineering Program and is researching Artificial Intelligence, Event Processing, and Big data from Multimedia University (MMU). He pursued a Master's degree in Engineering from MMU; and bachelor's degree in Computer Science and Engineering from International Islamic University Chittagong, Bangladesh. He can be contacted at email: tawsif.online@gmail.com.



Jaya Ganesan    is currently working as Professor in Alliance School of Business at Alliance University, Bangalore, India. Dr. Jaya Ganesan is the current project leader for this grant funded by TM. As an active researcher she is a member of the Faculty 's Research and Innovation Committee and served as Coordinator for Post Graduate Programmes by Research for five years. Dr. Jaya Ganesan has 25 years of teaching, research and administrative experience in the higher education sector. Her areas of expertise include Human Resource Analytics, Green Human Resource Management, OB, IHRM. She is a reviewer and editorial member for internationally recognized journals. She can be contacted at email: jaya.ganesan@mmu.edu.my.



J. Emerson Raja    is currently senior lecturing in the Faculty of Engineering and Technology at Multimedia University, Malaysia. His work is centered on applying soft computing techniques to monitor the health of machines. He has been selected to give lectures in "International Teaching Week" at Hof University of applied sciences, Hof, GERMANY, in June 2013. He had given key note speech in several International Conferences in China and India. His research interested is Soft computing, tool condition monitoring topics and Artificial Intelligent. He can be contacted at email: emerson.raja@mmu.edu.my.