Hybrid model of Correlation based Filter Feature Selection and Machine Learning classifiers applied on Smart Meter Data set

SINAYOBYE Janvier Omar
Makerere University
School of Computing and Information Technology
Kampala, Uganda.
sijaom2@gmail.com

KIWANUKA N. Fred
Makerere University
School of Computing and Information Technology
Kampala, Uganda.
kiwanoah@gmail.com

KAAWAASE KYANDA Swaib
Makerere University
School of Computing and Information Technology
Kampala, Uganda.
kwaibk@gmail.com

MUSABE Richard
University of Rwanda
School of Information and Communication Technology
Kigali, Rwanda.
rmusabe10@gmail.com

***Abstract*— Feature selection is referred to the process of obtaining a subset from an original feature set according to certain feature selection criterion, which selects the relevant features of the dataset. It plays a role in compressing the data processing scale, where the redundant and irrelevant features are removed. Feature selection techniques show that more information is not always good in machine learning applications. Apply different algorithms for the data at hand and with baseline classification performance values we can select a final feature selection algorithm. In this paper, we propose a hybrid classification model, which has correlation based filter feature selection algorithm and Machine learning as classifiers. The objective of this study is to select relevant features and analyze the outperform machine learning algorithms in order to train our model, predict and compare their classification performance. In this method, features are ordered according to their Absolute correlation value with respect to the class attribute. Then top K Features are selected from ordered list of features to form a reduced dataset. This proposed classifier model is applied to our smart meter datasets. To measure the performance of these selected features; seven benchmark classifier are used; Random Forest (RF), Logistic Regression (LR), k-Nearest Neighbor (kNN), Naïve Bayes (NB), Decision Tree (DT), Linear Discriminant Analysis (LDA) and Support Vector Machine (SVM). This paper then analyzes the performance of all classifiers with feature selection in term of accuracy, sensitivity, F-Measure, Specificity, Precision, and MCC. From our experiment, we found that Random Forest classifier performed higher than other used classifiers.**

Keywords- Feature selection; Feature Extraction; smart meter data sets; machine learning.

I. INTRODUCTION

Smart meter data sets as Environment data sets in real-world are characterized by the large quantity of noise, redundant or irrelevant misleading features that may affect model classification measures. With these factors removed, learning from data mining and machine learning techniques can benefit greatly. A high dimensional dataset increases the risks that data mining algorithms find wrong patterns that are void in general. Most techniques involve some degree of reduction. This is necessary in order to manage with large amounts of data. Machine learning methods are very difficult to handle with the large number of high dimensional data found. Data pre-processing is a necessary step in the use of effective machine learning methods. Feature selection is an important technique used in data pre-processing. The main aim of feature selection is to determine the minimum number of feature subsets from a problem domain while retaining a high classification measures in representing the original features [1].

When the number of features selected is rather small, chances of information content may be low. On the other hand, the presence of noise as also irrelevant data will be highly probable when many features are selected. Hence, feature selection should be on the right selection of subsets, avoiding too large or too small number of features. There are many benefits of an ideal feature selection, such as data visualization, data understanding, reduction of the memory storage and training time, reduce the dimensionality which may improve prediction and classification performance. Thus, feature selection methods reduce the features present in the data set without changing them [2].

Nowadays many systems in a variety of fields deal with large data sets with high dimensionality. Feature selection, which has been a research topic in methodology and practice for decades, is used in many fields, such as image recognition [3, 4], image retrieval [5], text mining [6], intrusion detection [7, 8], bioinformatics data analysis [9,10, 11], fault diagnosis [12, 13], and so on.

According to the theoretical principle, feature selection methods can be based on statistics [14-17], information theory [18, 19], manifold [20], and rough set [21-24], and can be categorized according to various standards.

a) According to the utilized train (labeled, unlabeled, or partially labeled), feature selection methods can be divided into supervised, unsupervised, and semi-supervised models.

978-1-7281-2278-6/19/$31.00 ©2019 IEEE
DOI 10.1109/SEiA.2019.00009

b) According to their relationship with learning methods, feature selection methods can be classified into filter, wrapper, and embedded models.
c) According to the evaluation criterion, feature selection methods can be derived from correlation, Euclidean distance, consistency, dependence, and information measure.
d) According to the search strategies, feature selection methods can be divided into forward increase, backward deletion, random, and hybrid models.
e) According to the type of the output, feature selection methods can be divided into feature rank (weighting) and subset selection models.

The performance of the feature selection method is usually evaluated by the machine learning algorithms. The commonly used includes Naïve Bayes, KNN, C4.5, SVM, BP-NN, RBF-NN, K-means, Hierarchical clustering, Density based clustering and so on [25, 26, 27]. A good feature selection method should have high learning accuracy but less computational overhead (time complexity and space complexity). Although there have been solid reviews on feature selection [28-33], they mainly focus on specific research fields in feature selection.

Feature selection is a technique that has an ability to decrease the number of attribute by eliminating the least significant features [34]. However, the problem in feature selection is finding the optimum features. Most of the features in data sets that did not contribute to end result are unknown. Some unimportant or irrelevant features need to be diminished in order to reduce the classification complexity and time processing [35]. As feature selection becomes the important process in order to improve the classification performance, not all the feature selection techniques reduce the same feature in dataset. For that reason, choosing the feature selection techniques is crucial when subset feature is needed for dimensionality reduction and gives better performance in classification.

In this study we used hybrid model of Correlation-based Filter attribute Evaluation (CBF) as a conventional technique and machine learning algorithms commonly used, as one of the possible solutions to resolve the data mining problems such as feature selection and classification.
The aim of this study is to select relevant features and analyze the outperform machine learning algorithms in order to train our model, predict and compare their classification performance.

The main contribution of this study is to run the experiment on our smart meter dataset 1) to select the relevant features using Correlation-based attribute Evaluation (CB) as a conventional technique, 2) to analyze the outperform machine learning algorithms commonly used and evaluate the performance of selected features by using six performance measures accuracy, sensitivity, F-Measure, Specificity, Precision, and MCC.

II. RELATED RESEARCH

The problem of Dimensionality Reduction [35] can be decomposed into two steps: feature extraction and feature selection. Feature extraction is a preprocessing transformation over the feature space [36]. In turn, feature selection aims to select relevant and informative features, considering distinct criteria, such as enhancement of performance or classification effectiveness [37]. The premise is that, usually, datasets contain features that are irrelevant, redundant or noisy and, hence, may be removed without loss of useful information.

Indeed, many studies have showed that a proper feature selection may improve efficiency or even effectiveness of learning methods, since they simplify the resulting models and reduce chances of overfitting [38]. Thus, the goal is to filter out the maximum number of 'unnecessary' features from the input space. This filtering process determines feature space projections that represent subsets of features able to better describe the data, by defining a score for each feature in order to assess its discriminative power in the learning task.

Feature selection techniques can be divided into three groups: (1) filter methods, corresponding to strategies that select features without using a learning predictor; (2) wrapper methods, which use learning algorithms as "black boxes" to score a subset of features according to the classification effectiveness; and (3) embedding methods that adopt learning predictors to perform feature selection, but in this case, the selection process is injected into the training of a learning classifier. In [39], a comprehensive study is presented comparing different types of feature selection approaches. The authors show that filters are usually faster than wrappers, although the latter using a simple classification algorithm many authors show that filters are usually faster than wrappers, although the latter using a simple classification algorithm may be faster than the former.

A. Feature Selection measures and metrics

Feature selection is a process where features can automatically be selected in the data that contribute most to the prediction variable. Having irrelevant features in the data can decrease the performance measure of many models. Three benefits of performing feature selection before modeling the data are: Reduce overfitting, improve accuracy and reduce training time.

Feature selection has been an active and fruitful field of research area in pattern recognition, machine learning, statistics and data mining communities [40]. It is a dimensionally reduction technique whose main goal is to reduce irrelevant data and find features that increase classification performance measure. The main objective of feature selection is to choose a subset of input variables by eliminating features, which are irrelevant or of no predictive information. It has been proven in both theory and practice to be effective in enhancing learning efficiency, increasing predictive accuracy and reducing complexity of learned results [41, 42].

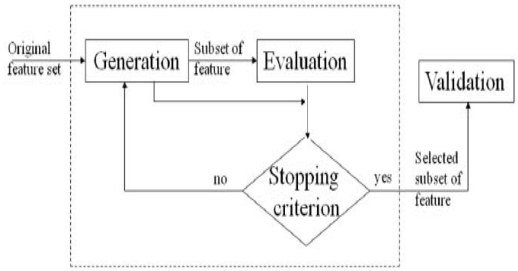

Figure.1: Feature selection process [43].

There are four basic steps in a typical feature selection process as shown in Fig. 1 [43]. The process of feature selection is as below;

- The generation procedure to generate the next candidate subset from original feature set.
- The evaluation function to evaluate the subset to determine the relevancy towards the classification task using measure for instances distance, dependency, information and consistency.
- Stopping criteria to decide when to stop. This is where it determines the relevant subset or optimal feature subset.
- Validation procedure is to check whether the selected feature subset is valid.

The feature measure or evaluation criterions play an important role in feature selection, which forms the basis of feature selection [44]. In the context of classification problems, optimal criterion would be the Bayesian error rate *E(S)* shown in formula (1) under the continuous or discrete condition, where $c_i \in C$ is a class from all the possible classes C exist in the data [45].

$$E(S) = \int_S p(S)\,(1 - max_i(p(c_i \mid S)))dS \quad (1)$$

Or $\sum_s p(S)\,(1 - max_i(p(c_i \mid S)))$

As can be seen from formula (1), $E(S)$ is in the form of sum, and $p(S)(1 - max_i(p(c_i \mid S)))$, Is non-linear and non-negative. It has an upper bound shown in formula (2), where $H(C \mid S)$ is the conditional entropy of C given S.

$$E(S) \leq H(C \mid S)/2 \quad (2)$$

It is hard to calculate $E(S)$ directly because S is the combination of features. Researchers prefer to use other measures, such as correlation, dependency and distance. Here, we show some general and representative evaluation measures as follows.

Let a and b be two features;

Correlation coefficient:

$$\gamma(a,b) = \frac{Cov(a,b)}{\sqrt{Var(a)}\sqrt{Var(b)}} \quad (3)$$

Where $Cov(a,b)$, is the covariance of a and b, and $Cov(.)$ is the variance.

Pearson correlation coefficient:

$$\gamma(a,b) = \frac{N\sum a_i b_i - \sum a_i \sum b_i}{\sqrt{N\sum a_i^2 - (\sum a_i)^2}\sqrt{N\sum b_i^2 - (\sum b_i)^2}} \quad (4)$$

Mutual information:

$$I(a;b) = \sum_a \sum_b p\ (ab)\log\frac{p(ab)}{p(a)p(b)} \quad (5)$$

Where $p(.)$ is the probability density function.

Symmetric uncertainty (SU):

$$SU(a;b) = \frac{2I(a;b)}{H(a)+H(b)} \quad (6)$$

Where $H(.)$, is the entropy of the feature.

Information distance:

$$d(a,b) = \frac{H(a|b)+H(b|a)}{2} \quad (7)$$

Where $H(a \mid b)$ is the conditional entropy of a given b

Euclidiean distance:

$$d(a,b) = \sqrt{\sum(a_i - b_i)} \quad (8)$$

These measures are often used in transformation for special applications. Moreover, there are other measures, such as Laplacian score, Fisher score, dependency index in rough set theory, and so on. Generally, information measures require the feature in discrete type, thus the feature discretization should be implemented before feature selection. Discretization methods include Equal-Depth, Equal-Width and manual setting, and etc. [46].

The relevance and applicability of feature selection metrics in practice have boosted the number of studies on this topic recently [47]. Thus, many distinct metrics have been proposed. Four distinct metrics (one and two-sided) widely used for this task are; Information Gain [48, 49] which quantifies how much information we obtain about a class when we know that a certain feature exists or not in a sample. Chi-square $\chi 2$ [50] used in statistical analysis to test whether two events are independent. In the context of feature selection, it is used to measure the association between features and classes. Odds-Ratio [51] measures the chances of a feature occur in the positive class normalized by that of the negative class and Correlation Coefficient [52, 53] used to estimate the correlation between classes and the interrelation among features.

B. Recent works

There exists several feature selection methods that are used by researchers. Some researchers trend to employ conventional method such as information gain and chi-square for instance [54, 55]. In the other researches, heuristic methods such as genetic algorithm [56] ACO [57] and [58] in memetic feature selection, noisy data, spam email, binary variables are used; respectively.

Feature selection also involves an active field of research such as in pattern recognition, machine learning and data mining area [59, 60]. Feature selection objective is to reduce irrelevant data and finding the most relevant features that would increase classification performance measures. It has been proven in both theory and practice to be effective in enhancing learning efficiency, increasing predictive accuracy and reducing complexity of learned results [61].

A wrapper feature selection approach based on BA and Optimum Path Forest had been proposed by Nakamura [62]. This approach modeled a problem of feature selection as a binary based optimization technique. Six datasets have been used in experiments that demonstrated that the proposed approach provides statistically significant more compact sets and in some cases it indeed improves the classification effectiveness.

Binary Bat Algorithm (BBA) was one of the inspired binary version feature selection that proposed to find the most significant feature in a search space [63]. BBA was proposed to associate each bat a set of binary coordinates that indicate whether that feature belongs to the final set of features or not. It combined the power of bat algorithm and Optimum Path Forest in finding the set of features that maximizes the accuracy of validating sets. It has been proved that the proposed techniques can outperform other well-known techniques such as PSO, FFA and GSA.

From [64] proposed bio-inspired method called Bat Algorithm hybridized with a Naive Bayes classifier (BANB). Twelve benchmarks datasets from different domains have been used in experiments to compare their performance measures with three well known feature selection techniques; GA, PSO and GPSO in term of the number of selected features from the original datasets. It shows that BANB significantly outperformed other algorithms in selecting significant number of features and lead to maintaining and improving classification accuracy.

As stated earlier, selection of a subset of features from existing set of features without a single extra effort (processing) is known as feature selection. Feature selection can be broadly done via filter, wrapper, and embedded approaches [65]. Minimum Redundancy Maximum Relevance, Fast Correlation-based Feature Selection (FCBF), and Correlation-based Feature Selection [65] are some filter methods used for feature selection. The Optimization algorithms are also employed for selection of good subsets.

Generally, these studies found that feature selection techniques are capable to improve the performance of learning algorithms thru increasing the accuracy of the classifier by removing irrelevant attributes. Therefore, with high quality features, it makes the classification process accurate, comprehensible and produces better results. For that reason, this research will conduct the experiment that focuses to analyze the outperform techniques among conventional and heuristic techniques.

C. Classification measures:

After doing the usual Feature Selection, and implementing a model and getting some output in forms of a probability or a class, the next step is to find out how effective is the model based on some metric using test datasets. Different performance metrics are used to evaluate different Machine Learning Algorithms for classification problems. We can use classification performance metrics such as Log-Loss, Accuracy, AUC (Area under Curve) etc. Another example of metric for evaluation of machine learning algorithms is precision, recall, etc. which can be used for sorting algorithms primarily used by search engines. The metrics that you choose to evaluate your machine learning model is very important. Choice of metrics influences how the performance of machine learning algorithms is measured and compared.

1) Confusion Matrix:

The Confusion matrix is one of the most intuitive and easiest metrics used for finding the correctness and accuracy of the model. It is used for Classification problem where the output can be of two or more types of classes. The confusion matrix is a table with two dimensions ("Actual" and "Predicted"), and sets of "classes" in both dimensions. The Actual classifications are columns and Predicted ones are Rows.

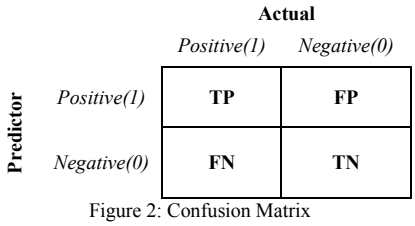

Figure 2: Confusion Matrix

The Confusion matrix in itself is not a performance measure as such, but almost all of the performance metrics are based on Confusion Matrix and the numbers inside it.

Let;

TP (True Positives); Denotes the number of positive patterns classified as positive. True positives are the cases when the actual class of the data point was 1 (True) and the predicted is also 1 (True).

TN (True Negatives); Denotes the number of negative patterns classified as negative. True negatives are the cases when the actual class of the data point was 0 (False) and the predicted is also 0 (False).

FP (False Positives); Denotes the number of negative patterns declared positive. False positives are the cases when the actual class of the data point was 0 (False) and the predicted is 1 (True). False is because the model has predicted incorrectly and positive because the class predicted was a positive one. (1)

FN (False Negatives); Denotes the number of positive patterns declared negative. False negatives are the cases when the actual class of the data point was 1(True) and the predicted is 0(False). False is because the model has predicted incorrectly and negative because the class predicted was a negative one. (0)

2) Performance measures and metrics:

Performance of classification model is measured using some techniques called classification measures. Classification accuracy is one of the measures to determine the performance of classification model. Classification accuracy is defined as the total percentage of correctly classified patterns. To find classification accuracy we can use the formula given by equation (9). Classification accuracy is mostly used but in many cases it is better to get

some other measures to define classification measure. Other measures are sensitivity, specificity, Precision, Recall. It is very necessary that along with accuracy these measures must be maximized. So it is very important that the model should maximize all these measures. Sensnasoitivity, Specificity, Precision and recall can be calculated using formula (9-14).

$$\text{Accuracy} = \frac{TP+TN}{TF+TN-FP+FN} \quad (9)$$

- Accuracy in classification problems is the number of correct predictions made by the model over all kinds predictions made. Accuracy is a good measure when the target variable classes in the data are nearly balanced. It should never be used as a measure when the target variable classes in the data are a majority of one class.

$$\text{Sensitivity/Recall} = \frac{TP}{TF+TN} \quad (10)$$

- Sensitivity also known as the True Positive rate or Recall is calculated as, Sensitivity = No. of True Positives / (No. of True Positives + No. of False Negatives)

$$\text{Specificity} = \frac{TN}{TN+FP} \quad (11)$$

- Specificity, also known as True Negative Rate is calculated as, Specificity = No. of True Negatives / (No. of True Negatives + No. of False Positives)

$$\text{Precision} = \frac{TP}{TP+FP} \quad (12)$$

- Precision is a measure that tells us what portion of cases that are True Positive. The predicted as positive (True positive (TP) and False Positive (FP)) and the actually True Positive (TP).

$$\text{F1-score / F measure} = \frac{TN}{TN+FN} \quad (13)$$

- F1 score incorporates both Recall and Precision and is calculated as, F1 score = 2 * (Precision * Recall) / (Precision + Recall)

$$\text{Matthews Correlation Coefficient (MCC)} = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)*(TP+FN)*(TN+FP)*(TN+FN)}} \quad (14)$$

- Unlike the other metrics discussed above, MCC takes all the cells of the Confusion Matrix into consideration in its formula. Similar to Correlation Coefficient, the range of values of MCC lie between -1 to +1. A model with a score of +1 is a perfect model and -1 is a poor model. This property is one of the key usefulness of MCC as it leads to easy interpretability

III. METHODOLOGY

This proposed method can be divided into two sections. In the first section we apply a filter criterion on each feature. Here we take Correlation with respect to class label as filtering criteria. In second section after selection of features, it is given to classifier for classification with either only selected feature of the dataset or with extended features.

Let D be a dataset having M number of patterns and N number of features. Each feature of the dataset will have M number of entries and dataset class will have also M number of entries corresponding to each entry of feature. Via correlation coefficient formula we can calculate the correlation coefficient between the feature of dataset and the class column of the dataset. Here we calculate the correlation value for each feature of the dataset and keep its absolute value into an array (say array0) of N dimension. After we sort this array0 in descending order of values, we sort features of datasets in order of sorted array0; this gives a list call sorted list as list1. We pick top K (a user defined number) features from sorted list1. From selected features we formed a reduced dataset. Now we extend dataset by applying some functions to each feature of reduced dataset. After extending features we have a new dataset. We give it to different machine learning algorithms for classification. The complete algorithm and model of hybrid classification method are given in Figure 3, and Figure 4.

A. The Model of the proposed Hybrid Method

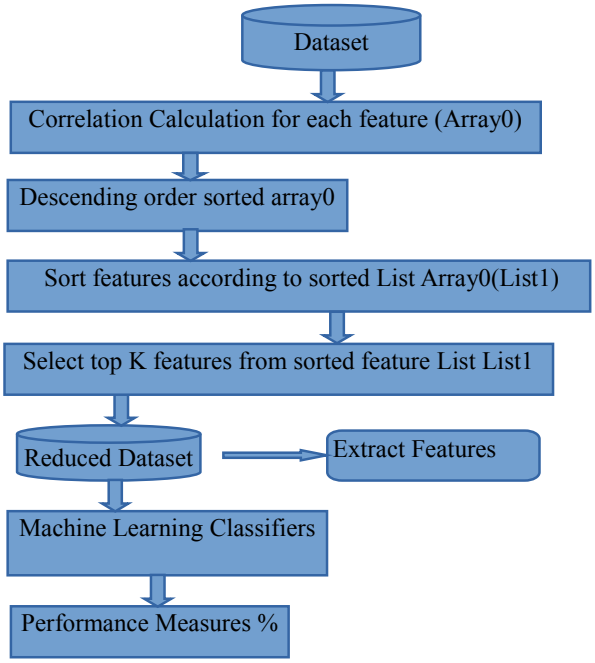

Figure 3: The Model of the proposed Hybrid Method

B. The Algorithm for the proposed hybrid method

Input: Dataset D[MxN]
K a user Defined Number
ML classifiers
Output: Classification Measures
{Accuracy, Sensitivity, *F-Measure,* Specificity, Precision, MCC and features}
Start:
1. Array0[1, N]
2. For ii=1: N
3. COR=CORRCAL(Feature(ii), Class)
4. Array0(ii)=COR
5. End
6. Sort array0 in descending order.
7. Sort feature in order of sorted array0 call list1.
8. Select top K features from Sorted List List1 call it SELFS
9. Reduced dataset= D (: SELFS)
10a. Extract features from (9).
10b. Classification Measures = ML classifiers (Reduced Dataset)
End.

Figure 4: The Algorithm for the proposed hybrid method

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we show the experimental results of our real-world prepaid smart meter data sets. All experiments are executed on an Intel Core i5 with a 3.40GHZ processing speed and 8GB main memory.

A. Datasets and parameters setting

To evaluate the usefulness of the proposed algorithm, we carry out experiments on our real-world prepaid smart meter data sets. This dataset have 21 numbers of features, 2 classes and 39249 instances, and they are used as representative sample of feature selection problems that the proposed algorithms can settle. The dataset characteristics are summarized in Table I.

TABLE I. THE DISTRIBUTION OF EXPERIMENTAL DATASETS

Dataset	# Instances	# Features	# classes
Prepaid SMD	39249	21	2

The classification performances of our method are evaluated by seven machine learning classifiers with 3-fold cross-validation on real-world data sets through the experiments. The results achieved in 10 independent runs are similar to each other in terms of the classification performances of the evolved feature subsets. Therefore, the best results from 10 independent runs are employed in this paper.

B. Experiment's Procedures

We focused on feature visualization and selection as a different from other kernels. Feature selection with correlation methods was used with 7 machine learning classifiers. Apart from these, principle component analysis was used to observe number of components.

The approach we used to load data for machine learning in python was Pandas and the pandas.read_csv() function. This function is very flexible and it returns a pandas DataFrame that we can immediately start summarizing and plotting.

Before making anything like feature selection, feature extraction and classification, we started with basic data analysis, by looking at features of data; dimensions of the data both in terms of rows and columns, data type for each attribute and descriptive statistics of the data that gave us great insight into the shape of each attribute. Normal, the describe() function on the Pandas DataFrame lists 8 statistical properties of each attribute which are count, mean, standard deviation, minimum value , 25^{th} percentile , 50^{th} percentile (Median), 75^{th} percentile and maximum value.

Data Preprocessing is a required step, but difficulty because different algorithms make different assumptions about data and may require different transforms. Further, when all of the rules are followed and prepare the data, sometimes algorithms can deliver better results without the preprocessing. Generally, creating many different views and transforms the data, then exercise a handful of algorithms on each view of dataset. This can help to flush out which data transforms might be better at exposing the structure of the problem in general.

Rescale data is to normalize and attributes into the range between 0 and 1. To rescale data we used scikit-learn using the MinMaxScaler class. After rescaling we saw that all of the values are in the range between 0 and 1.

Standardize Data; Standardization is a useful technique to transform attributes with a Gaussian distribution and differing means and standard deviations to a standard Gaussian distribution with a mean of 0 and a standard deviation of 1. We used scikit-learn with the StandardScaler class. The values for each attribute now have had a mean value of 0 and a standard deviation of 1.

Normalize Data Normalizing in scikit-learn refers to rescaling each observation (row) to have a length of 1 (called a unit norm in linear algebra). We used scikit-learn using the Normalizer class. The rows are normalized to length 1.

Binarize Data (Make Binary) you can transform your data using a binary threshold. All values above the threshold are marked 1 and all equal to or below are marked as 0. We used scikit-learn with the Binarizer class.

To visualize data, we used seaborn for diversity of plots. Univariate plots such as Histogram used to understand each attribute independently, and we used box and whisker plots or boxplots to review the distribution of each attribute.

In order to compare two features deeper, we used joint plot. By looking at the joint plot, it is correlated where Pearson correlation values is 1 as highest. Therefore, 0.9 is looks enough to say that they are correlated. For three or more features comparison, we used pair grid plot or multivariate plots such as heatmap plot method to observe all correlation between features**.** We calculated the correlation between each pair of attributes (correlation matrix). We then plotted the correlation matrix and got an

idea of which variables have a high correlation with each other. The matrix was symmetrical, i.e. the bottom left of the matrix is the same as the top right. This is useful as we saw two different views on the same data in one plot. We also saw that each variable is perfectly positively correlated with each other in the diagonal line from top left to bottom right.

C. *Results and Analysis*

In this part we used our proposed hybrid model of correlation based filter feature selection and 7 different selected machine learning classifiers, to evaluate our model; we applied it on our smart meter dataset.

Correlation between Attributes Correlation refers to the relationship between two variables and how they may or may not change together. For calculating correlation, we used Pearson's Correlation Coefficient, which assumes a normal distribution of the attributes involved. A correlation of -1 or 1 shows a full negative or positive correlation respectively. Whereas a value of 0 shows no correlation at all. As such, it is a good idea to review all of the pair-wise correlations of the attributes in the dataset. We used the corr() function on the Pandas DataFrame to calculate a correlation matrix.

The matrix lists all attributes across the top and down the side, to give correlation between all pairs of attributes (twice, because the matrix is symmetrical). The diagonal line through the matrix from the top left to bottom right corners of the matrix shows perfect correlation of each attribute with itself. Four things took our attention during this experiment; 1) There were some features that cannot be used for classification 2) LAST_AVAIL__002 feature is our class label 3) features includes NaN also are not needed. 4) We did not have any idea about other feature names because machine learning is awesome.

As it can be seen in below correlation matrix, there are not many correlated features.

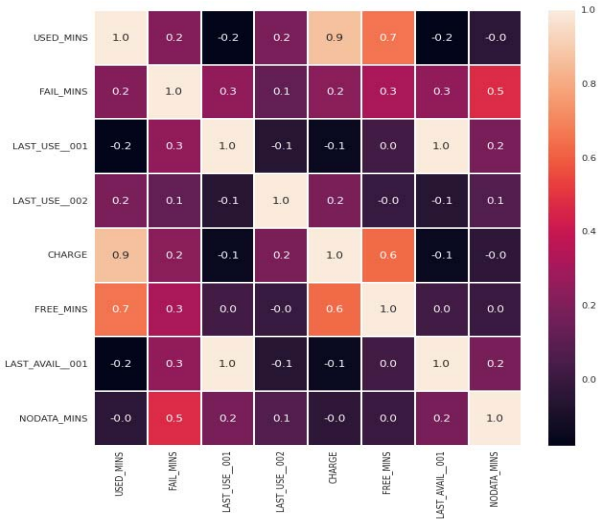

Figure 5: Correlation matrix for the correlated features.

From this figure 5, we see that there is correlation value 0.9 but let's see together what happens if we do not drop it. We choose our features but did we choose correctly? In the experiments, all the instances in dataset are randomly divided into two sets: 70% as the training set and 30% as the test set. By implementing our model, we used 7 different classifiers to evaluate it and find the classification performances according to chosen features.

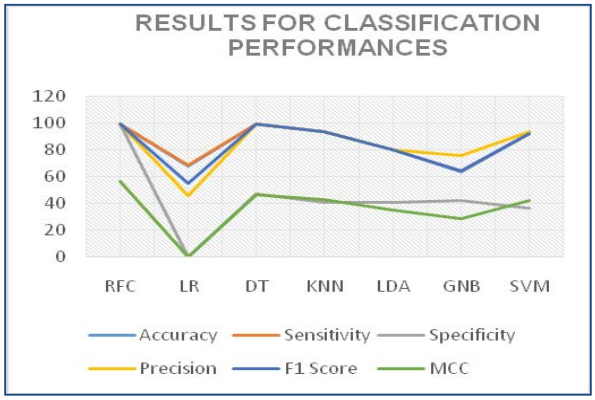

Figure 6: Classification performance Results

We compare the classification performances metrics against different machine learning classifiers on the number of features selected. The classification accuracies, sensitivities, Specificities, precisions F measures and MCC are the average of seven different classifiers (RF, LR, DT, KNN, LDA, GNB and SVM) to reduce the bias of a specific classifier. The results are shown in Figures 6. The numbers of features selected K are 8, on X-axis represents the averages of classification performances for different parameters used. The Y-axis represents the seven different classifiers. Different figures represent different parameters classification measures for our proposed methods.

A good feature selection method can find out informative features accurately. Here, the meaning of accuracy contains two aspects. On one hand, the feature selection method should select the features that obtain the better classification performance. On the other hand, it should select the number of features as small as possible.

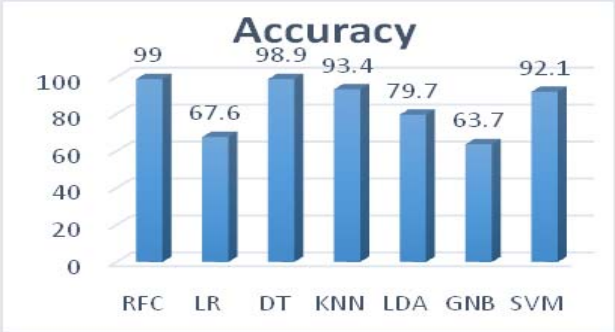

Figure 7: Average classification Accuracies on the proposed model by all compared ML classifiers (The high the better).

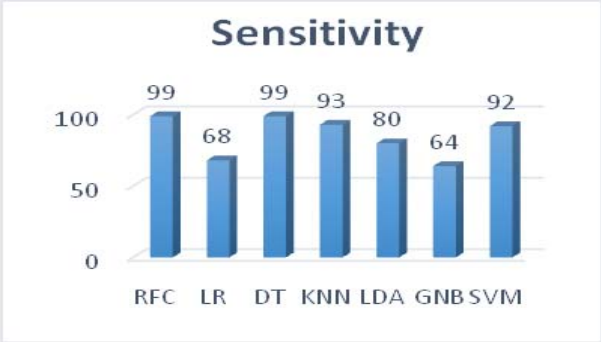

Figure 8: Average classification sensitivities on the proposed model by all compared ML classifiers (The high the better).

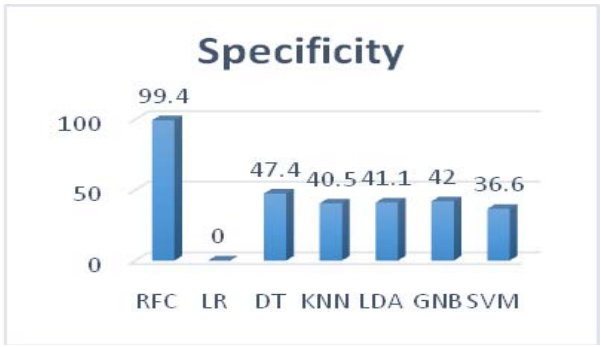

Figure 9: Average classification specificities on the proposed model by all compared ML classifiers (The high the better).

Figure 10: Average classification precisions on the proposed model by all compared ML classifiers (The high the better).

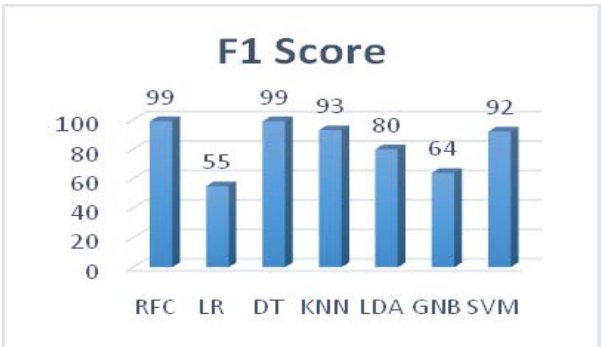

Figure 11: Average classification F-measures on the proposed model by all compared ML classifiers (The high the better).

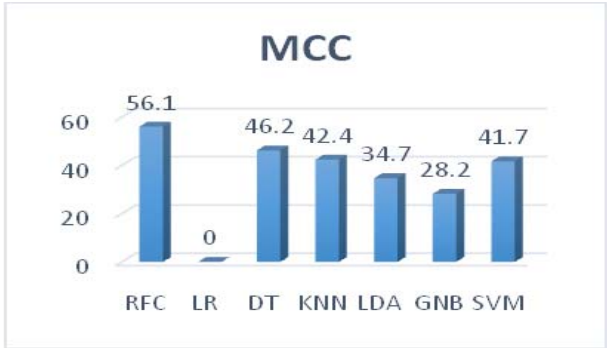

Figure 12: Average classification MCC on the proposed model by all compared ML classifiers (The high the better).

In figure 6, according to the number of selected features K, we list the highest value of the average classification parameters attained by seven classifiers on our data set using the features selected by our proposed feature selection method. Figure 6 shows that the highest accuracies of the proposed method is 99% for RF classifier, the highest sensitivities are 99% for RF and DT classifiers, the highest specificities is 99.4% for RF, the highest precisions and F1 scores are 99% for RF and DT and the highest MCC is 56.1% for RF.

Our method achieves the highest classification accuracy, sensitivity, specificity, precision, F1 scores and MCC for RF. Additionally, we can discover that our method outperforms in terms of the highest accuracies on seven classifiers. Seven benchmark classifier; RF, LR, DT, kNN, LDA, G NB and SVM were applied in these experiments to evaluate the performance of selected features by using six performance measures accuracy, sensitivity, specificity, precision, F-Measure and MCC.

In general, selected features by the proposed method got the good classification performance for all classifier as shown from figure 7, 8, 9 until 11.

From Figure 7, Average classification Accuracies on the proposed model by all compared ML classifiers show that RF outperform with 99%, against DT with 98.9%, kNN with 93.4, SVM with 92.1%, LDA with 79.7%, LR with 67.6% and GNB with 63.7%.

From Figure 8, Average classification sensitivities on the proposed model by all compared ML classifiers show that RF and DT outperform with 99%, against kNN with 93%, SVM with 92%, LDA with 80%, LR with 68% and GNB with 64%.

From Figure 9, Average classification specificities on the proposed model by all compared ML classifiers show that RF outperform with 99.4%, against DT with 47.4%, kNN with 40.5%, SVM with 36.6%, LDA with 41.1%, LR with 0% and GNB with 42%.

From Figure 10, Average classification precisions on the proposed model by all compared ML classifiers show that RF and DT outperform with 99%, against kNN with 93%, SVM with 93%, LDA with 80%, LR with 46% and GNB with 76%.

From Figure 11, Average classification F-measures on the proposed model by all compared ML classifiers show

that RF and DT outperform with 99%, against kNN with 93%, SVM with 92%, LDA with 80%, LR with 55% and GNB with 64%.

From Figure 12, Average classification MCC on the proposed model by all compared ML classifiers show that RF outperform with 56.1%, against DT with 46.2%, kNN with 42.4%, SVM with 41.7%, LDA with 34.7%, LR with 0% and GNB with 28.2%.

V. CONCLUSION AND FUTURE WORKS

We proposed a hybrid model of correlation based filter feature selection and machine learning classifiers. The aim of this study was to select relevant features and analyze the outperform machine learning algorithms in order to train our model, predict and compare their classification performance. In this study we ran the experiment on our smart meter dataset, using Correlation-based attribute Evaluation (CB) as a conventional technique to select the relevant features and to analyze the outperform machine learning algorithms commonly used by evaluating the performance of selected features using six performance measures accuracy, sensitivity, F-Measure, Specificity, Precision, and MCC. Since the purpose of feature selection method is not only to improve the classification accuracy but also to minimize the number of selected features. Additionally, we can discover that in the proposed method in terms of the associated number of selected features the RF outperforms with the highest classification performance on our data sets. In order to understand intuitively, we show the highest average classification performances of seven classifiers (RF, LR, DT, kNN, LDA, SVM, and NB) and the associated number of selected features are eight. Overall, our method outperforms better in all classifiers compared.

In the future work, we are planning to propose a fraud detection model and apply it to the extracted dataset, do feature engineering and compare the performances of the model.

REFERENCES

[1] Girish Chandrashekar and Ferat Sahin, A survey on feature selection methods, computer and Electrical Engineering 40(2014)16-28.

[2] Vipin kumar and sonajharia minz, feature selection: A literature review, Smart computing Review, vol 4 No 3 2014.

[3] J.Y. Choi, Y.M. Ro, K.N. Plataniotis, Boosting color feature selection for color face recognition, IEEE transactions on image processing 20 (2011) 1425-1434.

[4] A. Goltsev, V. Gritsenko, Investigation of efficient features for image recognition by neural networks, Neural Networks 28 (2012) 15-23.

[5] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, A simultaneous feature adaptation and feature selection method for content-based image retrieval systems, Knowledge-Based Systems 39 (2013) 85-94.

[6] S. Van Landeghem, T. Abeel, Y. Saeys, Y. Van de Peer, Discriminative and informative features for biomolecular text mining with ensemble feature selection, Bioinformatics 26 (2010) 554-560.

[7] F. Amiri, M.R. Yousefi, C. Lucas, A. Shakery, N. Yazdani, Mutual information-based feature selection for intrusion detection systems, Journal of Network and Computer Applications 34 (2011) 1184-1199.

[8] A. Alazab, M. Hobbs, J. Abawajy, M. Alazab, Using feature selection for intrusion detection system, in: Proceedings of International Symposium on Communications and Information Technologies (ISCIT), 2012, pp. 296-301.

[9] Q. Song, J. Ni, G. Wang, A fast clustering-based feature subset selection algorithm for high-dimensional data, IEEE Transactions on knowledge and data engineering 25 (2013) 1-14.

[10] Y.F. Gao, B.Q. Li, Y.D. Cai, K.Y. Feng, Z.D. Li, Y. Jiang, Prediction of active sites of enzymes by maximum relevance minimum redundancy (mRMR) feature selection, Molecular BioSystems 9 (2013) 61-69.

[11] H.J. Yu, D.S. Huang, Normalized feature vectors: a novel alignment-free sequence comparison method based on the numbers of adjacent amino acids, IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 10 (2013) 457-467.

[12] T.W. Rauber, F. de Assis Boldt, F.M. Varejão, Heterogeneous feature models and feature selection applied to bearing fault diagnosis, IEEE Transactions on Industrial Electronics 62 (2015) 637-646.

[13] K. Zhang, Y. Li, P. Scarf, A. Ball, Feature selection for high-dimensional machinery fault diagnosis data using multiple models and Radial Basis Function networks, Neurocomputing 74 (2011) 2941-2952.

[14] T. Khoshgoftaar, D. Dittman, R. Wald, A. Fazelpour, First order statistics based feature selection: A diverse and powerful family of feature seleciton techniques, in: Proceedings of 11th International Conference on Machine Learning and Applications (ICMLA), 2012, pp. 151-157.

[15] J. Gibert, E. Valveny, H. Bunke, Feature selection on node statistics based embedding of graphs, Pattern Recognition Letters 33 (2012) 1980-1990.

[16] M.C. Lane, B. Xue, I. Liu, M. Zhang, Gaussian based particle swarm optimisation and statistical clustering for feature selection, in: Proceedings of European conference on evolutionary computation in combinatorial optimization, 2014, pp. 133-144.

[17] H. Li, C.J. Li, X.J. Wu, J. Sun, Statistics-based wrapper for feature selection: an implementation on financial distress identification with support vector machine, Applied Soft Computing 19 (2014) 57-67.

[18] L. Shen, L. Bai, Information theory for Gabor feature selection for face recognition, EURASIP Journal on Applied Signal Processing (2006) 1-11.

[19] B. Bonev, Feature selection based on information theory, Universidad de Alicante, 2010.

[20] Z. Xu, I. King, M.R.T. Lyu, R. Jin, Discriminative semi-supervised feature selection via manifold regularization, IEEE Transactions on neural networks 21 (2010) 1033-1047.

[21] B. Jie, D. Zhang, B. Cheng, D. Shen, Manifold regularized multi-task feature selection for multi-modality classification in Alzheimer's disease, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2013, pp. 275-283.

[22] Y. Chen, D. Miao, R. Wang, A rough set approach to feature selection based on ant colony optimization, Pattern Recognition Letters 31 (2010) 226-233.

[23] [23] W. Shu, H. Shen, Incremental feature selection based on rough set in dynamic incomplete data, Pattern Recognition 47 (2014) 3890-3906.

[24] J. Derrac, C. Cornelis, S. García, F. Herrera, Enhancing evolutionary instance selection algorithms by means of fuzzy rough set based feature selection, Information Sciences 186 (2012) 73-92.

[25] [25] J. Wang, K. Guo, S. Wang, Rough set and Tabu search based feature selection for credit scoring, Procedia Computer Science 1 (2010) 2425-2432.

[26] A. Rodriguez, A. Laio, Clustering by fast search and find of density peaks, Science 344 (2014) 1492-1496.

[27] J. Han, J. Pei, M. Kamber, Data mining: concepts and techniques, Elsevier, 2011.

[28] D.S. Huang, J.X. Du, A constructive hybrid structure optimization methodology for radial basis probabilistic neural networks, IEEE Transactions on neural networks 19 (2008) 2099-2115.

[29] J. Tang, S. Alelyani, H. Liu, Feature selection for classification: A review, Data Classification: Algorithms and Applications, (2014) .

[30] S. Alelyani, J. Tang, H. Liu, Feature Selection for Clustering: A Review, Data Clustering: Algorithms and Applications 29 (2013) 110-121.

[31] J.R. Vergara, P.A. Estévez, A review of feature selection methods based on mutual information, Neural computing and applications 24 (2014) 175-186.

[32] [32] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, Recent advances and emerging challenges of feature selection in the context of big data, Knowledge-Based Systems 86 (2015) 33-45.

[33] J.C. Ang, A. Mirzal, H. Haron, H.N.A. Hamed, Supervised, unsupervised, and semi-supervised feature selection: a review on gene selection, IEEE/ACM transactions on computational biology and bioinformatics 13 (2016) 971-989.

[34] R. Shei khpour, M.A. Sarram, S. Gharaghani, M.A.Z. Chahooki, A Survey on semi-supervised feature selection methods, Pattern Recognition 64 (2017) 141-158.

[35] R. Amirreza, N. Hossein, A hybrid feature selection approach based on ensemble method for high-dimensional data, 2nd Conference on Swarm Intelligence and Evolutionary Computation (CSIEC), 7-9 March, pp. 16-20, (2017).

[36] [36] P. Cunningham, Dimension reduction, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.98.1478, technical Report UCD-CSI-2007-7 (2007).

[37] S. T. Roweis, L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326. arXiv:http://science.sciencemag.org/content/290/5500/2323.full.pdf, doi:10.1126/science.290.5500.2323. URL http://science.sciencemag.org/content/290/5500/2323

[38] G. Chandrashekar, F. Sahin, A survey on feature selection methods, Comput. Electr. Eng. 40 (1) (2014) 16–28. doi:10.1016/j.compeleceng.2013.11.024. URL http://dx.doi.org/10.1016/j.compeleceng.2013.11.024

[39] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, The Journal of Machine Learning Research 3 (2003) 1157–1182.

[40] B. Xue, M. Zhang, W. N. Browne, A comprehensive comparison on evolutionary feature selection approaches to classification, International Journal of Computational Intelligence and Applications 14 (02).

[41] J. Han, nd M. Kamber, Data Mining; Concepts and Techniques. Morgan Kaufmann Publishers, 2000.

[42] H. Almuallim, T. G. Dietterich, T. G. Learning boolean concepts in the presence of many irrelevant features. Artificial Intelligence, vol. 69, no. 1-2, pp.279– 305, (1994)

[43] D. Koller, M. Sahami, M. Toward optimal feature selection. In Proceedings of the Thirteenth International Conference on Machine Learning, pp. 284–292, (1996)

[44] M. A. Hall, L. A. Smith, Feature Subset Selection: A Correlation Based Filter Approach, In 1997 International Conference on Neural Information Processing and Intelligent Information Systems, pp.855-858, (1997)

[45] H. Liu, H. Motoda, Feature selection for knowledge discovery and data mining, Springer Science & Business Media, 2012.

[46] B. Bonev, Feature selection based on information theory, Universidad de Alicante, 2010.

[47] J. Han, J. Pei, M. Kamber, Data mining: concepts and techniques, Elsevier, 2011.

[48] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, The Journal of Machine Learning Research 3 (2003) 1157–1182.

[49] F. Sebastiani, C. N. D. Ricerche, Machine learning in automated text categorization, ACM Computing Surveys 34 (2002) 1–47.

[50] Y. Yang, J. O. Pedersen, A comparative study on feature selection in text categorization, in: Proceedings of the Fourteenth International Conference on Machine Learning, ICML '97, 1997, pp. 412–420.

[51] G. Forman, An extensive empirical study of feature selection metrics for text classification, The Journal of Machine Learning Research 3 (2003) 1289–1305.

[52] D. Mladenic, Machine learning on non-homogeneous, distributed text data., Ph.D. thesis, University of Ljubljana, Faculty of Computer and Information Science (1998).

[53] H. T. Ng, W. B. Goh, K. L. Low, Feature selection, perceptron learning, and a usability case study for text categorization, in: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '97, 1997, pp. 67–73.

[54] M. Y. Munirah, M. Rozlini, N. Wahid, A comparative analysis on feature selection techniques for medical datasets, APRN Journal of Engineering and Applied Sciences, vol 11, no 22, November 2016, (2016)

[55] C. Shang, M. Li, S. Feng, Q. Jiang, J. Fan, Feature Selection via Maximizing Global Information Gain for Text Classification, Knowledge-Based Systems, vol. 54, 298-309, (2013)

[56] J. H. Lee, J. R. Anaraki, C. W. Ahn, J. An, Eficient Classification System based on Fuzzy-Rough Feature Selection and Multitree Genetic Programming for intension Pattern Recognition using Brain Signal, Expert Systems with Applications, vol. 42, 1644-1651, (2015)

[57] S. Kashef, H. Nezamabadi-pour, An Advanced ACO Algorithm for Feature Subset Selection, Neurocomputing vol. 147, 271279, (2015).

[58] Y. Zhang, D. Gong, Y. Hu, W. Zhang,. Feature Selection Algorithm based on Bare Bones Particle Swarm Optimazation, Neurocomputing, vol. 148, pp.150-157, (2015)

[59] Y. Shen-Lan, R. Gang, F. Yi-Ping, Multiple kernel learning based feature selection for process monitoring, 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), 24-16 May 2017, pp. 809-814, (2017)

[60] B. Emel, S. Mustafa, Video classification based on ConvNet collaboration and feature selection, 25th Signal Processing and Communications Applications Conference (SIU), 15-18 May 2017, pp. 1-4, (2017).

[61] D. Koller, and M. Sahami, "Toward optimal feature selection". In Proceedings of the Thirteenth International Conference on Machine Learning, pp.284–292, (1996).

[62] Rodrigues, L. Pereira and R.Y.M Nakamura, "Wrapper approach for feature selection based on bat algorithm and optimum-path forest", Expert Systems with Applications, vol.41, pp.2250–2258, (2014).

[63] R.Y.M. Nakamura, L. Pereira, M. Acuckoo, K.A Costa, D. Rodrigues, J.P.Papa, X.S. Yang, "BBA: a binary bat algorithm for feature selection'. in 25th, SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), 22–25 August, IEEE Publication, pp. 291-297, (2012).

[64] A.M Taha, A. Mustapha and S.D. Chen, "Naïve Bayes-Guided bat algorithm for feature selection", The Scientific World Journal, vol 2013, http://dx.doi.org/10.1155/2013/325973, (2013).

[65] K. Dubey and A. K. Saxena, Hybrid Classification Model of Correlation-based Feature Selection and Support Vector Machine, 978-1-5090-1936-6/16/$31.00 ©2016 IEEE.