

A Novel Approach of Ensemble Learning with Feature Reduction for Classification of Binary and Multiclass IoT Data

Mr. Vijay M. Khadse¹, Dr. Parikshit N. Mahalle², Dr. Gitanjali R. Shinde³

¹Assistant Professor, College Of Engineering Pune (COEP), Pune, India, Email: vmk.comp@coep.ac.in

²Senior Member IEEE, Professor & Head, Department of Computer Engineering, Smt. Kashibai Navale College of Engineering, Pune India. PostDoc Researcher, Center for Communication, Media and Information Technologies (CMI), Aalborg University, Copenhagen, Denmark, Email: aalborg.pnm@gmail.com

³Assistant Professor, Smt. Kashibai Navale College of Engineering, Pune India, Email : gr83gita@gmail.com

Article History: Received: 10 November 2020; Revised 12 January 2021 Accepted: 27 January 2021; Published online: 5 April 2021

Abstract: The number of network and sensor-enabled devices in the Internet of Things (IoT) domains is growing extremely, leading to a huge production of data. These data contain important information which can be used in various areas, such as science, industry, medical, and even social life. To make the IoT system smart, the only solution is entering the world of machine learning. Many machine learning algorithms are introduced for handling such a huge amount of IoT data. It is very difficult to find the best-suited algorithm for problems in the IoT domain. This study combined three ensemble models and proposed a new model termed the “hybrid model”. A set of features are extracted from the raw IoT datasets from diverse IoT domains, using Principal component analysis (PCA), Linear discriminant analysis (LDA), and Isomap for classification problems. Performance comparison of the classifiers is provided in terms of their accuracy, area under the curve (AUC), and F1 score. This comparative study’s experimental result shows that Hybrid with PCA and Stacking ensemble technique in particular with PCA have better overall performance than other ensemble techniques for binary class and multie class datasets respectively

Keywords: Ensembles, Bagging, Boosting, Stacking, Random Forest, Classification, Binary classification, Multiclass classification, Hybrid ensembles.

1. Introduction

The Internet of Things (IoT) is the most widely spreading fields in every aspect of human life (Singh and Singh, 2015). IoT systems are integrated into many applications such as Home automation, Smart cities, Manufacturing, Aviation, Health care, Transports, Network security, Self-driven automobiles are the few to be mentioned (Atzori et al., 2010). IoT devices supports number of applications such as smart cameras and smoke detectors for security; smart light bulbs for home and industry, and sockets facilitate power savings; and so forth (Meidan et al., 2017).

Application of machine learning (ML) is expanding rapidly in IoT systems especially with the emergence of fast mobile devices that also have access to cloud computing (Ularu et al. 2013). IoT devices generate huge amounts of data in every field of their application. Data generated for IoT systems is mostly continuous values. It has an advantage over categorical data, as it can be naturally ordered and similarity and distance functions can be defined on them (Boriah et al., 2008; Wilson and Martinez, 1997). Raw data generated by IoT devices need to be abstracted. Analytics should be performed for patterns and useful inferences. ML is widely applied in IoT for knowledge extraction. There is a widespread use of ensemble models of ML and pattern recognition application due to their ability to significantly improve accuracy as compared to base algorithms.

Ensemble Learning (EL) is the state-of-the-art for different ML problems. In EL there are a group of base learners (on average 5 to 6) which means a group of models for processing. The main aim of EL is to combine these models, make the one strong learner. Therefore, the obtained result will be much better than the single base learner (Atzori et al., 2010).

There is a common situation where there is no availability of sufficient historical data from an IoT application for learning. One of the major challenges of ML to IoT systems is to identify an optimal learning algorithm for classification that could be applied over diverse IoT domain.

The objective behind this research is to:

- (1) To identify an optimal ensemble learning technique suitable for diverse IoT domain.
- (2) To identify suitable feature reduction technique to be applied for effective performance.
- (3) To identify suitability of learning algorithm based on number of class labels i.e binary or Multiclass data separately.
- (4) To study and compare proposed hybrid ensemble model of learning with bagging, boosting and stacking ensemble model over diverse IoT domain data.

This research tries to achieve above objectives by taking datasets of varying size, varying no of features from different IoT application domains and tries to address the problem by comparing ML ensemble techniques for classification based on their performance.

The paper is organized as: section 2 contains a literature survey about the previous and related work done by others authors. Section 3 focuses on the analysis of gaps in the previous work done. Section 4 describes our proposed work and methodology. Section 5 contains our experimentation and results. Finally, the study ends with section 6 containing observations and conclusions

2. Literature Survey

In learning techniques, the number of component classifiers in an ensemble model and extracted components from original features using the feature reduction technique has a great impact on its performance.

Junior et al. (2020) have compared feature selection and dimensionality reduction techniques on gesture recognition sensor data to increase the performance. They have used eight dimensionality reduction techniques namely Linear Discriminant Analysis, Manifold Charting, Autoencoder, t-distributed Stochastic Neighbor Embedding, Principal Component Analysis, Large Margin Nearest Neighbor (LMNN) and Isomap. They have also used seven different classifiers. They observed that 87% to 90% accuracy achieved for ELM, SVM and RBF classifiers in feature selection and 95% accuracy achieved for a combination of LMNN and SVM in the dimensionality reduction process. This study showed dimensionality reduction improves the performance of the hand gesture dataset.

Ribeiro and dos, (2020) compared the performance of Bagging (random forest), Boosting (gradient boosting), Boosting (extreme gradient boosting XGB) and Stacking ensemble techniques on agriculture business time-series dataset. Least absolute shrinkage and selection operator (LASSO), Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and XGB were used for the level-0 layer in stack ensemble technique. XGB or Stack and RF models have performed better for short-term forecasts.

Taşer et al. (2019) combined MIWrapper and SimpleMI algorithm with Naïve Bayes, SVM, C4.5, Multilayer perceptron and Decision tree. They proposed a new ensemble-based multiple instance learning algorithms. They compared multiple instances learning ensemble algorithms with individual MIWrapper and SimpleMI. Experimental results show that the proposed ensemble-based model provides higher classification accuracy than traditional solutions.

Alexandropoulos et al. (2019) proposed a stacking ensemble methodology using Logistic Model tree and three well-known ensembles namely extra tree, random forest and gradient boosting. They concluded that stacking methodology gave remarkable performance than individual classifiers performance.

Suganthi and Karunakaran (2019) used the Cuttlefish optimization algorithm for data point reduction. This optimally extracted subset of data points and a reduced set of features provided by PCA, providing almost the same accuracy, a false positive rate that they obtained from the original dataset.

Tounsi et al. (2018) compared seven base classifiers each for four ensemble methods. They have applied it to the financial domain dataset. LogR, MLP, C4.5, CDT, CART, SVM and Pegasos are base classifiers and AdaBoost, Bagging, Random Subspace, Decorate and Rotation Forest are ensemble methods. It has used four evaluation metrics, Area Under the Curve, Accuracy, False positive rate and Time taken to build the model, for performance evaluation for each base classifier. It is observed that Pegasos algorithm performed better as a base classifier than other base classifiers for AdaBoost. C4.5, CART and CDT classifiers performed better than others for bagging, Random Subspace, Decorate and Rotation Forest.

Vijay (2018) compared feature extraction and feature selection techniques on 6 IoT datasets. They have used Principal Component Analysis (PCA), Generalized Hebbian Algorithm (GHA), Independent component analysis (ICA), Singular Value Decomposition (SVD) and Self Organizing Map (SOM) for feature extraction. For feature selection, filter and wrapper techniques are used. For performance evaluation, compactness,

accuracy and computational time are used. They concluded that feature extraction techniques performed well on low dimensional data and feature selection perform better on high dimensional data.

Rojarath (2016) compared Naïve Bayes, Decision Tree, Multilayer Perceptron and K-Nearest Neighbour classification methods on a variety of UCI datasets. Multilayer Perceptron achieves high accuracy. They made two ensemble models namely 3-ensemble and 4-ensemble. Using majority voting technique on ensemble models, experimental results show that 3-ensemble models achieve high accuracy with 83.13%.

Yu et al. (2016) compared accuracy of PSEL ensemble approach with single classifiers (KNN, SVM, C4.5, Random Tree, Random Vector Functional Linking) on 18 gene expression datasets and four UCI datasets. They compared the accuracy of PSEL Ensemble models with Random subspace, AdaBoost, Random-Forest, Multiboost, Bagging, CEREP, CECMP and RTboost on UCI datasets. Results show that Progressive Subspace EL (PSEL) performed better as compared to other ensemble models.

Narassiguin (2016) compared 19 ensemble algorithms including Bagging, Boosting, Random forest, Rotation forest and its variants. They have used Decision tree base learners for meta-classifiers. For this study, 19 various binary UCI datasets were considered. Accuracy, Area Under the Curve (AUC), Root Mean Square (RMS) were used as performance matrices. They concluded that the family of the Rotation forest ensemble technique outperformed other ensemble techniques.

Pandey et al. (2014) applied ensemble classifiers on student performance. These classifiers are bagging, boosting, Random Forest, Rotation Forest and AdaBoost. It has been observed that the Rotation forest is performing very well and the Random Forest algorithm performing very low. AdaBoost and Bagging performed better than Random Forest and close to Rotation Forest. The accuracy of the Rotation Forest is 75.95%.

Wan and Yang (2013) compared four popular ensemble methods namely Bagging, Boosting, Stacking and Random forest on 31 UCI datasets and justified that, depending upon the dataset domain the result varies means accuracy varies. Therefore no one was the winner from EL methods.

Ye and Suganthan (2012) discussed four bagging-based ensemble classifiers, namely, the ensemble ANFSI, the ensemble SVM, the ensemble ELM and random forest. Ensemble classifiers evaluated on thirteen UCI binary datasets with different bagging numbers (20, 50 and 80). Out of four ensemble classifiers, the ensemble SVM has been identified to be the most favourable ensemble classifier and random forest tree identified second most favorable ensemble classifier.

Syarif et al. 2012 investigated network intrusion detection systems by applying three ensemble methods (bagging, boosting and stacking). Results show that only the stacking method was able to reduce false positive rate as compared to other ensemble methods. Among the four classifiers (J48, naive Bayes, JRip and iBK-nearest neighbour), J48 performed better than the three other methods by achieving the highest accuracy rates and lowest false positive rate.

Wang (2012) compared Bagging, Boosting and Stacking ensemble techniques with Decision tree, Artificial neural network, Support vector machine and logistic regression as a base learner on credit scoring datasets. In this study, Accuracy, type I and type II error were considered for performance measurement of models. They concluded that Stacking and Bagging with decision trees performed better than all ensemble models in terms of accuracy, type I and type II error.

Graczyk et al. (2010) examined six distinct ML classifiers to three ensemble techniques i.e. Bagging, Stacking and Additive Regression. Accordingly, models produced by stacking have lowest prediction error. Bagging approach found to be more stable but gave poor performance than stacking and additive regression.

3. Gap analysis

Based on the literature survey, it is observed that researchers while using ensemble methods for comparison on multi-domain datasets do not consider feature reduction techniques. Similarly, while comparing feature reduction techniques on multiple domain datasets they do not use ensemble methods.

Unlike existing studies, this study not only compares the performance of Bagging, Boosting and Stacking models but also the proposed hybrid ensemble model. This study considers PCA, LDA and Isomap as feature reduction techniques to improve the performance of the model on diverse multi-domain binary and multiclass multi-domain IoT datasets.

4. Proposed methodology

4.1. Design

Figure 1 is the proposed methodology for this study. It involves the EL of individual classifiers and feature reduction techniques which are mentioned above. This methodology is divided into five stages. Following are the stages included for performing the comparative study of feature reduction techniques and EL methods

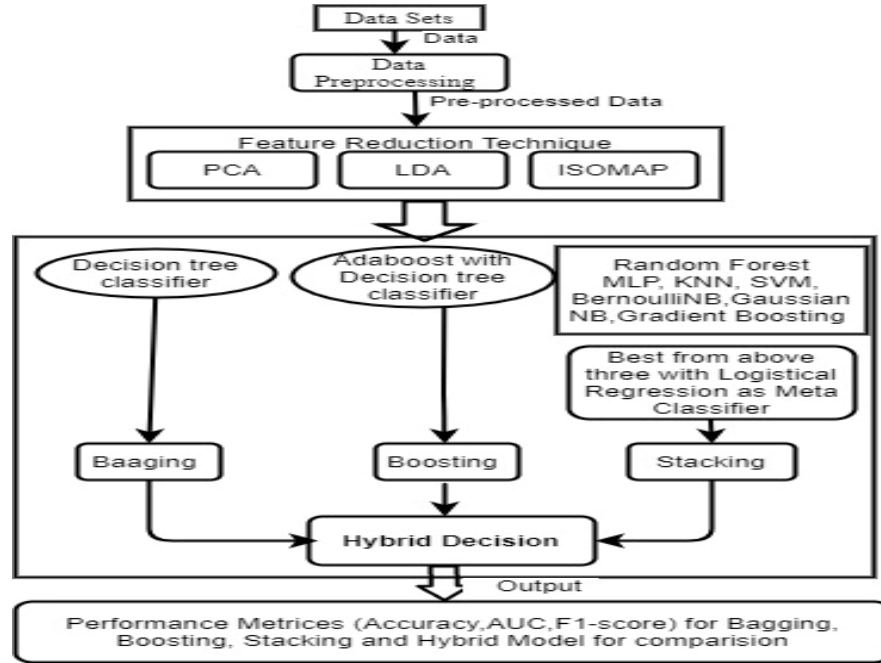


Figure 1. Methodology used in hybrid approach for classification of binary and multiclass data.

4.2 Dataset

This study has collected Ten (10) binary and Ten (10) multiclass IoT sensor datasets of different domains from UCI ML storehouse (Asuncion et al., 2007) and kaggle. Some of the data sets are of high dimension and some with low dimension to reduce any favourable and unfavourable impact on the performance of algorithms. Table 1 contains the information about features, classes, class types and instances of datasets.

Table 1. Description of dataset used for experimentation

Sr No.	Datasets	Total features	Total classes	Class types	Instances
1	Electric grid	14	2	Binary	10000
2	Extra sensory - B	277	2	Binary	2686
3	Football sensor	9	2	Binary	945
4	Pulsar star	9	2	Binary	9652
5	EEG signal	15	2	Binary	8123
6	Power system - A	129	2	Binary	5161
7	Hand gesture recognition	65	2	Binary	5811
8	Watch sensor	13	2	Binary	7386
9	Power system - B	99	2	Binary	4966
10	Machine sensor	75	2	Binary	10616
11	Cardiotocography sensor	41	10	Multiclass	2126
12	Extra sensory - A	277	6	Multiclass	2686
13	Mode detection	33	5	Multiclass	5894
14	Sky server	18	3	Multiclass	10000
15	Movement recognition	563	6	Multiclass	2948
16	Air quality sensor	16	5	Multiclass	9358
17	Energy prediction	29	8	Multiclass	19736
18	Big sensors	561	6	Multiclass	6373

19	Transport detection	38	5	Multiclass	5894
20	Direction sensor	25	4	Multiclass	5455

4.3 Data pre-processing

Out of range values, missing values, impossible data combination etc lead to undesirable effect on ML prediction model. Gathered data from the resources is not in a standardized form. This data contains a lot of null and missing values. This study removes the null or missing values in data. Large numerical values require normalizing data for feature reduction. Column normalization technique is applied datasets. Data values are rescaled between ranges of 0 to 1. Also checked the data for positive or negative infinity.

4.4 Feature reduction

Due to a large number of features in the dataset, it becomes complex to visualize the data. Many of the features are correlated, so it becomes redundant. Using feature reduction techniques, higher dimensions of dataset get converted into a new set of synthetic dimensions and extracted lower dimensions to avoid overfitting problem and improve the performance of the model.

This study used two linear and two nonlinear feature reduction techniques. Principal component analysis (PCA) is a linear technique for reducing dimensionality as well as minimizing information loss. Using these methods, new uncorrelated data is created by minimizing variance. Linear discriminant analysis (LDA) is also a linear technique for reducing the dimensionality based on the classes of the dataset. Using this technique, this study finds the dimensions which can maximize the separability between the classes to make a good decision to classify data. Isomap is a nonlinear technique for visualization of data and computes low dimensional embedding of high dimensional data. The number of neighbours depends on the number of instances in datasets. Isomap is very efficient and used for a high number of dimensionalities. For the visualization of the data, the study applied T - distributed stochastic neighbour embedding (T-SNE) technique. To observe how the data is separated, the study visualizes the data in 2 dimensions.

4.5 Ensemble method

This work considers Bagging (Breiman, 1996), Boosting (Freund and Schapire, 1996), Stacking (Wolpert, 1992) ensemble techniques and proposes a new “hybrid” model for classification. The proposed methodology used a decision tree classifier as a base learner for bagging and boosting. Also, Adaboost is used as a meta classifier for boosting.

Stacking combines multiple models, makes training data from their predictions and applies a meta classifier on the training data (Wolpert, 1992). Because of this approach, the performance of the model gets improved. For the first layer in the stacking model, eight classifiers were selected from several families of algorithms. Random forest classifier from tree family (Amasyali and Ersoy 2008), Multilayer layer perceptron classifier (MLP) from neural network family (Wilamowski, 2009), Gradient boosting classifier from ensemble family (Mason et al. 2000), Bernoulli and Gaussian classifiers from Naïve Bayes family, K-nearest neighbour classifier (KNN) from instance-based family (Aha et al. 1991), Logistic regression classifier from regression family (Gay and Welsch, 1988) and Support vector machine (SVM) classifier belong to the generalized linear classifier (Tyagi and Manry, 2019). Out of eight classifiers, three high-performance classifiers were selected for binary as well as multiclass datasets for making a stack. There are two methods to build a stack of classifiers in stacking ensemble models i.e. using prediction values of the classifier and using probabilities of prediction values of the classifiers. This research work selects probabilities of prediction values of the classifiers method for both binary and multiclass datasets to build the stacking model. This technique helps to boost the performance of the stacking model. For the second layer in the stacking model, since the data in this study is derived from the original dataset with complex transformation, it is not necessary to select complex classifiers in the output layer. Logistic regression is a good choice and it also prevents over-fitting. That is why Logistic Regression is selected as a meta classifier.

This study merged predictions of bagging, boosting, stacking ensemble models and proposed a new “Hybrid Ensemble Model” (HEM). For binary datasets, it combined the prediction values (values which are predicted on the training dataset) of bagging, boosting, stacking ensemble models and made new predicted values. For Multiclass datasets, it combined the prediction values as well as the probability of prediction values (probability

of predicted values made on the training dataset) of bagging, boosting, stacking ensemble models and made our new predicted values. Finally, the study compares predicted value to test dataset (contained actual value of the inputs) to measure the performance. The “HEM” aims to improve the stability of the EL model. Even if training data is slightly modified, the prediction will not change.

4.6 Performance matrices

For this comparative study, the performance of models is compared using three techniques. Accuracy is the most common and essential technique to measure the prediction rate of the model. For multiclass classification problems that come into the picture, Area under the ROC Curve (AUC) is a must for use. This technique shows how much a model can classify between labels. Higher the score of AUC well predicted the classes. F1 score shows the balance between precision and recall. However, it does not consider true negative in measurement.

5. Experimentation and results

Scikit learn is a useful library for ML (Pedregosa, 2011). It offers a number of supervised and unsupervised learning algorithms via a simple python framework. In the scikit learn, PCA, LDA and Isomap include common one parameter i.e. “n-components”.

This parameter indicates the number of features to be returned for further processing. To figure out the value of a parameter, a corresponding function of its feature reduction technique is used. Before applying the feature reduction technique, data is converted into the standardized form using the “standard scalar” function.

5.1 Selection of “n” Features

In PCA, using “Explained variance ratio” new “n” features are created from features. A cumulative sum of variance ratio of each feature is returned in ascending order and the cumulative variance is plotted. This tells how many “n-components” are required to cover the whole variance. For the study, threshold variance is set at 95%.

LDA creates its own new components based on the labels of the dataset. Suppose a dataset has ‘x’ features then LDA creates its own new ‘x-1’ features. Due to this strategy, this study observes the reduced dimensions of LDA are far lesser than the PCA and Isomap. To extract new feature set, “Explained variance ratio” is applied similar to PCA. A function is built which consecutively adds the explained variance of features until the threshold can’t fit any more features. Finally, it returns the number of features added. The threshold is set to be 95%.

Isomap determines the number of neighbours. Selecting a large number of neighbours makes it computationally expensive. Due to this problem, the square root of the total instances of the dataset is taken and initializes the number of neighbours for the Isomap. To extract the features, the “Reconstruction error” function of Isomap is applied. This function signifies the distance between the original data point and its projection point onto a lower-dimensional space. This study provides the error rate of each feature. This increases the number of features, decreases the reconstruction error rate. After a certain number of features, the reconstruction error rate stabilizes. Number of features was chosen where the reconstruction error rate gets stabilized. Table 2 contains details about the total features and reduced features of each IoT binary and multiclass dataset.

Table 2. Dimensions reduce by PCA, LDA and Isomap

Sr No.	Datasets	Total Features	Reduced dimension		
			PCA	LDA	Isomap
1	Electric grid	14	10	1	11
2	Extra sensory - B	277	18	1	21
3	Football sensor	9	7	1	8
4	Pulsar star	9	4	1	3
5	EEG signal	15	5	1	9
6	Power system - A	129	22	1	27
7	Hand gesture recognition	65	42	1	23
8	Watch sensor	13	6	1	4
9	Power system - B	99	18	1	22
10	Machine sensor	75	13	1	16
11	Cardiotocography sensor	41	12	1	15
12	Extra sensory - A	277	11	4	23
13	Mode detection	33	5	3	11

14	Sky server	18	8	2	10
15	Movement recognition	563	110	3	195
16	Air quality sensor	16	4	4	5
17	Energy prediction	29	13	1	14
18	Big sensors	561	190	3	209
19	Transport detection	38	7	3	13
20	Direction sensor	25	18	3	18

5.2 Results

To evaluate the efficiency of the model and reduce the overfitting and underfitting problems, the cross-validation technique is used. Datasets utilized have not enough instances to construct an optimal model and results fluctuate for different splits of the data. Due to these problems, K-fold cross-validation technique is applied. In K-fold cross-validation, there is a bias-variance trade-off correlated with the decision of K (James et al. 2013). Generally, despite these criteria, one performs K-fold cross-validation with K=5 or K=10, since values have been experimentally shown to provide test error rate estimates that do not suffer from extreme bias or extremely high variance. For performance evaluation, this study has used 5-fold cross-validation technique.

First, used bagging model with decision tree classifier working as a base learner and evaluated on each dataset for Accuracy, AUC and F1 score with PCA, LDA and Isomap. The results are averaged for PCA, LDA and Isomap over binary and multiclass datasets.

Secondly, the Adaboost model is used with decision tree classifier as a base learner and evaluated for each dataset for Accuracy, AUC and F1 score with PCA, LDA and Isomap. The results are averaged for PCA, LDA and Isomap over binary and multiclass datasets. Due to space constraints, individual results on each data set for bagging and boosting are not shown in the paper.

Next, for the stacking ensemble model, this study applied Random forest (RF), SVM, KNN, Bernoulli’s Naïve Bayes (BNB), Gaussian Naïve Bayes (GNB), Gradient boosting (GBM), MLP and Logistic regression (LR) on each dataset. The data is visualized to see how the predictions from all eight models are different. For that, TSNE technique is used and created scatter plot which shows the predictions of different models. Then created a heat plot to compare the correlation of their prediction. Finally, frequencies of predicted classes are visualized using count plot of all classifiers. Detailed experimentation is carried out to calculate accuracy for eight classifiers of stacking model with PCA, LDA and Isomap. Due to space constraints, individual values are shown here in the paper.

To select the three best performing classifiers among the eight classifiers, accuracy is averaged across all feature reduction techniques for both binary and multiclass datasets. Table 3 shows the average accuracy value of all eight classifiers on binary and multiclass datasets. KNN, SVM and GBM are the three best performer classifiers for both binary and multiclass datasets, selected for further process.

Table 3 Average accuracy value of all eight classifiers used for stacking model for binary and multiclass data

No.	Dataset type	RF	BNB	GNB	MLP	KNN	SVM	GBM	LR
1	Binary class	83.89	75.08	81.69	85.67	88.31	85.88	87.01	77.44
2	Multi class	79.98	72.07	77.91	81.99	86.31	85.66	86.61	77.80

Next, the study builds one level prediction set for stacking classifier. This work created a level-1 train dataset using 5-fold cross-validation. Level-1 test dataset is created by selected models on complete original train dataset and predicted on the test dataset. Finally, LR is trained as a meta-classifier on level-1 train data and predicted on level-1test dataset. Results are obtained by averaging values on each dataset for binary and multiclass data set for Accuracy, AUC and F1 score for PCA, LDA and Isomap.

Next, for the hybrid ensemble model, predictions of bagging, boosting and stacking models were merged and created data for predicted classes. The concept of majority voting is used, that is, if ‘yes’ is predicted more times than ‘no’ then ‘yes’ is selected and vice versa. In binary datasets, it is certain to have a majority vote for either 0 or 1 class. But in multiclass datasets, which have 3 or more classes, 3 ensemble models can predict different classes. In this scenario, there is no clear winner. Therefore, a class that has the highest probability value is selected. Finally, it is tested data on the original test dataset. Table 4 contains Accuracy, AUC and Prediction of hybrid ensemble model with PCA, LDA and Isomap.

Table 4. Accuracy, AUC and F1 score with PCA, LDA and Isomap using hybrid ensemble model

No.	Class types	Dataset	Feature Reduction Technique	Accuracy	AUC	F1 score
1		Electric grid	PCA	91	0.9	0.929
			LDA	98.05	0.977	0.984
			Isomap	91.95	0.909	0.937
2		Extra sensory - B	PCA	96.275	0.963	0.964
			LDA	95.53	0.954	0.957
			Isomap	96.461	0.964	0.966
3		Football sensor	PCA	89.417	0.891	0.879
			LDA	85.714	0.851	0.832
			Isomap	90.476	0.902	0.891
4		Pulsar star	PCA	97.203	0.911	0.886
			LDA	96.685	0.923	0.872
			Isomap	97.203	0.913	0.886
5		EEG signal	PCA	98.338	0.983	0.983
			LDA	54.892	0.548	0.556
			Isomap	94.523	0.945	0.946
6		Power system - A	PCA	93.61	0.906	0.879
			LDA	74.056	0.649	0.486
			Isomap	91.287	0.868	0.83
7		Hand-gesture recognition	PCA	97.076	0.97	0.971
			LDA	66.809	0.667	0.675
			Isomap	96.474	0.964	0.965
8	Binary class	Watch sensor	PCA	99.458	0.994	0.994
			LDA	73.68	0.737	0.729
			Isomap	99.526	0.995	0.995
9		Power system - B	PCA	92.253	0.861	0.815
			LDA	74.647	0.612	0.397
			Isomap	92.857	0.873	0.831
10		Machine sensor	PCA	91.854	0.897	0.875
			LDA	65.301	0.595	0.447
			Isomap	91.384	0.892	0.868
11		Cardiotocography sensor	PCA	100	1	1
			LDA	100	1	1
			Isomap	100	1	1
12		Extra sensory - A	PCA	75.232	0.44	0.752
			LDA	78.584	0.527	0.785
			Isomap	71.322	0.522	0.713
13		Mode detection	PCA	89.228	0.959	0.892
			LDA	71.586	0.925	0.715
			Isomap	85.581	0.948	0.855
14	Multi class	Sky server	PCA	89.4	0.895	0.894
			LDA	94.35	0.948	0.943
			Isomap	84.65	0.854	0.846
15		Movement recognition	PCA	93.22	0.989	0.932
			LDA	84.406	0.95	0.844
			Isomap	92.542	0.985	0.925
16		Air quality sensor	PCA	90.17	1	0.901
			LDA	95.673	1	0.957
			Isomap	90.384	1	0.903
17		Energy prediction	PCA	98.378	0.998	0.983
			LDA	100	1	1

18	Big Sensors	Isomap	75.95	0.929	0.759
		PCA	89.725	0.996	0.897
		LDA	81.803	1	0.818
19	Transport detection	Isomap	83.764	0.993	0.837
		PCA	91.687	0.962	0.916
		LDA	78.71	0.936	0.787
20	Direction sensor	Isomap	89.737	0.959	0.897
		PCA	1189.459	0.88	0.894
		LDA	77.451	0.82	0.774
		Isomap	88.267	0.876	0.882

For measuring the performance of models, the average value of Accuracy, AUC and F1 score of PCA, LDA and Isomap is calculated with all ensemble models, on all binary and multiclass datasets. Table 5 shows the average accuracy rate of PCA, LDA and Isomap with Bagging, Boosting, Stacking and hybrid ensemble model on all binary and multiclass IoT datasets.

Table 5. Average accuracy using PCA, LDA and Isomap with all ensemble models on binary and multiclass data

No	Dataset Type	Feature Reduction Technique	Bagging Model	Boosting Model	Stacking Model	Hybrid Model
1	Binary class	PCA average	93.879	91.185	93.816	94.648
		LDA average	79.117	77.029	81.685	78.536
		Isomap average	93.320	90.542	93.555	94.215
2	Multi class	PCA average	88.963	84.523	91.712	90.649
		LDA average	86.276	82.970	85.865	86.256
		Isomap average	85.586	80.269	86.090	86.219

Table 6 shows the average AUC score of PCA, LDA and Isomap with Bagging, Boosting, Stacking and hybrid ensemble model on binary and multiclass data for IoT dataset.

Table 6 Average AUC using PCA, LDA and Isomap with all ensemble models on binary and multiclass data

No.	Dataset Type	Feature Reduction Technique	Bagging Model	Boosting Model	Stacking Model	Hybrid Model
1	Binary class	PCA average	0.915	0.895	0.914	0.927
		LDA average	0.757	0.741	0.768	0.751
		Isomap average	0.909	0.891	0.912	0.922
2	Multi class	PCA average	0.903	0.897	0.924	0.911
		LDA average	0.907	0.904	0.950	0.910
		Isomap average	0.871	0.855	0.916	0.906

Table 7 shows the average F1 score of PCA, LDA and Isomap with Bagging, Boosting, Stacking and hybrid ensemble model on all binary and multiclass IoT datasets.

Table 7. Average F1 Score using PCA, LDA and Isomap with all ensemble models on binary and multiclass datasets.

No.	Dataset Type	Feature Reduction Technique	Bagging Model	Boosting Model	Stacking Model	Hybrid Model
1	Binary class	PCA average	0.902	0.871	0.896	0.917
		LDA average	0.701	0.683	0.698	0.693
		Isomap average	0.896	0.867	0.891	0.911
2	Multi class	PCA average	0.889	0.844	0.916	0.906
		LDA average	0.861	0.829	0.858	0.862
		Isomap average	0.860	0.802	0.860	0.861

For better understanding, the average value of Accuracy, AUC and F1 score is visualized. The graphical representation shows a visualization of Table 5. Figure 2 describes the average accuracy rate of PCA, LDA and Isomap with Bagging, Boosting, Stacking and Hybrid ensemble models on all binary and multiclass IoT datasets.

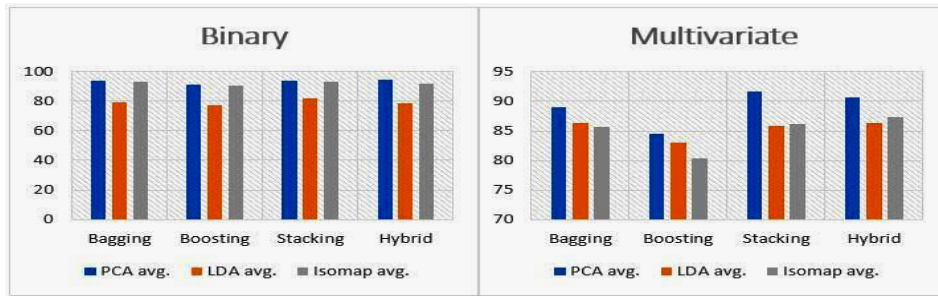


Figure 2. Average Accuracy for PCA, LDA and Isomap with all ensemble models on all binary and multiclass data.

Below graphical representation shows a visualization of Table 6. Figure 3 describes the average AUC rate of PCA, LDA and Isomap with Bagging, Boosting, Stacking and Hybrid ensemble models on all binary and multiclass IoT datasets.

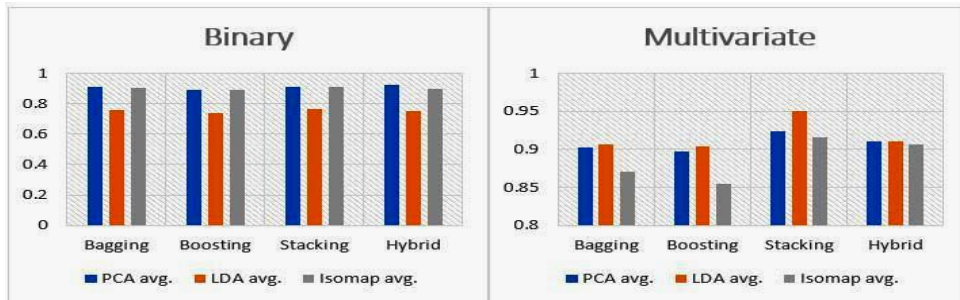


Figure 3. Average AUC for PCA, LDA and Isomap with all ensemble models binary and multiclass data.

Below graphical representation shows a visualization of Table 7. Figure 4 describes the average F1 score of PCA, LDA and Isomap with Bagging, Boosting, Stacking and Hybrid ensemble models on binary and multiclass IoT datasets.

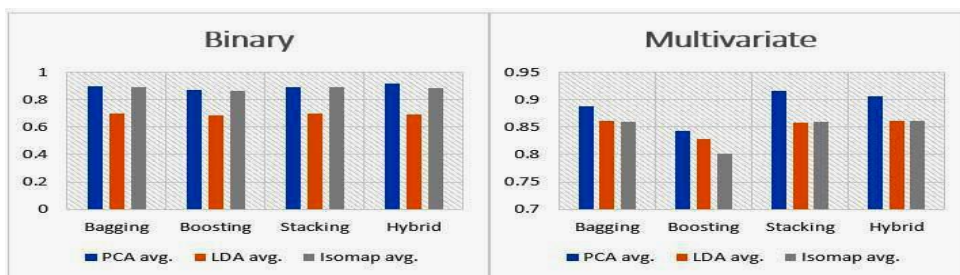


Figure 4. Average F1-score for PCA, LDA and Isomap with all ensemble models on all binary and multiclass data.

6. Observations and Conclusion

6.1 Observations

It is observed from table 5 , for binary datasets, hybrid with PCA model achieved highest accuracy score with 94.468% and boosting with LDA model achieved lowest accuracy score with 77.029%. LDA average accuracy scores of all ensemble models are very less as compared to PCA and Isomap. For multiclass dataset, stacking with PCA model achieved top score with 91.712% while boosting with Isomap model get a low score with 80.269%.

From table 6, it is seen that, for binary datasets, the hybrid model with PCA obtained the best average AUC score of 0.927 and boosting with LDA earned the lowest AUC average score of 0.683. Compared to LDA and

Isomap average AUC scores of all ensemble models, PCA average scores of all ensemble models are relatively high. For multiclass datasets, Stacking with LDA performs the best and with 0.950 average AUC score while boosting with Isomap model received a low average AUC score of 0.855.

From table 7, for binary datasets, the hybrid model with PCA performed excellently and obtained a 0.917 average F1 score while boosting with LDA model performed very poorly and obtained an average F1 score of 0.683. For multiclass datasets, stacking model with PCA received the highest mean F1 score of 0.916 and the boosting with Isomap model obtained the lowest mean F1 score of 0.802.

6.2 Conclusions

This comparative study investigated the possibility of applying bagging, boosting, stacking and hybrid ensemble algorithms with PCA, LDA and Isomap to improve the performance on IoT sensor datasets. In both binary and multiclass cases, PCA works perfectly to all ensemble models compared to LDA and Isomap. For binary datasets, Hybrid with PCA works the best against other models. Boosting with LDA performed ineffectively compared to other models. For multiclass datasets, Stacking with PCA performed better than other models in question and the close runner-up is Hybrid with PCA. Boosting with Isomap worked very poorly in case multiclass datasets. Bagging performed average in binary as well as multiclass datasets.

References

- [1]. Aha, D.W., Kibler, D. & Albert, M.K., (1991). Instance-based learning algorithms. *Machine learning*, 6(1), 37-66.
- [2]. Alexandropoulos, S.A.N., Aridas, C.K., Kotsiantis, S.B. & Vrahatis, M.N. (2019). Stacking strong ensembles of classifier. *International Conference on Artificial Intelligence Applications and Innovations*, 545-556.
- [3]. Amasyali, M. F, and Ersoy, O. (2008). Cline: A new decision-tree family', *IEEE Transactions on Neural Networks* 19(2), 356-363.
- [4]. Asuncion, A. & Newman, D. (2007). UCI machine learning repository.
- [5]. Atzori, L., Iera, A. and Morabito, G. (2010). The internet of things: A survey. *Computer networks*, 54 (15), 2787-2805.
- [6]. Boriah, S., Chandola, V. & Kumar, V. (2008). Similarity measures for categorical data: A comparative evaluation. In *Proceedings of the SIAM international conference on data mining*, 243-254.
- [7]. Breiman, L. (1996). Bagging predictors. *Machine learning* 24(2), 123-140.
- [8]. Freund, Y. & Schapire, R.E. (1996). Experiments with a new boosting algorithm. *International conference for machine learning*, 96, 148-156.
- [9]. Gay, D.M. & Welsch, R.E. (1988), Maximum likelihood and quasi-likelihood for nonlinear exponential family regression models. *Journal of the American Statistical Association* 83(404), 990-998.
- [10]. Graczyk, M., Lasota, T., Trawiński, B. & Trawiński, K. (2010). Comparison of bagging, boosting and stacking ensembles applied to real estate appraisal. In *Asian conference on intelligent information and database systems*, Springer, Berlin, Heidelberg, 340-350
- [11]. James, G., Witten, D., Hastie, T. & Tibshirani, R. (2013). An introduction to statistical learning. Vol. 112. Springer. New York.
- [12]. Junior, J. J. A. M., Freitas, M. L., Siqueira, H. V., Lazzaretti, A. E., Pichorim, S. F., & Stevan Jr, S. L. (2020). Feature selection and dimensionality reduction: An extensive comparison in hand gesture classification by sEMG in eight channels armband approach. *Biomedical Signal Processing and Control*, 59, 101920.
- [13]. Mason, L., Baxter, J., Bartlett, P. & Frean, M. (2000). Boosting algorithms as gradient descent, *Advances in neural information processing systems* 12, 512-518.
- [14]. Meidan, Y., Bohadana, M., Shabtai, A., Ochoa, M., Tippenhauer, N.O., Guarnizo, J.D. & Elovici, Y. (2017). Detection of unauthorized IoT devices using machine learning techniques. *arXiv preprint arXiv:1709.04647*.
- [15]. Narassiguin, A., Bibimoune, M., Elghazel, H. & Aussem, A. (2016). An extensive empirical comparison of ensemble learning methods for binary classification. *Pattern Analysis and Applications* 19(4), 1093-1128.
- [16]. Pandey, M. & Taruna, S. (2014). A comparative study of ensemble methods for students' performance modelling. *International Journal of Computer Applications* 103(8), 26- 32.
- [17]. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *The Journal of machine Learning research* 12, 2825- 2830.
- [18]. Ribeiro, M. H. D. M., & dos Santos Coelho, L. (2020). Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series. *Applied Soft Computing*, 86, 105837.

- [19]. Rojarath, A., Songpan, W. & Pong-inwong, C. (2016). Improved ensemble learning for classification techniques based on majority voting. *In 7th IEEE international conference on software engineering and service science (ICSESS)*, 107-110.
- [20]. Singh, S. and Singh, N. (2015). Internet of Things (IoT): Security challenges, business opportunities & reference architecture for E-commerce, *International Conference on Green Computing and Internet of Things (ICGCIoT)*, 1577-1581. IEEE.
- [21]. Suganthi, M. & Karunakaran, V. (2019). Instance selection and feature extraction using cuttlefish optimization algorithm and principal component analysis using decision tree. *Cluster Computing* 22(1), 89-101.
- [22]. Syarif, I., Zaluska, E., Prugel-Bennett, A. & Wills, G. (2012). Application of bagging, boosting and stacking to intrusion detection. *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, 593- 602.
- [23]. Taşer, P.Y., Birant, K.U. & Birant, D. (2019). Comparison of Ensemble-Based Multiple Instance Learning Approaches, *In IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*, 1-5.
- [24]. Tounsi, Y., Hassouni, L. & Anoun, H. (2018) .An Enhanced Comparative Assessment of Ensemble Learning for Credit Scoring. *International Journal of Machine Learning and Computing* 8(5),15.
- [25]. Tyagi, K. & Manry, M. (2019). Multi-step training of a generalized linear classifier, *Neural Processing Letters* 50(2), 1341-1360.
- [26]. Ularu, E.G., Puican, F.C., Suci, G., Vulpe, A. and Todoran, G. (2013). Mobile Computing and Cloud maturity-Introducing Machine Learning for ERP Configuration Automation. *Informatica Economica*, 17(1), 40-52
- [27]. Vijai, P . (2018). Performance comparison of feature reduction techniques in-terms of compactness, computation time and accuracy. *IEEE Symposium Series on Computational Intelligence (SSCI)*, 374-380.
- [28]. Wan, S., & Yang, H. (2013). Comparison among methods of ensemble learning. *International Symposium on Biometrics and Security Technologies*, 286-290.
- [29]. Wang, G., Hao, J., Ma, J. & Jiang, H. (2011). A comparative assessment of ensemble learning for credit scoring, *Expert systems with applications* 38(1), 223 -230
- [30]. Wilamowski, B.M. (2009). Neural network architectures and learning algorithms, *IEEE Industrial Electronics Magazine* 3(4), 56-63.
- [31]. Wilson, D.R. & Martinez, T.R., (1997). Improved heterogeneous distance functions. *Journal of artificial intelligence research*, 6, 1-34.
- [32]. Wolpert, D. H. (1992). Stacked generalization. *Neural networks* 5(2), 241-259.
- [33]. Ye, R. & Suganthan, P.N. (2012). Empirical comparison of bagging-based ensemble classifiers. *International Conference on Information Fusion*. 917-924.
- [34]. Yu, Z., Wang, D., You, J., Wong, H.S., Wu, S., Zhang, J. and Han, G. (2016). Progressive subspace ensemble learning. *Pattern Recognition* 60, 692-705.