IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON INTELLIGENT BIG DATA ANALYTICS FOR INTERNET OF THINGS, SERVICES AND PEOPLE

Received April 12, 2021, accepted April 25, 2021, date of publication April 28, 2021, date of current version May 10, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3076264

ElStream: An Ensemble Learning Approach for Concept Drift Detection in Dynamic Social Big Data Stream Learning

AHMAD ABBASI[1], ABDUL REHMAN JAVED[2], CHINMAY CHAKRABORTY[3], JAMEL NEBHEN[4], WISHA ZEHRA[1], AND ZUNERA JALIL[2], (Member, IEEE)

[1]Faculty of Computing and AI, Air University, Islamabad 44000, Pakistan
[2]Department of Cyber Security, Air University, Islamabad 44000, Pakistan
[3]Department of Electronics and Communication Engineering, Birla Institute of Technology, Mesra, Ranchi 835215, India
[4]College of Computer Science and Engineering, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

Corresponding author: Abdul Rehman Javed (abdulrehman.cs@au.edu.pk)

ABSTRACT With the rapid increase in communication technologies and smart devices, an enormous surge in data traffic has been observed. A huge amount of data gets generated every second by different applications, users, and devices. This rapid generation of data has created the need for solutions to analyze the change in data over time in unforeseen ways despite resource constraints. These unforeseeable changes in the underlying distribution of streaming data over time are identified as concept drifts. This paper presents a novel approach named *ElStream* that detects concept drift using ensemble and conventional machine learning techniques using both real and artificial data. *ElStream* utilizes the majority voting technique making only optimum classifier to vote for decision. Experiments were conducted to evaluate the performance of the proposed approach. According to experimental analysis, the ensemble learning approach provides a consistent performance for both artificial and real-world data sets. Experiments prove that the *ElStream* provides better accuracy of 12.49%, 11.98%, 10.06%, 1.2%, and 0.33% for PokerHand, LED, Random RBF, Electricity, and SEA dataset respectively, which is better as compared to previous state-of-the-art studies and conventional machine learning algorithms.

INDEX TERMS Internet of Things, big data, smart concept drift, social data, online learning, ensemble learning.

I. INTRODUCTION

Big data has received an enormous amount of attention during the last decade. The main reason for this attention is that every organization generates data and Big data promises insights that can help an organization grow in a way that was not possible before. Big data enables growth in every industry like banking, healthcare [1]–[3], food [4], manufacturing and consumers [5]. Data extracted from using social media platforms alone can help to perform various types of analysis [6]. Big data analytics offers cost reduction, faster decision-making, innovation in new products, and many other advantages. While big data provides invaluable insights, the fact that this data is often in the form of continuous streams and the volume and the velocity with which the data is streamed makes it challenging to implement in real-life scenarios [7].

Due to the complexity of big data, the traditional approach to data analysis cannot be utilized, and instead, Machine learning approaches enable a system to identify patterns and learn without being programmed to perform those specific tasks. Machine Learning models thrive on big datasets. The bigger the dataset for training a machine learning model, the better the performance of that model [8]. However, the velocity and volume at which the data is streaming arise memory storage issues, and the typical offline approach where the prediction is made by learning the complete training dataset at once becomes impossible. The online learning approach embraces the fact that data changes from second to second and that predictions must be made before seeing the entire data. Online learning takes a 'learn-as-you-go approach and thus enables our model to learn one instance at a time,

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Khurram Khan.

therefore not requiring entire data to be held in memory [7]. As the model is learning continuously, it can remedy the hurdle of concept drift.

One main challenge when dealing with constantly streaming big data is the evolution of online stream data distribution, a.k.a concept drift. Concept drift occurs due to the dynamic behavior of network activities. We cannot train a model once and use it for new constantly streaming data due to concept drift. Ensemble learning has proved to be an effective solution as combining the effect of multiple classifiers leads to enhanced predictive power, more efficient drift handling [9] and are comparatively easy to deploy in real-world applications [10]. Existing studies [9], [11] have used various techniques and have tried assigning weights or thresholds to each classifier but have failed to deliver competent performance to detect concept drift.

To effectively and efficiently detect concept drift by addressing the above-explained limitations, this paper makes the following contributions:

- We propose an effective ensemble learning approach named *ElStream* to detect concept drift in the online streaming data comprising distinct classifiers to participate in the voting-based decision. A classifier can only assign a vote to the input data streams when the confidence level has passed a certain threshold.
- Evaluate the effectiveness of the proposed *ElStream* technique by performing a comparative analysis of *ElStream* with conventional machine learning techniques: Random Forest (RF), K-Nearest Neighbour (KNN), Extreme Gradient Boosting (XGB), and Multi-layer Perceptron (MLP) and State-of-the-art studies.
- Experiments demonstrate that the *ElStream* provides better accuracy of 12.49%, 11.98%, 10.06%, 1.2%, and 0.33% for PokerHand, LED, Random RBF, Electricity, and SEA dataset respectively as compared to previous state-of-the-art studies and conventional machine learning algorithms.

Table 1 presents the notation used in the entire paper. The rest of the paper is organized as follows. Section II gives a brief review of the state-of-the-art related work. The details of the dataset are provided in Section III. Section IV describes the proposed approach. The evaluation criteria and results of the proposed methods are presented in Section V. Section VI provides a discussion on the experimentation results. Finally, Section VII concludes along with directions for future work.

TABLE 1. Key notations.

Notation	Description
RF	Random Forest
KNN	K-Nearest Neighbor
XGB/XGBoost	Extreme Gradient Boosting
MLP	Multilayer perceptron
EoBag	efficient online bagging
$EoBag_m$	m represents moderate
$EoBag_f$	F represents fast
$EoBag_{nbk}$	nbk represents no backup classifier
$EoBag_{std}$	std represents standard
$EoBag_{maj}$	maj represents majority

II. RELATED WORK

Due to the magnitude of the impact that big data analytics have, much research has been conducted over the past decade. The authors in [12] performed experiments to check the effect of dimensionality reduction on big data and found that ML algorithms with principle component analysis (PCA) work better when the dimensionality of data is high. In [13], the authors conducted an extensive survey and concluded that to handle the dimension and velocity of big data and its security, blockchain can be used. The authors in [14] used spiked neural networks for online stream learning and found that they perform well in drift situations. The authors in [15] proposed a class-based ensemble that updates the base learner for each class as an instance arrives. They found that the focus on evolved classes may damage the results for non-evolved classes. The authors in [16] assigned higher weights to new instances for faster detection of concept drifts. In [17], the authors introduced the Fast Hoeffding Drift Detection Method that uses a sliding window and Hoeffding's inequality which enables the detection of drifts with a shorter delay. In [18] the authors contemplated using historical drift trends to predict the occurrence of future drifts.

The authors in [9] assigned dynamic threshold to classifiers in the ensemble, and only classifiers that cross that threshold are allowed to contribute to the prediction. This exploits the diversity of classifiers and increased robustness. In [19] the authors proposed an Evolving Fuzzy System (EFS) with self-learning thresholds that adjust the speed of evolving based on the relationship between overfitting, underfitting, training error, and testing error. The authors in [20] used random re-sampling and heterogeneous classifiers in the ensemble to improve the classification performance and found it achieves better generalization performance. In [21] the authors proposed a Heterogeneous Dynamic Weighted Majority (HDWM) that uses learners of various types to maintain ensemble heterogeneity, overcoming problems of existing dynamic ensembles that may undergo loss of diversity due to the exclusion of base learners. The authors in [22] proposed a framework that dynamically assigns weights to each classifier's vote in the ensemble and found it competitive while giving a faster performance. The authors in [23] used random projection and Naïve Bayes classifiers in the ensemble for classification. The authors in [24] proposed an online ensemble that determines the block size dynamically to capture concept drifts promptly. In [11] the authors used the online bagging ensemble method that uses the online re-sampling method and robust coding method for big data stream learning.

III. NETWORK MODEL, DATASET AND PRELIMINARIES

In this paper, We use three real-world datasets and four synthetic datasets for experimental analysis. The experiments are implemented in Python using Google Colab. Three datasets

consist of binary class labels, and four datasets consist of multi-class labels. The real dataset is obtained through The University of California Irvine (UCI) machine learning repository [25] to evaluate the classification performance of the data stream classifier. TABLE 2 presents an overview of the data sets used in this work.

A. ARTIFICIAL DATASETS

Artificial datasets' primary purpose is to be versatile and robust enough to be useful for training machine learning models. Artificial datasets bear a low cost of storage and transmission and provide an advantage of knowing where specifically concept drift happens and the type of that drift. **Hyperplane**: is utilized in many stream classification experiments over the past years. Hyperplane is generally used to produce data streams with concept drift. It is a binary-class dataset and in a d-dimensional space. We set the hyperplane generator to create a dataset where the set of points y that satisfy equation (1).

$$\sum_{i=1}^{d} \alpha_i y_i = w \tag{1}$$

where $y_0, y_1 \ldots y_i$ is the i^{th} coordinate of y. When

$$\sum_{i=1}^{d} \alpha_i y_i >= w \tag{2}$$

the instances are labeled positive otherwise negative. In this work, the hyperplane generator is set to create a dataset comprising 10,000 instances described by 10 features with gradual drifts by the modification weight w_i changing by 0.001 with each instance with 5% noise to streams.

1) SEA

dataset consists of three attributes in which all the attributes of the dataset have values between 0 to 10, but only two attributes are relevant. The points of the dataset belong to one of the two possible decision classes. The SEA generator is utilized to produce a Sea dataset with abrupt concept drifts. Every concept is the aggregate of two functions. If $f_1 + f_2 \leq \theta$ then data point belongs to class 1. The first two attributes are represented by f_1 and f_2 where θ is a threshold value between the two class labels. The threshold values are 9, 8, 7, and 9.5. Finally, We have 2,500 instances with sudden drifts and 10% class noise.

2) LED

is a popular synthetic dataset. The objective of this dataset is to foretell the next digit on a seven-segment LED display. The chance of being displayed for every digit is 10%. LED dataset consists of a stream of 24 binary features, 17 of which are irrelevant. Concept drift is generated by interchanging 7 class relevant attributes. This work generates a stream of 100,000 instances, where concept drift occurs at every 25,000 instances. To produce gradual concept drifts, a transition length of w = 500 is set. 10% noise may also be inserted into each data stream.

TABLE 2. Overview of the datasets.

Type	Dataset	No. of Features	No. of Classes
Artificial Dataset	LED	24	10
	SEA	3	2
	Hyperplane	10	2
	Random RBF	10	5
Real Dataset	Covertype	54	7
	PokerHand	10	10
	Electricity	8	2

3) RANDOM RBF

The Radial Basis Function generator is used to create a user-specified number of drifting centroids. Each drifting centroid is defined by the number of class labels, position, weight, and standard deviation. The random Available at: https://www.win.tue.nl/~mpechen/data/DriftSets/

positions, weights, and standard deviations are moved with constant speed v in d-dimensional space with the parametrization as 10 dimensions, 50 Gaussians, 5 classes, and v = 0.001. We have made this synthetic dataset ourselves which contains 500,000 instances described by 10 features and 5 class labels.

B. REAL DATASETS

When using real-world datasets, it is impossible to detect when drift occurs and to identify the drift. The real dataset is used with the artificial dataset to analyze if the proposed approach works well even when the drift will occur unknown. The real-world datasets employed in this series of experiments can be obtained through The University of California Irvine's (UCI) Machine Learning Repository [25] or can be obtained at https://github.com/alipsgh/data_streams, https://moa.cms.waikato.ac.nz/datasets/2013/. for evaluating machine learning techniques.

1) COVERTYPE

dataset contains 54 features that describe possible forest cover types. It has 581,012 instances, which describe 7 forest cover types for cells of 30 × 30 meters, obtained from the US Forest Service (USFS) Resource Information System (RIS). It also has been used in [16].

2) ELECTRICITY

consists of 45,312 instances, each defined by 8 input attributes. The classification task predicts whether the cost of electricity will grow or decline in the Australian New South Wales Electricity Market. This dataset was collected through successive measurements every half an hour for two years from 1996 to 1998. A class label identifies the change in electricity price at a specified time, where the price is higher or lower than the moving average of the last 24 hours.

3) PokerHand

dataset signifies the problem of identifying the hand in a Poker game. It contains 1,025,010 instances representing all the possible poker hands, where each instance depicts a hand consisting of five cards pulled from a regular set of 52 cards. Each card in hand is represented by two attributes (suit and rank). Ten features are used for describing each hand.

IV. METHODOLOGY

FIGURE 1 demonstrate the proposed model of this study, divided into two major phases: pre-processing and the selection of classification models. First of all, the dataset is divided into two sets for training and testing purposes. Then We handle the different aspects of the dataset such as small, large, noisy datasets, overfitting, and class imbalance. We have different concept drift types and different noise levels in our datasets. Another problem is that there are some relevant and irrelevant attributes to the class in our datasets. To address these issues with the dataset, We proposed an Algorithm named ElStream that uses predefined features. This proposed ElStream approach performs competently to detects concept drifts, new class detection, feature drifts, and it classifies the data with better accuracy.

A. DATA PRE-PROCESSING

The experiments performed in this work are implemented in Python using Google Colab to evaluate the data stream's performance. Before training the classifiers, pre-processing techniques are applied that handle missing values and remove outliers and noise. In this study, We used two normalization methods, robust scaler, and standard scaler. Robust scaler is robust to outliers in the sense that it uses the interquartile range. This scaler removes the median and scales the data according to the interquartile range. The IQR is the range between the 1st quartile (Q_1) and the 3rd quartile (Q_3). Robust scaler can be defined as in equation (3), where X is standardized form of x_i and Q_1 is first quartile, Q_3 is third quartile. Arbitrary scaling is performed to normalize the data within a predefined range using a standard scaler. Standard scaler works by rescaling features to be approximately standard normally distributed. To achieve this, we have used standardization to transform the data such that it has a mean of 0 and a standard deviation of 1. Standard scaler can be defined as in equation (4), where Z is our standardized form of x_i.

$$X = \frac{x_i - Q_1}{Q_3 - Q_1} \tag{3}$$

$$Z = \frac{(x_i - mean)}{standard\_deviation} \tag{4}$$

B. ENSEMBLE CLASSIFIERS

Ensemble learning has emerged to be a popular technique among researchers for several different ML enigmas [26]–[29]. Each dataset contains different drifts, with different levels of severity and speed, so an ensemble learning algorithm is used which explicitly detects different types of drifts and handles data classification accordingly. The idea behind Ensemble learning methods is to combine multiple machine learning classifiers and use various voting mechanisms to achieve better performances [30].

In a majority voting scheme, whenever a new instance arrives, each classifier in the ensemble predicts a class label, and then the class label which most of the classifiers predictor, in other words, has the most votes is assigned as the class label of that instance. These methods are helpful to achieve better generalization performance than the traditional single learning approach. In this study, the proposed approach *ElStream* aggregates the predictions of the multiple classifiers and outputs the results on the basis of the majority voting mechanism. The best results are presented after fine-tuning each classifier. The majority of votes are collected using the approach in equation (5).

$$\widetilde{y} = argmax(N_c(y_t^1), N_c(y_t^2), \ldots, N_c(y_t^n)) \tag{5}$$

In equation (5) $N_c(y_t)$ represents the class with the highest number of votes. Different classification models such as Random Forest (RF), XGBoost, and Multilayer Perceptron (MLP) are used in the ensemble.

1) RANDOM FOREST

is an ensemble learning method used for classification, regression, and other tasks. It is a machine learning ensemble model that creates several trees to execute a classification [31]. By using the ensemble method, the classification tree becomes more precise than an individual member. Random forest is often used to handle complex data, unlike traditional classifiers. The Random Forest (RF) is a classification algorithm consisting of many decision trees, and each decision tree votes for the best target label. The prediction is made based on the majority voting strategy. The parametrization for our ensemble method are: n-estimators = 100, bootstrap = True, criterion = Gini, min-samples-leaf = 1, min-samples-split = 2, random-state = 0.

2) EXTREME GRADIENT BOOSTING

is a boosting technique that incorporates efficiency and memory resources. This ensemble classification model is using XGBoost to improve the performance of classification. XGBoost classifier is used to achieve better classification accuracy. XGBoost is a scalable machine learning model that produces a prediction in the form of a boosting ensemble of weak classification trees by a gradient descent that optimizes the loss function [32]. Gradient boosting is the unique model of XGBoost by combining weak base learning models into a stronger learner in an iterative model. At every iteration of gradient boosting, the residual error is used to correct the previous predictor to optimize the loss function. Regularization is added to lose function so that objective function can be established in XGBoost measuring the model performance, which is defined by

$$J(\Theta) = L(\Theta) + \Omega(\Theta). \tag{6}$$

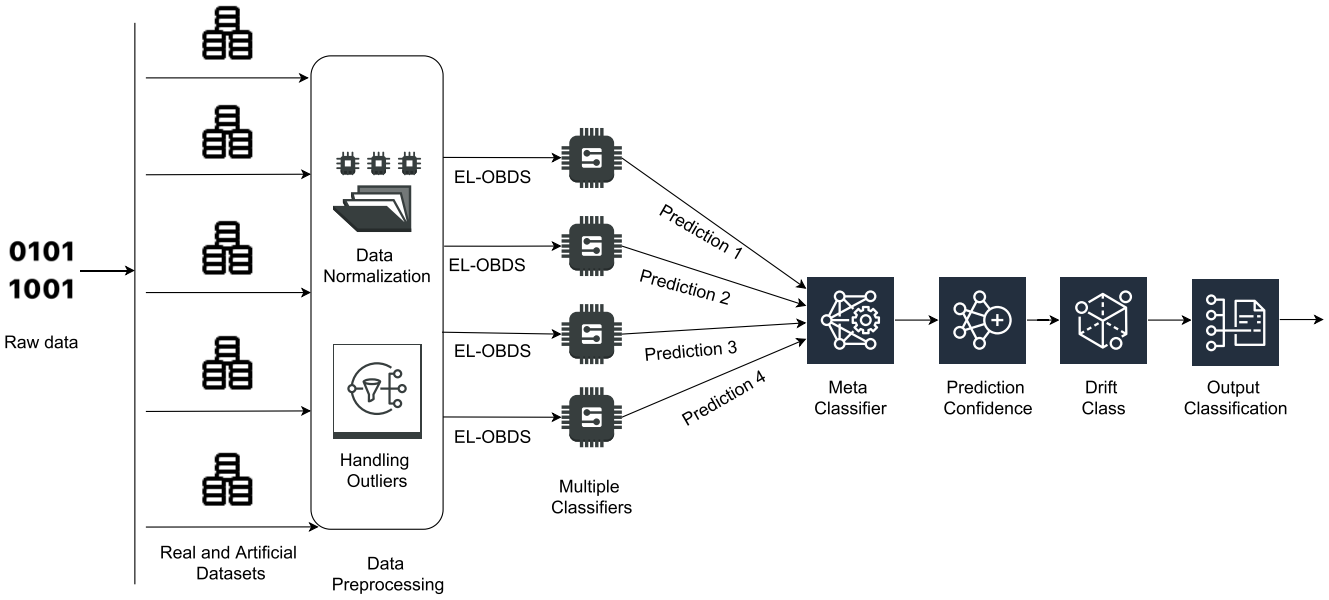


FIGURE 1. Graphical representation of proposed approach for the detection of concept drift in the online data stream classification.

TABLE 3. Computing environment.

Parameters	Values
Framework	Google Colab
Operating System	Windows 10 Professional 1909
CPU	Intel Xeon Processor, two cores@2.30 GHz
GPU	Up to Tesla K80 with 12 GB GDDR5 VRAM
RAM	13GB
Programming Language	Python
Python Version	3.6.9

In equation (6), Θ shows the parameters trained from given data. L is the training Loss function, and ω is the regularization term that measures the model's complexity. In this study, the parameters of the XGBoost algorithm are: booster = gbtree, eta = 0.3, min-child-weight = 1, max-depth = 6 and scale-pos-weight = 1.

3) MULTI-LAYER PERCEPTRON (MLP)

is a part of a feed-forward artificial neural network (ANN). MLP is suitable for classification prediction problems. MLP uses back probation, which is a supervised learning technique, for training. The MLP model consists of at least three layers of nodes: an input layer, a hidden layer of computation nodes, and an output layer of computation nodes. The input layer nodes are the feature values of an attribute, and the output layer nodes are discriminators between the class of the attribute and those of all other attributes. MLP utilizes a supervised learning algorithm that uses a function $f(X) : R^m \rightarrow R^o$ for training on dataset. Here o is the number of dimensions for output, and m is the number of input dimensions. We have a set of features $X = x_1, x_2 \ldots .x_m$ and a target y. Each node is a neuron that uses a nonlinear activation function approximator for either classification or regression. In this study, the parameters of MLP classifier are: hidden-layer-size staple:length = 100, activation = relu, solver = adam, alpha = 0.0001, max-iter = 200, shuffle = True and verbose = False.

C. PROPOSED ElStream ALGORITHM

Let D represents the dataset containing attributes $I = i_1, i_2 \ldots .i_n$. PC represents each model's predicted confidence, and TC represents the threshold confidence set to evaluate the PC of each model. TL represents the labels of the target class to be predicted by each classifier. NTL denotes the total target classes. Let IC represent the instance counter. The attribute class count that is incremented when a classifier votes for the class label is denoted by ICC. Each attribute in I is an input to the prediction model and is appended in IC. Then We evaluate the confidence of TL and ICC. Each observation gets a vote from each classifier. The threshold value of 80% is set to compare the confidences. The attribute values must have to achieve a threshold of 80% or more to fall in a particular class. If the given criteria do not meet, then another attribute is added until the requirement is not fulfilled. If more than one class participates in the classification result and has the same number of votes, anyone can be randomly selected. If CL is greater than the threshold value of 80%, then the target class is considered a label of that instance.

V. EVALUATION AND RESULTS

The objective of this study is to evaluate the classification performance of a data stream classifier. Experiments are performed using multiple machine learning algorithms, i.e., Random Forest, KNN, XGB, MLP, and then using our proposed Ensemble (*ElStream*) method on real and synthetic

Algorithm 1 Ensemble of Distinct Classifiers

Input: *Reading* ← *Big Data Streams*
Output: Concept Drift Detection
Evaluation Measures: Accuracy, F-Score, Recall, Precision

```
i ← [Reading] {Current Instance}
IC ← [] {Instances Count}
PC ← [] {Predicted Confidence}
TC ← 80 {Threshold Confidence}
TL ← [N] {Labels of Target Class}
CL ← NULL {Confidence Level of i}
NTL ← len(TL) {Total target Class Labels}
CI ← NULL {Class of Instance i}
ICC ← NULL {Count of Instance Class}
for each i in I do
    IC ← IC + +
    CI ← classify(i)
    ICC[TL] ← IC + +
    (CL, NTL) ←highest_confidence_level (ICC, TL)
    if (CL ≥ TC) then
        ICC ← TL
    end if
end for
return max(ICC)
```

datasets. After experimentation, the results are compared with a state-of-the-art method 5. The performance evaluation metrics include (i) accuracy, (ii) precision, (iii) recall, and (iv) f-score. These standard performance metrics are primarily chosen to testify to the models' capability to generate the best classification performance for data stream classification. This study uses three real datasets and four synthetic datasets for experimental analysis. The computing environment used for experiments present in TABLE 3.

Evaluation measures are used to measure the quality of the machine learning model. Evaluation metrics are critical to analyzing the performance of a machine learning model. We split the data for training and testing to conduct the experimental evaluation, where 80% of data is used for training and 20% for testing. The performance evaluation measures include accuracy, precision, recall, and f-score to evaluate the given proposed method's performance in TABLE 4.

A. RESULTS

This work uses five different machine learning algorithms on both real and artificial datasets and analyzes their performance using the evaluation metrics defined. The ensemble learning approach *ElStream* is also used to classify data into its respective categories and detect concept drifts.

1) COVERTYPE

TABLE 4 demonstrates the results achieved on Cover type dataset. KNN model gives the best accuracy of 96.55%. Other conventional techniques: Random Forest, XGB classifier, MLP classifier, and Ensemble method achieves the accuracy of 95.85%, 86.87%, 87.30%, and 92.22% respectively. XGB classifier obtained the least accuracy because XGB is slightly weak when it has many categorical variables. Random Forest, MLP, and Ensemble classifier perform quite well on Cover type dataset. KNN shows a proficient gain of 97.31%, 97.75%, and 97.82% in terms of precision, recall, and f-score compared to other models. FIGURE 2a shows the confusion matrix of the KNN. Only 0.0344% labels are misclassified, and 0.9655% labels are accurately classified.

A ROC curve (receiver operating characteristic curve) showing the performance of a KNN classification model is shown in FIGURE 3a. ROC curve plots two parameters, True Positive Rate and False Positive Rate. ROC curve shows the interval in which the true population Area under the ROC curve lies with 95% confidence. The ROC curve value 95% denotes the KNN classifier as an excellent classifier.

2) PokerHand

The results of the PokerHand dataset are shown in TABLE 4. Ensemble classifier delivers the best accuracy of 99.99%. KNN, Random Forest, XGB classifier, and MLP classifier achieve the accuracy of 99.94%, 99.96%, 99.99%, and 99.99%, respectively. All the classifiers perform considerably well on Poker-Hand because of the rules for this dataset. Every one of these rules acknowledges an entire class with 100% confidence. It does not misclassify any value belonging to other classes. These rules exist for classes: Royal flush, Straight flush, Four of a kind, Full house, Flush, Straight, Three of a kind, Two pairs, One pair, and Nothing. Ensemble classifier shows a proficient gain of 99.95%, 99.97%, and 99.98%, in terms of precision, recall, and f-score compared to other models.

FIGURE 2b depicts the confusion matrix to evaluate the quality of an ensemble classifier's output on the Pokerhand data set. The diagonal values signify the number of points where the predicted label is equal to the true label. The classifier mislabels the off-diagonal values. The higher the diagonal values, shows better and correct predictions. We plot a ROC curve that shows the performance of an ensemble classification model. In FIGURE 3b ROC curve that passes through the upper left corner shows 100% sensitivity and 100% specificity. The ROC curve of 100% deems the ensemble classifier as the best classifier.

3) ELECTRICITY

The average accuracy of an Electricity dataset is 87.85, in which the XGB classifier gets the best accuracy of 91.58%. All other Classification models such as KNN, Random Forest, MLP classifier, and ensemble classifier achieve the accuracy of 84.74%, 91.22%, 81.06%, and 90.65%, respectively. TABLE 4 shows the classification accuracy of all the classifiers. MLP uses a supervised learning technique, and it is sensitive to feature scaling. That is why MLP obtained the least accuracy. The performance evaluation measures include recall, f-score, and precision. XGB classifier gains

TABLE 4. Classifier performance to detect concept drift.

Classifiers	Results	Covertype	PokerHand	Electricity	Sea	Hyperplane	Random rbf	LED
RF	Accuracy	95.85	99.96	91.22	88.80	87.45	99.94	85.50
	Precision	96.25	68.76	89.64	86.35	86.43	99.25	85.44
	Recall	96.32	68.73	89.63	86.33	86.42	99.25	85.44
	F1-score	96.15	68.62	89.74	86.42	86.52	99.24	85.43
KNN	Accuracy	96.55	99.94	84.74	88.80	88 .00	91.18	84.91
	Precision	97.31	99.92	85.58	89.52	88.54	91.72	85.01
	Recall	97.75	99.92	85.57	89.51	88.53	91.73	85.03
	F1-score	**97.82**	99.93	85.54	89.54	88.51	91.64	85.04
XGB	Accuracy	86.87	99.99	91.58	88.20	88.50	99.84	85.67
	Precision	87.41	99.95	92.15	88.53	89.51	99.53	86.53
	Recall	87.43	99.95	92.16	88.51	89.35	99.62	86.52
	F1-score	87.42	99.96	91.78	88.54	89.43	99.61	86.26
MLP	Accuracy	87.30	99.99	81.06	91.20	90.14	99.21	85.74
	Precision	87.52	99.96	81.01	91.03	90.23	99.31	86.03
	Recall	87.36	99.96	81.03	91.03	90.28	99.31	86.12
	F1-score	87.42	99.95	81.05	90.07	90.15	99.30	86.22
Ensemble	Accuracy	92.22	99.99	90.65	90.00	90.10	99.94	85.92
	Precision	92.16	99.95	91.31	90.64	90.53	99.65	86.89
	Recall	92.75	99.97	91.72	90.83	90.52	99.75	86.87
	F1-score	92.61	**99.98**	**91.86**	**91.34**	**90.57**	**99.80**	**86.89**

the performance of 92.16%, 91.78%, 92.15% in terms of recall, f-score, and precision, but the ensemble classifier outperforms the other classifiers with the highest f-score, which is 91.86%.

FIGURE 2d depicts the confusion matrix of the Xgb classifier on the Electricity dataset to evaluate the quality of the output of an XGB classifier. It shows that 0.9158% predicted labels are equal to the true label, and the classifier mislabels only 0.0842% labels. We plot a ROC curve that shows an XGB classification model's performance in FIGURE 3d. ROC curve of 91% suggests the XGB Classifier performs exceptionally for the electricity dataset.

4) SEA

In the SEA dataset, the MLP classifier achieves the best accuracy of 91.20% in comparison to other models. The classification accuracies of all the other models: KNN, Random Forest, XGB classifier, and ensemble classifier are 88.80%, 88.80%, 88.20%, and 90.00%, respectively. Random Forest, KNN, and XGB need more data instances to work better. This dataset also has 10% of noise so that on average, all the classifiers perform 89.40% accuracy, which is quite well on this dataset. The evaluation is performed according to the standard measures of precision, recall, and f-score. MLP achieves a proficient gain of 91.03%, 91.03%, and 91.07% for precision, recall, and f-score respectively. The ensemble model produces the highest f-score of 91.34% as compared to all the other classifiers shown in TABLE 4.

A better way to evaluate the classification performance of a model is to look at the confusion matrix. FIGURE 2e depicts the confusion matrix to evaluate the classification performance of MLP classifier on Sea dataset. it shows that the classifier truly classifies 0.9120% of class labels, and the MLP classifier misclassifies only 0.0880% class labels. FIGURE 3e depicts the ROC curve that shows an MLP classification model's performance. The ROC value of 90% indicates that this classifier is most suited for the sea dataset.

5) HYPERPLANE

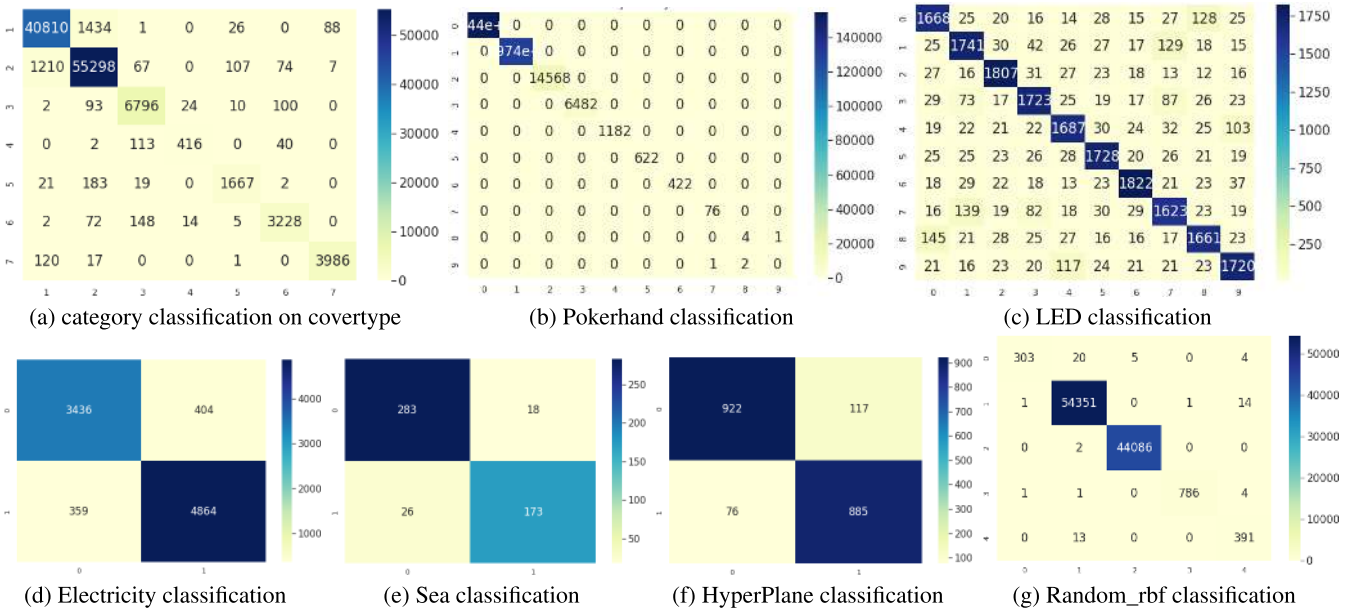
The average accuracy of the Hyperplane dataset from all the models is 88.83%. MLP classifier achieves the best accuracy of 90.14% as compare to all other models. MLP is a feed-forward ANN class, and it uses a supervised learning technique called backpropagation so that it performs well on the binary class dataset. All the other models such as KNN, Random Forest, XGB classifier, and Ensemble classifier achieve the accuracy score of 88.00%, 87.45%, 88.50%, and 90.10%, respectively. MLP classifier gives precision, recall, and f-score values of 90.23%, 90.28%, and 90.15% respectively but ensemble classifier outperforms MLP classifier in terms of precision, recall, and f-score with values of 90.53%, 90.52%, and 90.57% respectively.

FIGURE 2f shows the confusion matrix of MLP classifier on Hyperplane dataset to evaluate the classification performance. We plot confusion matrices to understand which classes are most easily confused, so only 76 values of class 0 instances are confused with other class instances, and on the other hand, only 117 values of class 1 are getting confused with class 0 instances.

FIGURE 3f shows the ROC curve of MLP classifier on hyperplane dataset. ROC curve shows the performance of a classification model. ROC curve value of 91% shows that MLP classifier is the best classification model for hyperplane dataset.

6) RandomRBF

The Ensemble classifier gains the highest accuracy of 99.94% as for the Random_rbf dataset. The average accuracy of the Random_rbf dataset from all the models is 98.02%, whereas the accuracy of KNN, XGB classifier, MLP classifier, and Random forest classifier is 91.18%, 99.84%, 99.21%, and



(a) category classification on covertype (b) Pokerhand classification (c) LED classification

(d) Electricity classification (e) Sea classification (f) HyperPlane classification (g) Random_rbf classification

FIGURE 2. Confusion matrix on all drift detection datasets.

TABLE 5. Performance comparison of the training classifiers with the baseline approaches. key: EoBagm-A, EoBagf-B, EoBagnbk-C, EoBagstd-D, EoBagmaj-E.

Baseline Approach Average Accuracy						Average Accuracy					Gain
Dataset	A	B	C	D	E	RF	KNN	XGB	MLP	Ensemble	-
Artificial Dataset	85.88	86.01	85.97	79.44	85.88	90.42	88.22	90.55	91.57	91.49	5.82%
Real Dataset	88.84	89.60	89.26	86.71	88.79	95.67	93.74	92.81	89.45	94.28	4.63%
Overall Average	87.15	87.55	87.38	82.55	87.13	92.67	90.58	91.52	90.52	92.68	5.24%

99.94%. The best score of precision, recall, and f-score are gained from the Ensemble classifier dataset, which is 99.65%, 99.75%, and 99.80%, respectively. We plot the confusion matrix of the category classification on the random_rbf dataset using the ensemble classifier, which gains the best accuracy compared to all other classifiers. The diagonal values signify the number of points where the predicted label is not confused and accurately classified in the confusion matrix. On the other hand, the off-diagonal values are mislabeled by the classifier. The higher the diagonal values, represent better and accurate predictions. FIGURE 2g shows the confusion matrix of ensemble classifier on Random_rbf dataset. FIGURE 3g shows the ROC curve of ensemble classifier on Random_rbf dataset. The ensemble classifier's ROC curve indicated that this model is excellent for random_rbf classification because of its ROC curve value.

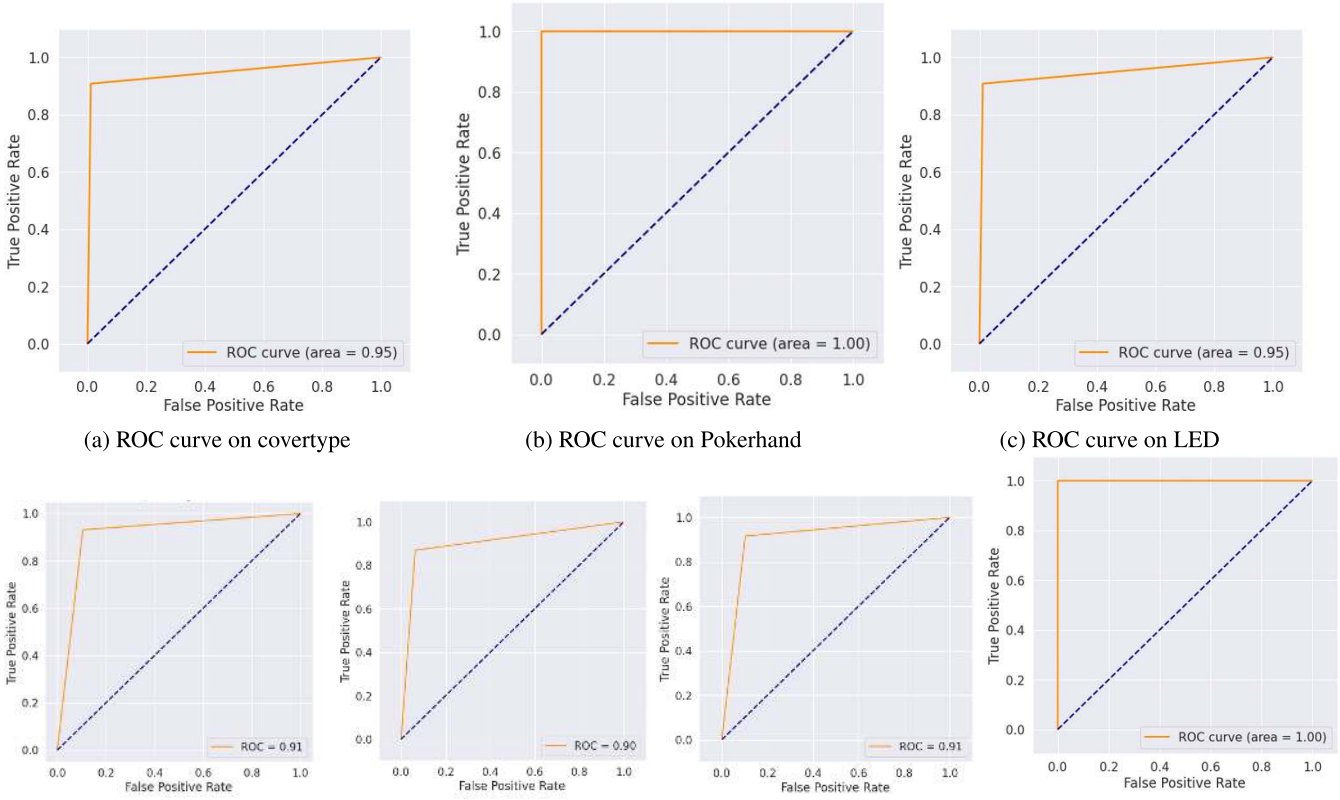
7) LED

The average accuracy of the LED dataset from all models is 85.54%, in which the ensemble classifier performs the best with an accuracy of 85.92% as compare to other models. All the other models KNN, Random Forest, MLP classifier, and XGB classifier, gain an accuracy score of 84.91%, 85.50%, 85.74%, and 85.67%. The LED dataset's main aim is to predict the digit on a seven-segment display, where each digit has an equally 10% chance of being displayed. The concept drifts happen after every 25, 000 instances and a transition length of w = 500 to simulate gradual concept drifts, also 10% class noise added to each data stream. This dataset becomes so complex that all of these parameters give all models' average accuracy only 85.54%. We also calculated the evaluation matrix precision, recall, and f-score, which are 86.89%, 86.87%, and 86.89%, respectively, for the ensemble classifier.

FIGURE 2c shows the confusion matrix of the category classification on the Led dataset using ensemble classifier, which achieves the best accuracy compared to all other classifiers. The diagonal values in the confusion matrix represent the number of points where the predicted label accurately classify. Only 0.8592% class labels are accurately classified. On the other hand, the off-diagonal values are mislabeled by the classifier, where the classifier mislabels only 0.1408% values. The higher diagonal values signify the better and accurate prediction of the classifier. FIGURE 3c shows the ROC curve of ensemble classifier. ROC curve is showing the performance of the ensemble classification model. Its ROC curve value is 95%, which indicates that the ensemble model performs exceptionally well for the LED dataset.

B. COMPARATIVE ANALYSIS WITH BASELINE APPROACH

To analyze the performance of the *ElStream*, we compare the results with state-of-the-art study [11] whose experimental



(a) ROC curve on covertype (b) ROC curve on Pokerhand (c) ROC curve on LED

(d) ROC curve on Electricity (e) ROC curve on Sea (f) ROC curve on HyperPlane (g) ROC curve on Random_rbf

FIGURE 3. ROC curve on all drift detection datasets.

TABLE 6. Accuracy gain comparison of *ElStream* with baseline approach.

Dataset	Highest Baseline	ElStream	Gain
PokerHand	87.50%	**99.99%**	**12.49%**
Covertype	**92.35%**	92.22%	-0.13%
Electricity	89.45%	90.65%	1.2%
SEA	89.67%	90.00%	0.33%
Hyperplane	90.56%	90.10%	-0.46%
Random RBF	89.88%	99.94%	10.06%
LED	73.94%	85.92%	11.98%

settings resembled the settings followed in this study. The authors used an online ensemble bagging method to classify streaming data. A lenient threshold was set to generate a warning of concept drift and start training a base classifier. When a concept drift occurs, the new base classifier takes the worst-performing classifier's place in the ensemble.

TABLE 5 gives an overview of the comparison of this study with our proposed approach. For artificial dataset baseline approach achieved 85.88% average accuracy using EoBagm, 86.01% using EoBagf, 85.97% using EoBagnbk, 79.44% using EoBagstd and 85.88% using EoBagmaj. They achieved the highest average accuracy of 86.01% using EoBagf. *ElStream* achieved an average accuracy of 90.42% using Random Forest, 88.22% using KNN, 90.55% for XGBoost, 91.57% using MLP, and 91.49% using the ensemble. *ElStream* achieves the highest accuracy of 91.57% using MLP followed closely by the ensemble with an accuracy of 91.49%. For real dataset baseline approach achieved 88.84% average accuracy using EoBagm, 89.60% using EoBagf, 89.26% using EoBagnbk, 86.71% using EoBagstd, and 88.79% using EoBagmaj. They achieved the highest accuracy of 89.60% using EoBagf. *ElStream* achieved an average accuracy of 95.67% using RF, 93.74% using KNN, 92.81% using XGBoost, 89.45% using MLP, and 94.28% using the ensemble. *ElStream* achieved the highest accuracy of 95.67% using RF, followed by the ensemble with an accuracy of 94.28%.

The overall average accuracy of datasets from the baseline approach achieved 87.15% accuracy using EoBagm, 87.55% using EoBagf, 87.38% using EoBagnbk, 82.55% using EoBagstd, and 87.13% using EoBagmaj. They achieved the highest accuracy of 87.55% using EoBagf. While Our *ElStream* achieved an overall average accuracy of 92.67% using RF, 90.58% using KNN, 91.52% using XGBoost, 90.52% using MLP, and 91.68% using the ensemble. *ElStream* achieved the highest accuracy of 92.68% using the Ensemble Method. We produced a proficient gain of 5.82% and 4.63% in terms of accuracy from Artificial and real datasets. We get a 5.24% gain in terms of accuracy from the baseline approach's overall average accuracy.

TABLE 6 provides the comparison of this study with our proposed approach compared with the highest performing classifier in the referred study. The proposed ensemble

approach (*ElStream*) achieved an increased accuracy of 12.49%, 11.98%, 10.06%, 1.2%, and 0.33% for PokerHand, LED, Random RBF, Electricity, and SEA dataset respectively. A decline of 0.13% for Cover type and 0.46% for the Hyperplane dataset was seen using the *ElStream* approach, which is almost negligible compared with the gain in accuracy for other datasets.

VI. DISCUSSION

In this study, an ensemble learning method, namely *ElStream* is proposed that enables precise classification of online streaming data with concept drifts. *ElStream* utilizes seven different real and artificial datasets for classification. Different machine learning and ensemble learning methods using majority voting are used. In the proposed ensemble method, classifiers can only vote if they cross a certain prediction threshold. *ElStream* approach outperforms the classical machine learning algorithms in terms of accuracy and the f-score evaluation metrics. No single classifier works best for every type of dataset, either real or artificial, whereas the *ElStream* approach gives comparable results for every type of data. The overall most accurate method is the Ensemble classifier. On average, our ensemble method performs the best as compare to all the methods. The Random Forest method performed much better on average than KNN, XGB, and MLP classifiers but as compared to baseline approaches our all methods performed well in which Ensemble method is the most accurate method from all methods. The baseline approach achieved the highest accuracy of 92.35%, but *ElStream* method gets the best accuracy which is 99.99%, which shows a proficient gain of 7.64%. Experimental analysis proved that the proposed *ElStream* approach performs competently to detect concept drifts and can classify the data with better accuracy than other state-of-the-art studies.

VII. CONCLUSION AND FUTURE WORK

With the increasing velocity and volume of streaming data, there is a need for the models to evolve based on that data constantly. Ensemble learning uses a combination of different classifiers to make predictions that lead to better performance. This research conducted experiments with multiple classifiers on both artificial and real datasets and found the classifiers performing best to use in the ensemble. An ensemble learning approach *ElStream* is proposed that uses majority voting to make predictions. Each classifier in the ensemble can only predict if the confidence level passes a certain threshold. *ElStream* approach gave higher accuracy and f-score rates when compared with similar state-of-the-art studies. This finding can significantly help in the application of big data analytics in real-life applications. Our strategy is to investigate the most acceptable methods for determining which classifiers can handle various concept drifts in future work. Besides, We also plan to reduce the computational complexity of our algorithm more.*

ACKNOWLEDGMENT

The authors would like to thank the Deanship of Scientific Research at Prince Sattam Bin Abdul-Aziz University, Saudi Arabia.

REFERENCES

[1] N. N. Thilakarathne, M. K. Kagita, and D. T. R. Gadekallu, "The role of the Internet of Things in health care: A systematic and comprehensive study," *Int. J. Eng. Manage. Res.*, vol. 10, no. 4, pp. 145–159, Aug. 2020.

[2] A. Banerjee, C. Chakraborty, A. Kumar, and D. Biswas, "Emerging trends in IoT and big data analytics for biomedical and health care technologies," in *Handbook of Data Science Approaches for Biomedical Engineering*. Amsterdam, The Netherlands: Elsevier, 2020, pp. 121–152.

[3] K. N. Mishra and C. Chakraborty, "A novel approach towards using big data and iot for improving the efficiency of m-health systems," in *Advanced Computational Intelligence Techniques for Virtual Reality in Healthcare*. Cham, Switzerland: Springer, 2020, pp. 123–139.

[4] W. Chen, G. Feng, C. Zhang, P. Liu, W. Ren, N. Cao, and J. Ding, "Development and application of big data platform for garlic industry chain," *Comput., Mater. Continua*, vol. 58, no. 1, p. 229, 2019.

[5] M. U. Khan, A. R. Javed, M. Ihsan, and U. Tariq, "A novel category detection of social media reviews in the restaurant industry," *Multimedia Syst.*, vol. 219, pp. 1–14, Oct. 2020.

[6] G. R. Bojja, M. Ofori, J. Liu, and L. S. Ambati, "Early public outlook on the coronavirus disease (COVID-19): A social media study," in *Proc. Americas Conf. Inf. Syst. (AMCIS)*, Salt Lake City, UT, USA, 2020.

[7] A. L'Heureux, K. Grolinger, H. F. Elyamany, and M. A. M. Capretz, "Machine learning with big data: Challenges and approaches," *IEEE Access*, vol. 5, pp. 7776–7797, 2017.

[8] K. Grolinger, M. Hayes, W. A. Higashino, A. L'Heureux, D. S. Allison, and M. A. M. Capretz, "Challenges for mapreduce in big data," in *Proc. IEEE World Congr. Services*, Jul. 2014, pp. 182–189.

[9] B. Krawczyk and A. Cano, "Online ensemble learning with abstaining classifiers for drifting and noisy data streams," *Appl. Soft Comput.*, vol. 68, pp. 677–692, Jul. 2018.

[10] H. M. Gomes, J. P. Barddal, F. Enembreck, and A. Bifet, "A survey on ensemble learning for data stream classification," *ACM Comput. Surv.*, vol. 50, no. 2, pp. 1–36, Jun. 2017.

[11] Y. Lv, S. Peng, Y. Yuan, C. Wang, P. Yin, J. Liu, and C. Wang, "A classifier using online bagging ensemble method for big data stream learning," *Tsinghua Sci. Technol.*, vol. 24, no. 4, pp. 379–388, Aug. 2019.

[12] G. T. Reddy, M. P. K. Reddy, K. Lakshmanna, R. Kaluri, D. S. Rajput, G. Srivastava, and T. Baker, "Analysis of dimensionality reduction techniques on big data," *IEEE Access*, vol. 8, pp. 54776–54788, 2020.

[13] N. Deepa, Q.-V. Pham, D. C. Nguyen, S. Bhattacharya, B. Prabadevi, T. R. Gadekallu, P. K. R. Maddikunta, F. Fang, and P. N. Pathirana, "A survey on blockchain for big data: Approaches, opportunities, and future directions," 2020, *arXiv:2009.00858*. [Online]. Available: https://arxiv.org/abs/2009.00858

[14] J. L. Lobo, J. D. Ser, A. Bifet, and N. Kasabov, "Spiking neural networks and online learning: An overview and perspectives," *Neural Netw.*, vol. 121, pp. 88–100, Jan. 2020.

[15] Y. Sun, K. Tang, L. L. Minku, S. Wang, and X. Yao, "Online ensemble learning of data streams with gradually evolved classes," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 6, pp. 1532–1545, Jun. 2016.

[16] A. Pesaranghader, H. L. Viktor, and E. Paquet, "McDiarmid drift detection methods for evolving data streams," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–9.

[17] A. Pesaranghader and H. L. Viktor, "Fast Hoeffding drift detection method for evolving data streams," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Italy: Springer, 2016, pp. 96–111.

[18] D. T. J. Huang, Y. S. Koh, G. Dobbie, and A. Bifet, "Drift detection using stream volatility," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Porto, Portugal: Springer, 2015, pp. 417–432.

[19] D. Ge and X.-J. Zeng, "Learning data streams online—An evolving fuzzy system approach with self-learning/adaptive thresholds," *Inf. Sci.*, vol. 507, pp. 172–184, Jan. 2020.

[20] A. O. M. Abuassba, Y. Zhang, X. Luo, D. Zhang, and W. Aziguli, "A heterogeneous ensemble of extreme learning machines with correntropy and negative correlation," *Tsinghua Sci. Technol.*, vol. 22, no. 6, pp. 691–701, Dec. 2017.

[21] M. M. Idrees, L. L. Minku, F. Stahl, and A. Badii, "A heterogeneous online learning ensemble for non-stationary environments," *Knowl.-Based Syst.*, vol. 188, Jan. 2020, Art. no. 104983.
[22] J. N. van Rijn, G. Holmes, B. Pfahringer, and J. Vanschoren, "The online performance estimation framework: Heterogeneous ensemble learning for data streams," *Mach. Learn.*, vol. 107, no. 1, pp. 149–176, Jan. 2018.
[23] T. T. Nguyen, T. T. T. Nguyen, X. C. Pham, A. W.-C. Liew, and J. C. Bezdek, "An ensemble-based online learning algorithm for streaming data," 2017, *arXiv:1704.07938*. [Online]. Available: https://arxiv.org/abs/1704.07938
[24] Y. Sun, Z. Wang, H. Liu, C. Du, and J. Yuan, "Online ensemble using adaptive windowing for data streams with concept drift," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 5, May 2016, Art. no. 4218973.
[25] C. Blake. (1998). *UCI Repository of Machine Learning Databases*. [Online]. Available: https://www. ics.uci. edu/~ mlearn/MLRepository. html
[26] A. R. Javed, Z. Jalil, S. A. Moqurrab, S. Abbas, and X. Liu, "Ensemble adaboost classifier for accurate and fast detection of botnet attacks in connected vehicles," *Trans. Emerg. Telecommun. Technol.*, p. e4088, Aug. 2020.
[27] S. Hakak, M. Alazab, S. Khan, T. R. Gadekallu, P. K. R. Maddikunta, and W. Z. Khan, "An ensemble machine learning approach through effective feature extraction to classify fake news," *Future Gener. Comput. Syst.*, vol. 117, pp. 47–58, Apr. 2021.
[28] W. Zehra, A. R. Javed, Z. Jalil, H. U. Khan, and T. R. Gadekallu, "Cross corpus multi-lingual speech emotion recognition using ensemble learning," *Complex Intell. Syst.*, vol. 749, pp. 1–10, Jan. 2021.
[29] A. Basit, M. Zafar, A. R. Javed, and Z. Jalil, "A novel ensemble machine learning method to detect phishing attack," in *Proc. IEEE 23rd Int. Multitopic Conf. (INMIC)*, Nov. 2020, pp. 1–5.
[30] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on ensemble learning," *Frontiers Comput. Sci.*, vol. 14, no. 2, pp. 1–18, 2020.
[31] S. Saha, M. Saha, K. Mukherjee, A. Arabameri, P. T. T. Ngo, and G. C. Paul, "Predicting the deforestation probability using the binary logistic regression, random forest, ensemble rotational forest, REPTree: A case study at the Gumani river basin, India," *Sci. Total Environ.*, vol. 730, Aug. 2020, Art. no. 139197.
[32] E. K. Sahin, "Assessing the predictive capability of ensemble tree methods for landslide susceptibility mapping using XGBoost, gradient boosting machine, and random forest," *Social Netw. Appl. Sci.*, vol. 2, no. 7, pp. 1–17, Jul. 2020.

AHMAD ABBASI is currently pursuing the master's degree in data science with Air University, Islamabad, Pakistan. He is also with the Department of Cyber Security, Air University. He is also working with the National Cybercrimes and Forensics Laboratory, Air University.

ABDUL REHMAN JAVED received the master's degree in computer science from the National University of Computer and Emerging Sciences, Islamabad, Pakistan. He worked with the National Cybercrimes and Forensics Laboratory, Air University, Islamabad. He is currently a Lecturer with the Department of Cyber Security, Air University. He has authored more than 20 peer-reviewed articles on cybersecurity, mobile computing, and digital forensics topics. His current research interests include, but are not limited to, mobile and ubiquitous computing, data analysis, knowledge discovery, data mining, natural language processing, smart homes, their applications in human activity analysis, human motion analysis, and e-health. He aims to contribute to interdisciplinary research of computer science and human-related disciplines.

CHINMAY CHAKRABORTY worked with the Faculty of Science and Technology, ICFAI University, Agartala, India, as a Senior Lecturer. He worked as a Research Consultant in the Coal India Project at Industrial Engineering and Management, IIT Kharagpur. He worked as a Project Coordinator of the Telecommunication Convergence Switch Project under the Indo-U.S. joint initiative. He worked as a Network Engineer in system administration at MISPL, India. He is currently a Senior Assistant Professor in electronics and communication engineering with the Birla Institute of Technology, Mesra, India. He has published 90 articles in reputed international journals, conferences, book chapters, and books. His main research interests include the Internet of Medical Things, wireless body sensor networks, wireless networks, telemedicine, m-Health/e-health, and medical imaging. He is a member of the Internet Society, the Machine Intelligence Research Labs, and the Institute for Engineering Research and Publication. He received the Best Session Runner-Up Award, the Young Research Excellence Award, the Global Peer Review Award, the Young Faculty Award, and the Outstanding Researcher Award. He was the Speaker for AICTE, DST sponsored FDP, and CEP Short Term Course. He has served as a Publicity Chair Member of renowned international conferences, including IEEE Healthcom and IEEE SP-DLT. He is an editorial board member of the different journals and conferences. He is serving as a Guest Editor for *Future Internet Journal* (MDPI), *Internet Technology Letters* (Wiley), *Annals of Telecommunications* (Springer), and the *International Journal of System Assurance Engineering and Management* (Springer). He is also a Lead Guest Editor of the *International Journal of E-Health and Medical Communications* (IGI), *Multimedia Tools and Applications* (Springer), *CMC* (TechScience), *Interdisciplinary Sciences: Computational Life Sciences* (Springer), the *International Journal of Nanotechnology* (Inderscience), *Current Medical Imaging* (BenthamScience), and *Journal of Medical Imaging and Health Informatics*. He is also a Lead Series Editor of *Advances in Smart Healthcare Technologies* (CRC). He is an Associate Editor of the *International Journal of End-User Computing and Development*. He has conducted a session of SoCTA-19, ICICC-2019, Springer CIS 2020, SoCTA-20, and SoCPaR 2020. He is also a Reviewer for international journals, including IEEE Access, IEEE Sensors Journal, IEEE Internet of Things Journal, Elsevier, Springer, Taylor & Francis, IGI, IET, TELKOMNIKA Telecommunication Computing Electronics and Control, and Wiley. He is co-editing several books on Smart IoMT, Healthcare Technology, and Sensor Data Analytics with Elsevier, CRC Press, IET, Pan Stanford, and Springer.

JAMEL NEBHEN has worked as a Postdoctoral Researcher in many laboratories in France, like LIRMM Montpellier, IM2NP Marseille, ISEP Paris, LE2I Dijon, Lab-Sticc Telecom Bretagne Brest, and IEMN Lille. Since 2019, he has been with PSAU University, Saudi Arabia, as an Assistant Professor. His research interests include the design of analog integrated circuits, analog, and RF circuits for wireless communications, analog CMOS instrumentation, wireless sensors, and the IoT systems.

WISHA ZEHRA received the bachelor's degree in computer science from Air University, Islamabad, Pakistan. She is currently a Research Assistant with the National Center for Cyber Security, Air University. Her current research interests include, but are not limited to, data analysis, data mining, natural language processing, human–robot interaction, audio processing, and computer forensics.

ZUNERA JALIL (Member, IEEE) received the master's degree in computer science from the National University of Computer and Emerging Sciences, Islamabad, Pakistan. He worked with the National Cybercrimes and Forensics Laboratory, Air University, Islamabad. He is currently a Lecturer with the Department of Cyber Security, Air University. He has authored more than 20 peer-reviewed articles on cybersecurity, mobile computing, and digital forensics topics. His current research interests include, but are not limited to, mobile and ubiquitous computing, data analysis, knowledge discovery, data mining, natural language processing, smart homes, their applications in human activity analysis, human motion analysis, and e-health. He aims to contribute to interdisciplinary research of computer science and human-related disciplines.