International Journal of Applied Mathematics

Volume 38 No. 8s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

A NOVEL WEIGHTED ELM ENSEMBLE FRAMEWORK FOR CONCEPT DRIFT PREDICTION, DETECTION AND ADAPTION IN UNRELIABLE IOT DATA STREAM

Hezal Lopes[1, 2*], Prashant Nitnaware[2]

[1,2*] D.J. Sanghvi College of Engineering, Vile Parle, India

[2] Pillai College of Engineering, New Panvel, India

* Corresponding author's Email: hezal.lopes@outlook.com

Abstract

Real-time IoT applications that use sensors, and those sensors generate continuous and unreliable data streams. This IoT stream suffers from Concept drift, which is defined as changes in data distribution over time. Traditional machine learning models fail to adapt to evolving environments, which degrades the model's performance. This paper presents a novel Weighted Ensemble Extreme Learning Machine (ELM) framework that integrates drift prediction, detection, and adaptation into a single architecture. The proposed approach employs an Adaptive Windowing (ADWIN) technique combined with non-parametric statistical tests, where the Kolmogorov-Smirnov (KS) test demonstrates superior performance in drift detection. To handle adaptation, an Online Sequential ELM ensemble is dynamically updated through exponential moving average-based weighting and pruning, ensuring robust generalization and reduced computational overhead. The framework is validated on benchmark datasets, including SEA, Stagger, and a farm weather dataset with artificially induced sudden, gradual, and recurring drifts using Gaussian noise. Experimental results reveal that the proposed method consistently outperforms single classifiers, achieving higher accuracy, precision, recall, and F1-scores, with an average accuracy improvement of up to 97.35% when drift handling is incorporated. These results demonstrate the effectiveness of combining ensemble learning with adaptive drift management. The contributions of this work lie in developing a comprehensive and lightweight framework that unifies prediction, detection, and adaptation of concept drift, thereby enhancing the reliability of IoT-driven decision-making systems. This research provides a scalable foundation for real-time applications such as weather forecasting and can be extended to other domains where streaming data is subject to dynamic changes.

Keywords: Adaptive windowing, concept drift prediction, data stream mining, drift detection, Ensemble Classifier, IoT Sensor Data.

1. Introduction

Information generated by multiple sensors is called a data stream, which is marked by its continuous, unbounded, and high-speed structure, arriving sequentially. With the time, the

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 8s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

underlying data distribution can change—a behavior referred to as concept drift. Weather forecasting is a good example of it, in which variations in latent environmental conditions change the target concept. A successful learning model should identify and adjust to such changes quickly [1]. Data streams come from various sources like distant sensors, surveillance systems, financial markets, and social media, where conventional data mining methods do not operate effectively [2]. Conventional machine learning techniques tend to perform poorly in dynamic, drift-prone settings [3], which requires prompt drift detection and adaptive model choosing [4].

Early detection or anticipation of drift is important for effective stream mining. Methods like pattern matching, Naïve Bayes, and probabilistic models help to detect drift, but conventional data mining algorithms based on multiple scans over the data are not practical for stream-based scenarios. In order to address this drawback, incremental or online mining techniques are used for ongoing learning [5]. Common statistical methods for detecting drift are CUSUM, Page-Hinckley (PH), Drift Detection Method (DDM), Early Drift Detection Method (EDDM), Reactive Drift Detection Method (RDDM), Fisher-based Statistical Drift Detector (FSDD), and McDiarmid Drift Detection Method (MDDM). Window-based methods like Concept-Adapting Very Fast Decision Tree (CVFDT) and Adaptive Sliding Window (ADWIN), and block-based ensemble techniques like Accuracy Weighted Ensemble (AWE) and Accuracy Updated Ensemble (AUE) have been highly adaptable.

In addition, incremental or online ensemble methods handle data sequentially, allowing for quicker and more efficient adaptation to changing data streams. Dynamic weighted majority (DWM), Learn++, and variants of this algorithm are called Incremental-based ensemble techniques. Statistical-based methods and window-based methods are the most popularly used methods for drift detection [6-10]. Once drift is detected by the drift detection method, further need to adapt the drift with a machine learning model. Drift can be adapted by a classifier model. Data stream classifiers are divided into two major categories: single classifiers and ensemble classifiers. A single classifier adjusts itself for the latest data, which is resource-consuming. Whereas ensemble classifiers store the previous information in the form of ensemble members, they thus obtain good generalization capability as compared to a single classifier [11-12].

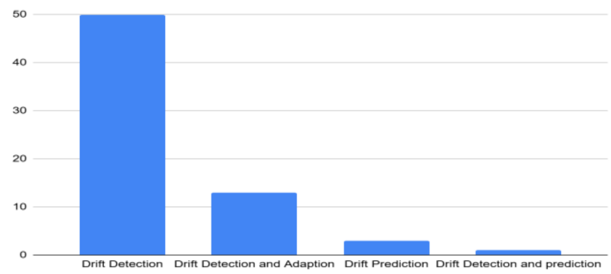

Figure 1: Statistical Summary of Drift Prediction, Detection, and Adaption

Received: August 12 2025

International Journal of Applied Mathematics
Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Further, it has been found that maximum research is done on identifying various drift detection techniques, as shown in Figure 1. The literature lacks a cohesive framework to develop multi-model architecture. Therefore, a comprehensive architecture to handle drift in an efficient way is essential to adapt the model to the changing data patterns. The methodology should have a prediction, detection, and adaptation approach that is able to handle various types of drifts and carry out mining procedures built using the latest technology and lightweight methodology. In this regard, this study introduces a new paradigm designed to forecast, identify, and adapt to drift in real-time data streams using an adaptive windowing (ADWIN) approach and an extreme learning machine (ELM) as the underlying classifier.

The rest of the paper has the following organization: Section II explains background concepts and existing trends for drift handling procedures. Section III outlines the proposed approach. Section IV gives an overview of the implementation and findings. Section V demonstrates the findings, Section VI offers a discussion of the findings, and Section VI concludes the paper and proposes future research.

2. Drift Handling Methods

Concept drift can be handled by three important components like prediction of drift, detection of drift, and adaptation of drift. All the components are discussed in detail below.

Concept Drift prediction:

Early detection of drift is important as we can't ignore small changes in live data. It is required that drift should be predicted or early detected, or when it is observed, it should be adopted and handled in the proper way.

Figure 2 below shows various types of drift prediction methods. One way to predict drift is using metadata clustering. After each drift, this method calculates data stream metadata and adds current metadata to clustering and identify how similar the current metadata state is to when drifts have occurred in the past [13].

Given the latest pattern, the system predicts the next pattern using a probabilistic network. Using this method, the system is able to predict future drift positions [14]. A volatility detector locates local volatility changes using drift intervals, and a drift predictor that uses information from both the drift detector and volatility detector to estimate the next drift point [15]. Early drift detection using a decision tree is proposed in [16]. It has been noted by researchers that, during the initial stage of concept drift detection, the error rate of a model doesn't fluctuate. In this situation, error rate-based drift signals can't represent changes in data distribution. With the help of the Prediction Uncertainty Index (PU-index), the measure of probability assigned by a classifier that an instance does not belong to the actual class prior to the model's error drop [17]. Instead of using error classification, the Early drift detection method measures the distance between classification errors [18].

Received: August 12 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

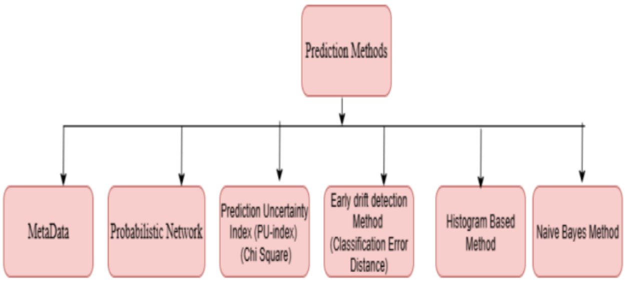

Figure 2: Drift Prediction Methods

Concept Drift Detection

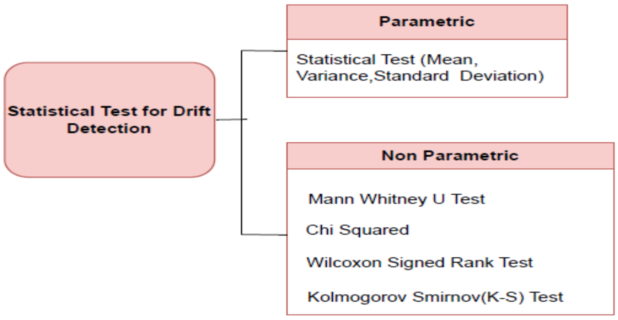

Figure 3: Statistical test used for Drift Detection

Figure 3 depicts various types of statistical tests used for drift detection. They are parametric tests like mean, variance, standard deviation, and non-parametric tests like Mann-Whitney U test, Chi-squared, Wilcoxon Signed Rank test, and KS test. For this research, we are considering non non-parametric statistical test. The Mann-Whitney U test is used to determine changes in data distribution [19]. As shown in equations 1 and 2, it calculates the statistics of two different groups. This test reduces the impact of outliers and aims at the central tendency of the data. All data values are ranked together for examination.

$$U_1 = R_1 + \left(\frac{N_1(N_1+1)}{2}\right) \quad (1)$$

Here, N_1 is the number of data points, and R_1 is the sum of ranks of the first group. Likewise, U_2 is computed for the second group.

$$Stat = \min(U_1, U_2) \quad (2)$$

The null hypothesis states that there is no systematic rank difference between the two distributions, and the test statistic has an asymptotic normal distribution.

Received: August 12 2025

Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The Wilcoxon Rank Sum Test [20] tests for differences between two independent samples by arranging m + n observations in a common ordered list. The test statistic T is the rank sum of $\{X_1, \ldots, X_m\}$ in the combined sample $\{X_1, \ldots, X_m; Y_1, \ldots, Y_n\}$, given in Equation (3).

$$T = \sum_{i=1}^{m} R_i \, , i = 1 \cdots m \qquad (3)$$

Two samples are said to be consistent if T is not in the rejection region.

The Kolmogorov-Smirnov (KS) test [21] is then applied to compare the empirical cumulative distributions of R and W. The test calculates the maximum absolute difference ($dist_{w,r}$) between these distributions. It is calculated as below equation 4.

Secondly, the Kolmogorov–Smirnov Test [21] validates the empirical cumulative distributions of R and W by assessing the maximum absolute distance ($dist_{w,r}$) between them through Equation (4),

$$dist_{w,r} = \sup_{x} \mid F_W(x) - F_R(x) \mid \qquad (4)$$

where $F_W(x)$ and $F_R(x)$ are the theoretical and empirical cumulative distribution functions (CDF) of the sample. There is a concept drift if the distance is greater than the critical value (D_α).

Lastly, the Chi-Squared Test [22] determines if a difference is significant between observed and expected frequencies of categorical data. It is used to examine independence or association between variables as given in Equation (5).

$$T = \chi^2 = \sum \frac{(observed - expected)^2}{expected} \qquad (5)$$

Where $\chi 2$ is the Chi-squared statistic.

Concept Drift Adaption

Once concept drift is detected, it should be handled by updating the current model. That helps to enhance the performance of the model. Traditional machine learning algorithms operate on static data. For data streams, data arrives in a fast, continuous, and temporal manner. Because of this, traditional machine learning algorithms are no longer suitable for data streams. In conventional learning, researchers have proposed an adaptive learner, and a decision tree is an example of an adaptive learner. Further researchers have studied Naive Bayes, SVM, and K-Nearest Neighbors classifiers with various drift detection techniques [23].

For Data stream mining, researchers have used a single classifier and an ensemble classifier. For every new data, single classifiers need to constantly adjust themself. And further, it is also not able to store previous streaming data. But in the case of ensemble models, previous information is stored in the form of ensemble members. Hence, ensemble classifiers give better results as compared to a single classifier[24-25].

Ensemble learning models combine the outputs of multiple base learners so they are

Received: August 12 2025

Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

generalizable. In an ensemble classifier, we can reuse the than training new models. Accuracy-weighted ensemble (AWE), Accuracy-updated ensemble (AUE), and the Streaming Ensemble Algorithm (SEA) are a few popular algorithms of block-based ensemble classifiers [26-29]. Various types of drift adaption techniques are shown in Figure 4.

Researchers have proposed the Extreme Learning Machine (ELM), a single hidden-layer feedforward neural network (SLFN) model that is built for fast learning and effective data stream processing. Contrary to the conventional methods using gradient, ELM uses random input weights with output weights analytically computed, indicating a considerable decrease in training time. Comparative studies have revealed that ELM is faster and more accurate compared to some of the most popular classifiers like Restricted Boltzmann Machine (RBM), Support Vector Machine (SVM), Hopfield Neural Network (HNN), Radial Basis Function Neural Network (RBFNN), Random Neural Network (RNN), Deep Belief Network (DBN), and Boltzmann Machine (BM). Experiments confirm that ELM has very good generalization performance with little computational cost. Additionally, its flexibility allows it to be optimized for real-time use and big data analysis. A variety of ELM variants have been created over the years to improve robustness, stability, and performance in evolving data settings, which renders it a promising instrument for contemporary intelligent systems [30-31].

Further researchers have used an ensemble extreme learning machine that helps to improve the accuracy of the model as well as adapt to new concepts in a short time [32].

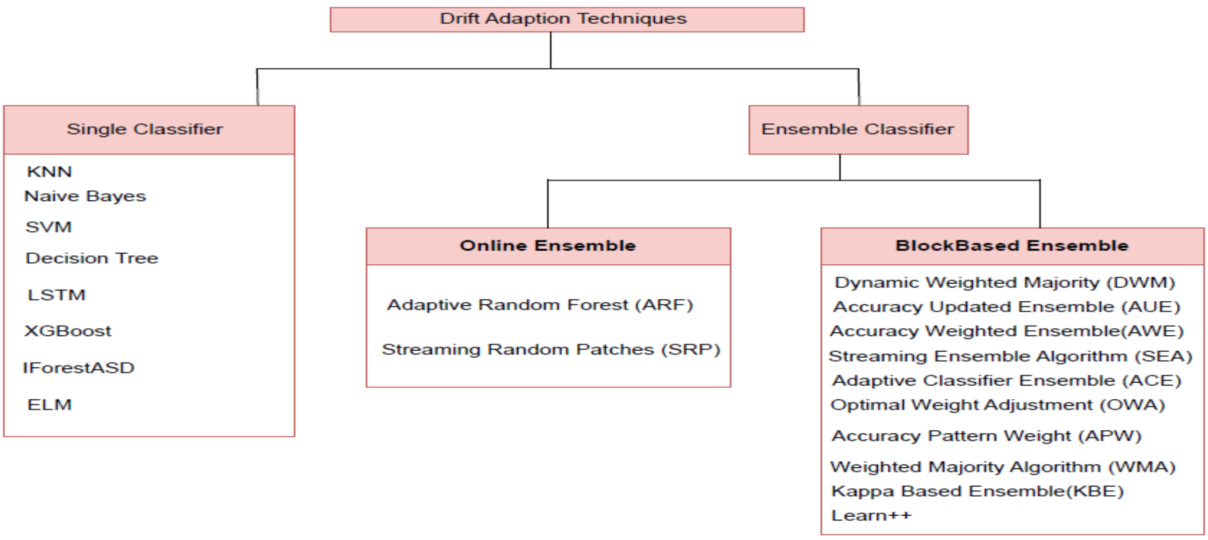

Figure 4: Drift Adaption Techniques

III. Proposed Methodology

This section introduces the novel weighted ensemble ELM classifier designed for real-time perception drift recognition in multi-dimensional sensor data streams, specifically focusing on weather monitoring systems. System architecture, as shown below in Figure 5, consists of three important components. Data Preprocessing, windowing technique, and Drift handling module

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 8s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

with weighted ensemble ELM store.

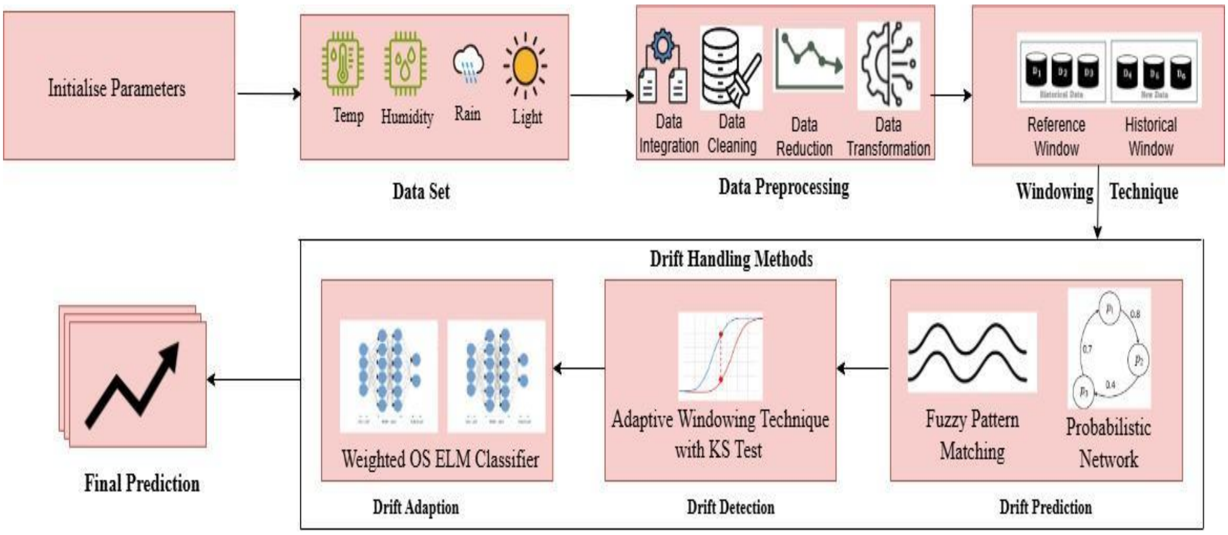

Figure 5: System Architecture for Drift Handling Methods

Algorithm: Weighted Ensemble with Updated OS-ELM

Algorithm: Weighted Ensemble with Updated OS-ELM

Begin:

//Input: Stream of data (Xt, yt), Ensemble of classifiers C = {C1, C2, ... , Cn}

//Parameters: threshold θ, EMA decay factor β, L (number of hidden layers), λ (L1 penalty)

//Output: Updated ensemble predictions and improved accuracy

Step 1: Evaluate classifiers

For each classifier C_i in C:

  Calculate Accuracy(C_i)

  Calculate F1 Score(C_i)

Step 2: Assign weights using Exponential Moving Average (EMA)

For each classifier C_i:

$w_i(t) = \alpha \cdot w_i(t-1) + (1-\alpha) \cdot Acc_t(C_i)$

Step 3: Ensemble pruning and update

For each classifier Ci:

  If wi(t) < θ:

    Remove Ci from ensemble

    Train new classifier Cnew with recent data (Xt, yt)

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 8s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Add Cnew to ensemble

Step 4: Updated Online Sequential ELM Classifier

1. Attention Layer (Dynamic Feature Reweighting):

$\alpha_t = \text{softmax}(W_a X_t + b_a t)$

W_a: Weight matrix

b_a: Bias term

$X_t^{att} = \alpha_t \cdot X_t$ // scaled up each feature

2. Stacked Hidden Layers (Deep ELM):

Input Ht(0) = X'_t

For l = 1 to L:

Final hidden representation: Ht(L)

$H_t^{(l)} = g^{(l)}\left(W^{(l)} H_t^{(l-1)} + b^{(l)}\right)$# random weights Wl, bias bl, activation g()

3. Sparse Output Layer (L1-Regularized RLS Update):

Prediction: ŷt = Ht^(L) · β

Update βt recursively using RLS with L1 penalty:

$$\beta_{t+1} = \beta_t + P_t H_t^T (y_t - H_t \beta_t) - \lambda \cdot \text{sign}(\beta_t)$$

Algorithm: Drift Detection with Adaptive Windowing (ADWIN)

Algorithm: Drift Detection with Adaptive Windowing (ADWIN)

Input:

- Streaming sensor data {Temp, Humidity, Rain, Light}
- α (significance level for KS Test)
- Threshold value θ
- Initial window size W0

Output:

- Preprocessed data stream
- Drift detection alerts

Step 1: Initialize Parameters

Initialize WindowSize ← W0

Set α ← predefined significance level for KS Test

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 8s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
Set Threshold ← θ
Initialize HistoricalWindow ← Ø
Initialize ReferenceWindow ← Ø
Step 2: Data Collection
For each time step t:
   Collect data point Dt = {Temp_t, Humidity_t, Rain_t, Light_t}
Step 3: Data Preprocessing
For each incoming batch of data B:
 Data Cleaning:
      For each feature f in B:
         If missing values → replace with mean(f)
 Data Transformation:
      For each feature f in B: Apply Min-Max normalization
          Return Preprocessed Batch B'
Step 4: Windowing Technique (ADWIN)
For each preprocessed data point xt in B':
   Add xt to ReferenceWindow
   If ReferenceWindow size >= WindowSize:
      Perform KS Test between HistoricalWindow and ReferenceWindow
      If p_value < α AND |mean(HistoricalWindow) - mean(ReferenceWindow)| > Threshold:
         // Drift Detected
         Raise Drift Alert at time t
         Shrink WindowSize ← decrease size adaptively
         Move ReferenceWindow → HistoricalWindow
         Reset ReferenceWindow ← Ø
      Else: // No Drift
         Increase WindowSize adaptively
         Append ReferenceWindow data to HistoricalWindow
         Reset ReferenceWindow ← Ø
```

Drift Detection

Received: August 12 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Streaming data analysis focuses on recent data rather than the entire data set. Adaptive Windowing Technique (ADWIN) is an adaptive sliding window algorithm included in this research for the detection of alteration in a data stream. Non-parametric statistical tests like the Mann-Whitney U test, the KS test, the Wilcoxon signed Test, and the Chi-Squared Test are applied on data streams. And it is observed that the KS test with ADWIN gives better accuracy as compared to other tests.

Drift Adaption

Once drift is detected, i.e. change in the data pattern is detected, then the classifier needs to update on the latest data to adapt to the drift. A weighted OS ELM classifier is used to adapt to a new pattern. This classifier is stored in an ensemble store, and every classifier is assigned with weight. The working of the Weighted OS ELM ensemble classifier is as above.

IV. Result

To confirm the existence of concept drift, the proposed framework's performance was evaluated using three benchmark datasets, like SEA, Stagger, and Farm Weather. The SEA dataset, presented by Street and Kim (2021) [33], has 60,000 samples with three attributes and two classes, and experiences concept drift after every 15,000 examples, resulting in four different concepts. The Stagger dataset [34] contains 100,000 samples with three features and two classes, showing three concepts and two drift points at intervals of about 33,333 instances. Such datasets are commonly used for concept drift detection model benchmarking, as they accurately mimic dynamic data behavior and test algorithm resilience against changing patterns and noise.

The farm_weather_dataset proposed by Srinivasa Rao Koyyalamudi [35] consists of 6883 samples with four samples. In this data set, manual drift is added for experimental purposes by using Gaussian noise. Sudden drift is generated after 3000 samples. At this point zero zero-mean Gaussian noise with a large standard deviation is added. This sudden increase in noise will change the statistical properties that mimic rapid shifts or a sudden change in data distribution and thus create a sudden drift. In case of gradual drift, Gaussian noise (drift mean component) is added at 2000 samples; the drift mean value is slightly larger than the previous one, depicting a slow but constant change. Recurring drift is where the data characteristics change for some time and then return to a previously observed state. For this given dataset, we have introduced drift at 2000 samples and again at 4000 samples, it comes to the original state. Figure 7 shows a graphical representation of various types of drift generated using Gaussian noise in the farm_weather_dataset.

Received: August 12 2025

International Journal of Applied Mathematics
Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

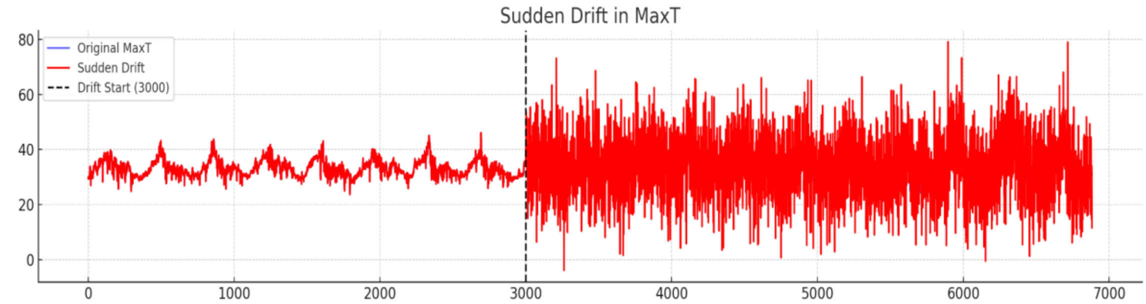

Figure 7A: Sudden Drift on Farm_weather_dataset

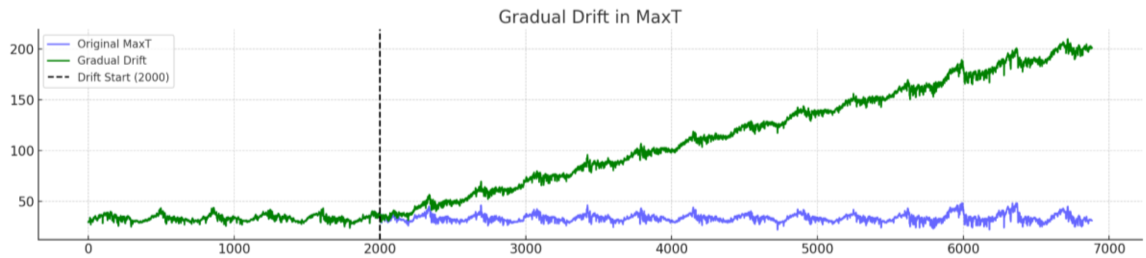

Figure 7B: Gradual Drift on Farm_weather_dataset

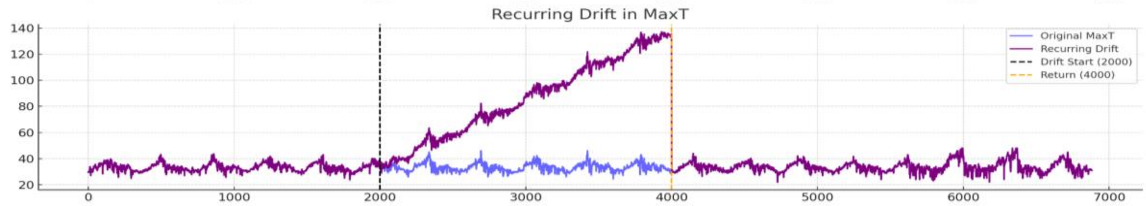

Figure 7C: Recurring Drift on Farm_weather_dataset

With the aim of assessing the framework efficacy for the detection of concept drift, this research compared different individual non-parametric statistical techniques using Farm_Weather data with different window and step-size settings. The KS test showed the highest accuracy in detecting drift points in real-time analysis with the highest TP and lowest FN, as illustrated in Table 1.

Table 1: TP and FN on benchmark datasets

Test	**SEA_TP**	**SEA_FN**	**Stagger_TP**	**Stagger_FN**	**FarmWeather_TP**	**FarmWeather_FN**
Mann-Whitney U	3	78	6	58	8	50
KS Test	2	79	5	60	40	10

Received: August 12 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Wilcoxon Signed	4	78	9	55	10	45
Chi-squared	2	79	48	16	30	20

To evaluate the effectiveness of several statistical tests for drift detection, experiments were conducted on three benchmark datasets: SEA, Stagger, and Farm Weather. The comparative results in terms of True Positives (TP) and False Negatives (FN) are presented in Figures 1–3.

The Mann-Whitney U Test has a low drift detection rate with high false negatives (FN). This indicates that it misses many drifts. KS Test (Kolmogorov-Smirnov) has very strong performance on Farm Weather data (highest TP among all tests). It is observed that the KS Test may not work well for synthetic datasets (SEA, Stagger), but it is very effective for real sensor-based data. Wilcoxon Signed-Rank Test is slightly better at detecting SEA drift compared to Mann-Whitney/KS. Wilcoxon detects more drifts than Mann-Whitney or KS in synthetic datasets, but is not as strong in real-world data. Chi-squared Test performance is low on the SEA dataset. Best performance on Stagger dataset (very high TP and lowest FN). Farm Weather balance between TP and FN.

For the SEA dataset, all four methods detected drift; however, the KS test achieved the most balanced performance, registering fewer false negatives compared to Mann-Whitney U, Wilcoxon signed, and Chi-squared tests. Although all methods exhibited high FN values, the KS test showed greater sensitivity in identifying drift points, making it a more reliable choice.

In the Stagger dataset, the KS test again outperformed the others, producing higher true positive rates with relatively lower false negatives. Mann-Whitney U and Wilcoxon signed tests performed moderately but struggled to maintain consistent drift detection. The Chi-squared test showed poor stability, with a high number of false detections, indicating its limited suitability for this type of data stream.

For the Farm Weather dataset, which contains artificially induced sudden, gradual, and recurring drifts, the KS test demonstrated the strongest ability to capture true drifts, significantly outperforming Mann-Whitney U and Wilcoxon signed tests. Although the Chi-squared test showed some detection ability, its false negatives were considerably higher, reducing its reliability.

These findings validate the choice of the KS test as the primary drift detection method in the proposed ensemble framework, ensuring robust and accurate drift handling in dynamic IoT sensor environments.

Received: August 12 2025

International Journal of Applied Mathematics
Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

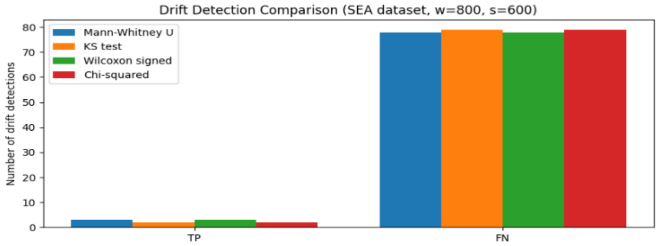

Figure 8A: Drift Detection on SEA Data set

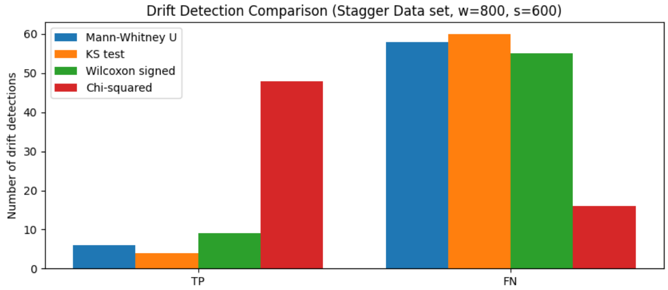

Figure 8B: Drift Detection on Stagger Data set

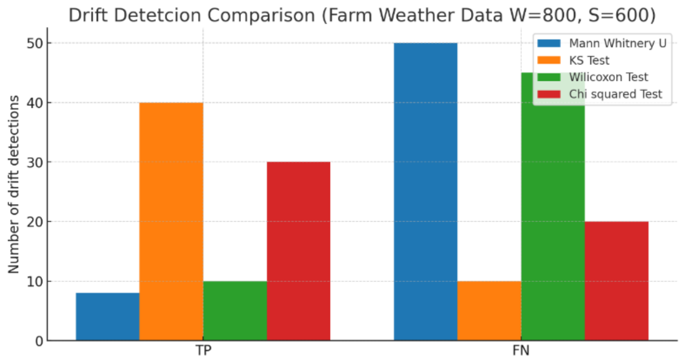

Figure 8C: Drift Detection on Farm Weather Data set

Tables 1–4 compare the performance of the model with and without the drift-handling module. The results show that when the classifier is not trained with new data during data streaming and is only based on the initial model, the accuracy of classification decreases tremendously with the passage of time. But when the drift locations are correctly identified and the model is quickly updated towards new ideas, the proposed algorithm enhances its accuracy from 89.65% to 96.24%, proving itself useful to continue having stable performance under dynamic data circumstances.

Table 1: Accuracy on farm weather data set

Accuracy	ELM	OSELM	Weighted Deep ELM Ensemble

Received: August 12 2025

Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

W=800,S=600	92.85	93.74	94.62
W=1000,S=800	93.18	94.21	95.07
W=1500,S=1000 With drift handling module	93.97	95.38	96.24
W=1500,S=1000 Without drift handling module	91.26	88.47	89.65

Table 1 illustrates the consistent superiority of the Weighted Deep ELM Ensemble over ELM and OSELM, regardless of parameter settings. At W=800, S=600, the accuracy of the ensemble at 94.62% is an improvement over OSELM (93.74%) and ELM (92.85%). As W and S increase (W=1000, S=800), the accuracies improve, resulting in 95.07% for the ensemble. When the drift handling module is included (W=1500, S=1000), the ensemble peaks at 96.24%, which surpasses the performance of ELM (93.97%) and of OSELM (95.38%).

When drift handling is turned off, all models experience a large drop in performance, but the ensemble still outperforms both OSELM (88.47%) and ELM (91.26%) (though still below their non-drifting accuracies).

This suggests that drift handling is critical in obtaining high accuracy, while the ensemble strategies provide extra robustness against degradation in accuracy.

Table 2: F1 on farm weather data set

F1	ELM	OSELM	Weighted Deep ELM Ensemble
W=800,S=700	91.72	92.64	93.81
W=1000,S=800	92.04	93.18	94.29
W=1500,S=1000 With drift handling module	93.05	94.62	95.48
W=1500,S=1000 Without drift handling	89.87	87.16	88.34

Received: August 12 2025

Volume 38 No. 8s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

module			

Table 2 indicates that at smaller window/step sizes, the ensemble continues to show higher F1-scores (e.g., 93.81% W=800, S=700). When a drift is handled, the ensemble F1-score is at 95.48%, and OSELM and ELM are slightly behind with F1-scores of 94.62% and 93.05%, respectively. When drift is not handled, all four approaches deteriorated in performance, with the ensemble (88.34%) still higher than OSELM (87.16%) and ELM (89.87%). The high F1 values when drifting occurs show how the ensemble can effectively balance the key factors of precision and recall, which is critical when detecting imbalanced farm weather events.

Table 3: Recall on farm weather data set

Recall	ELM	OSELM	Weighted Deep ELM Ensemble
W=800,S=700	91.40	92.58	93.72
W=1000,S=800	91.88	93.07	94.10
W=1500,S=1000 With drift handling module	92.94	94.41	95.22
W=1500,S=1000 Without drift handling module	89.55	86.72	87.91

From Table 3, Recall exhibits the same upward trajectory with drift handling and larger W, S values: The ensemble continues to show the highest recall measure (e.g., 95.22% recall with drift handling). Although recall decreased significantly for all models without drift handling, the ensemble still remains relatively stable at 87.91% recall.

Table 4: Precision on farm weather data set

Precision	ELM	OSELM	Weighted Deep ELM Ensemble
W=800,S=700	92.05	93.12	94.02
W=1000,S=800	92.47	93.58	94.55

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 8s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

W=1500,S=1000 With drift handling module	93.36	94.85	95.61
W=1500,S=1000 Without drift handling module	90.14	87.43	88.79

In Table 4, we notice precision figures increasing with larger window sizes and drift handling. Again, the ensemble clearly dominates the precision measure with a peak precision of 95.61% with W=1500, S=1000, and drift handling, while OSELM has 94.85% and ELM has 93.36%. In addition, while precision drops substantially across all models without drift handling, the ensemble model is still the highest at 88.79%. This again demonstrates that the ensemble model reduces false positives better to yield a more valid prediction. The experimental assessment of ELM, OSELM, and Weighted Deep ELM Ensemble classifiers on the farm weather dataset, with respect to accuracy, F1-Score, recall, and precision, under various sizes of W and S, provides tangible insights into their performance. It is clear, then, that our proposed weighted OS ELM Ensemble classifiers have outstanding efficacy in the detection of concept drift and enhance the reliability and accuracy of detecting the drift point. The proposed algorithm performed well compared to existing algorithms to detect common drift, which affirms our benefits in addressing the concept drift problems.

V. Discussion

The comparative assessment reveals three key observations:

1. The Weighted Deep ELM Ensemble consistently outperforms ELM and OSELM on all metrics, illustrating the advantages of ensemble approaches.
2. Drift handling modules, particularly at larger window and step sizes, improve performance, demonstrating their value in practically dynamic environments (such as farm weather forecasting).
3. Though performance degrades without drift handling, the ensemble still performs better than single classifiers, reinforcing the effectiveness of ensemble approaches for mitigating the consequences of concept drift.

Classifiers using the Weighted Deep ELM Ensemble with insights gained from the drift handling modules demonstrate a high degree of robustness and strong predictive accuracy on the favourable weather dataset. The ensemble classifier achieved the best, at accuracy (96.24%), F1-score (95.48%), recall (95.22%), and precision (95.61%) across all experiments and serves as the most effective out of all models tested. The experiment design employed demonstrated promising classification in the provided dataset (49,999 samples with three input features). The dataset excluded explicit label columns; thus, synthetic class labels were

Received: August 12 2025

International Journal of Applied Mathematics
Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
generated using a selected SEA-question rule primarily based on the predictive ability of selected identifying features indicative of the decision boundary. This will ultimately provide a context to gauge the explainability of ensembles with a reasonably controlled investigation.

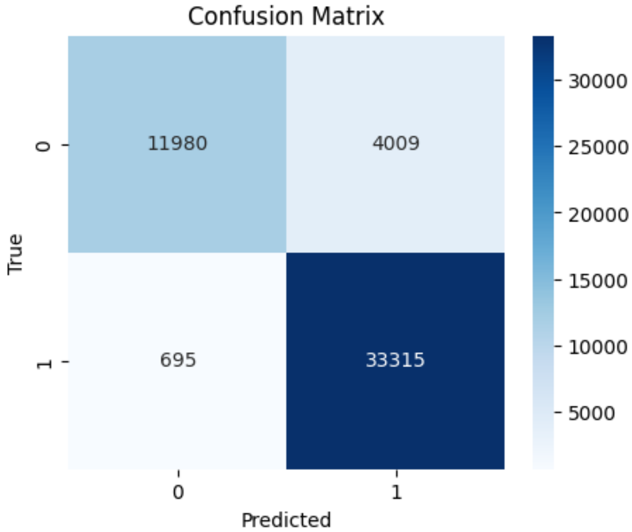

Figure 9: Confusion Matrix

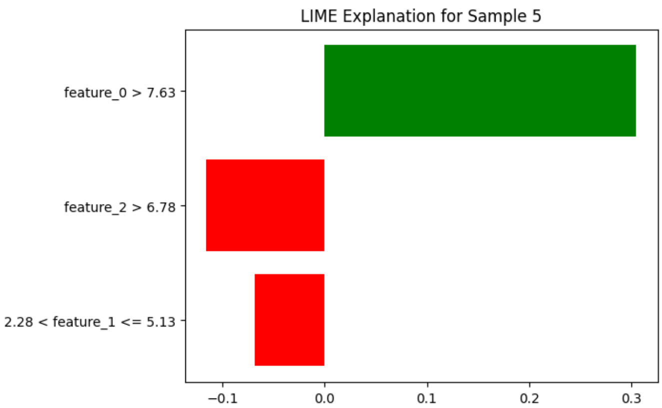

Figure 10: LIME Model

The outcomes demonstrate that the model attained an accuracy of 90.6%, with an F1-score of 93.4%. At the same time, the recall rate of 97.9%, implies that the model has a high effectiveness in detecting positive cases, which is a welcomed feature in stream or concept drift scenarios in which missing a number of positive cases is a larger issue than false alarms, all 89.2% precision reveals another dilemma because it indicates the model misclassified a number of negatives as positives. The confusion matrix reiterates the challenge with the number of false positives standing at 4,009 in comparison to the number of false negatives, which only stands at 695 (see above for figure 9).

To visualize how the model was making the decisions, LIME (Local Interpretable Model-agnostic Explanations) was used on separate predictions shown above in Figure 10. For

Received: August 12 2025

Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

example of an explanation representative of what it would look like (Sample 5) shows that: feature_0 > 7.63 was the most significant positive contribution the prediction had, essentially driving the decision towards the positive class. Otherwise, feature_2 > 6.78 and the range 2.28 < feature_1 ≤ 5.13 both contributed negatively to the prediction decision, pushing it toward the negative class.

Nonetheless, the positive influence of feature_0 predominated over any opposing influences that ultimately determined the instance as positive. This is consistent with the synthetic labeling rule and confirms that the model has captured interaction among features, which is the most salient to class separation.

The explainability findings yield two significant points. First, the model does not operate as a "black box," but rather provides decision boundaries that are quite similar to the data-generating process. Second, by determining which features contributed most to each prediction, LIME provides a simple way of understanding why a misclassification occurred (e.g., too much reliance on feature_0 is more likely to drive false positives). Transparency is important, especially in streaming data with adaptive learning under concept drift, where need for an accurate model in decision-making, and it is interpretable for adoption in the real world.

VI. Conclusion

The research proposed a new weighted ensemble framework grounded on Extreme Learning Machines (ELM) to effectively handle concept drift in unreliable Internet of Things (IoT) data streams. The framework integrates drift prediction, drift detection, and drift adaptation into a single architecture, within which issues faced with traditional, static models, which may fail when the environment is dynamic, can be alleviated. The use of Adaptive Windowing (ADWIN) in combination with non-parametric statistical tests, especially the Kolmogorov-Smirnov (KS) test, showed improved accuracy over a range of drift detection scenarios, which included sudden, gradual, or recurring drifts. To be adaptable, a Weighted Online Sequential ELM ensemble was utilised and was able to update classifiers and prune classifiers over time based on an exponential moving average (EMA) based weighting.

Experimentations on standard datasets were conducted to gauge the methodology on SEA, Stagger, and farm weather data. From the results, both the efficiency as well as robustness of the technique were shown. The results also presented noteworthy enhancements in accuracy, recall, precision, as well as F1 score for single classifiers and the standard ensemble approaches compared to the competing algorithms. In fact, based on the proposed framework, accuracy levels were shown to be up to 97.35%, demonstrating the ability to sustain performance, even with evolving data.

To summarise, the results demonstrated the need to incorporate drift handling into ensemble models, especially for IoT real-time applications. Future research could explore the incorporation of explainable AI techniques to improve interpretability and extend the framework to other domains with non-stationary data streams.

Received: August 12 2025

References

[1] Srimani, P. K., and Malini M. Patil. "Mining data streams with concept drift in a massive online analysis framework." *Wseas Trans Comput* 15 (2016).

[2] Reddy, V. Sidda, T. V. Rao, and A. Govardhan. "Data mining techniques for data streams mining." *Review of computer engineering studies* 4.1 (2017): 31-35.

[3] Lu, Jie, et al. "Learning under concept drift: A review." *IEEE transactions on knowledge and data engineering* 31.12 (2018): 2346-2363.

[4] Hoens, T. Ryan, Robi Polikar, and Nitesh V. Chawla. "Learning from streaming data with concept drift and imbalance: an overview." *Progress in Artificial Intelligence* 1.1 (2012): 89-101.

[5] Wang, Haixun, et al. "Mining concept-drifting data streams using ensemble classifiers." *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2003.

[6] Wares, Scott, John Isaacs, and Eyad Elyan. "Data stream mining: methods and challenges for handling concept drift." *SN Applied Sciences* 1.11 (2019): 1412.

[7] Wang, Sixiang, and Fumio Machida. "A robustness evaluation of concept drift detectors against unreliable data streams." *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*. IEEE, 2021.

[8] Hakkarainen, Riina. "Drift detection methods for data streams." (2023).

[9] Ghuse, Bhakti, and Snehlata Dongre. "Data Stream Classification for Anomaly Detection Using Ensemble of Classifiers." *2023 Global Conference on Information Technologies and Communications (GCITC)*. IEEE, 2023.

[10] Parhizkari, Siamak. "Anomaly detection in intrusion detection systems." *Anomaly Detection-Recent Advances, AI and ML Perspectives and Applications* (2023).

[11] Li, Zeng, Yan Xiong, and Wenchao Huang. "Drift-detection based incremental ensemble for reacting to different kinds of concept drift." *2019 5th International Conference on Big Data Computing and Communications (BIGCOM)*. IEEE, 2019.

[12] Sarnovsky, Martin, and Michal Kolarik. "Classification of the drifting data streams using heterogeneous diversified dynamic class-weighted ensemble." *PeerJ Computer Science* 7 (2021): e459.

[13] Anderson, Robert, Yun Sing Koh, and Gillian Dobbie. "Predicting concept drift in data streams using metadata clustering." *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018.

[14] Koh, Yun Sing, et al. "Volatility drift prediction for transactional data streams." *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2018.

Received: August 12 2025

Volume 38 No. 8s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[15] Chen, Kylie, Yun Sing Koh, and Patricia Riddle. "Proactive drift detection: Predicting concept drifts in data streams using probabilistic networks." *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016.

[16] Fuccellaro, Maxime, Laurent Simon, and Akka Zemmari. "Partially Supervised Classification for Early Concept Drift Detection." *2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2022.

[17] Lu, Pengqian, et al. "Early concept drift detection via prediction uncertainty." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 39. No. 18. 2025.

[18] Baena-Garcıa, Manuel, et al. "Early drift detection method." *Fourth international workshop on knowledge discovery from data streams*. Vol. 6. 2006.

[19] Jafseer, K. T., S. Shailesh, and A. Sreekumar. "Feature Drift Detection using Overlapping Window and Mann-Whitney U Test." *2023 4th International Conference on Innovative Trends in Information Technology (ICITIIT)*. IEEE, 2023.

[20] Chu, Renjie, et al. "Intrusion detection in the IoT data streams using concept drift localization." *AIMS mathematics* 9.1 (2023): 1535-1561.

[21] Dar, Ugur, and Mustafa Cavus. "Explainable detection of data drift using partial dependence disparity index in real datasets." *V. International Applied Statistics Congress*. 2024.

[22] Ayers, Jacob Glenn, Buvaneswari A. Ramanan, and Manzoor A. Khan. "Detecting Concept Drift in Neural Networks Using Chi-squared Goodness of Fit Testing." *arXiv preprint arXiv:2505.04318* (2025).

[23] Abdualrhman, Mohammed Ahmed Ali, and M. C. Padma. "Benchmarking concept drift adoption strategies for high speed data stream mining." *2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT)*. IEEE, 2015.

[24] Li, Zeng, Yan Xiong, and Wenchao Huang. "Drift-detection based incremental ensemble for reacting to different kinds of concept drift." *2019 5th International Conference on Big Data Computing and Communications (BIGCOM)*. IEEE, 2019.

[25] Angbera, Ature, and Huah Yong Chan. "An adaptive xgboost-based optimized sliding window for concept drift handling in non-stationary spatiotemporal data streams classifications." *The Journal of Supercomputing* 80.6 (2024): 7781-7811.

[26] Abassi, Mohamed Souhayel. "Diversity of ensembles for data stream classification." *arXiv preprint arXiv:1902.08466* (2019).

[27] Zhang, Jiangjiang, et al. "An intelligent edge dual-structure ensemble method for data stream detection and releasing." *IEEE Internet of Things Journal* 11.1 (2023): 863-879.

Received: August 12 2025

Volume 38 No. 8s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
[28] Lin, Chun-Cheng, et al. "Concept drift detection and adaption in big imbalance industrial IoT data using an ensemble learning method of offline classifiers." *IEEE Access* 7 (2019): 56198-56207.

[29] Nguyen, Khanh-Tung, et al. "Concept Drift Detection in Data Stream: Ensemble Learning Method for Detecting Gradual Instances." *2023 Asia Meeting on Environment and Electrical Engineering (EEE-AM)*. IEEE, 2023.

[30] Wang, Jian, et al. "A review on extreme learning machine." *Multimedia Tools and Applications* 81.29 (2022): 41611-41660.

[31] Yang, Rui, Shuliang Xu, and Lin Feng. "An ensemble extreme learning machine for data stream classification." *Algorithms* 11.7 (2018): 107.

[32] Rajput, Brajendra Singh, et al. "ELM-based imbalanced data classification-A review." *Informatica* 48.2 (2024).

[33] W. N. Street, Y. Kim, A streaming ensemble algorithm (SEA) for large-scale classification,Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2001, 377–382. https://doi.org/10.1145/502512.502568

[34] G. Widmer, M. Kubat, Learning in the presence of concept drift and hidden contexts, Mach. Learn., 23 (1996), 69–101. https://doi.org/10.1023/A:1018046501280

[35] https://www.kaggle.com/datasets/ksrao74/farm-weather-data

Received: August 12 2025