An Ensemble Approach to Improve the Performance of Real Time Data Stream Classification

Dhara Joshi
Department of Computer Engineering, Marwadi University, India
dhara.joshi@marwadieducation.edu.in (corresponding author)

Madhu Shukla
Department of Computer Science and Engineering – Artificial Intelligence, Machine Learning and Data Science, Marwadi University, India
madhu.shukla@marwadieducation.edu.in

Received: 30 July 2024 | Revised: 29 August 2024 | Accepted: 8 September 2024

ABSTRACT

In the era of the Internet of Things (IoT), data stream mining has gained importance to make accurate and profitable decisions. Various techniques are used to gain insight into data streams, including classification, clustering, pattern mining, etc. Data are subject to changes over time. When this happens, predictive models that assume a static link between input and output variables may perform poorly or even degrade, which is called concept drift. This study proposes an ensemble architecture designed to improve performance and effectively detect concept drift in stream data classification. Using an ensemble approach, the proposed architecture incorporates three classifiers to improve accuracy and robustness against concept drift. The proposed architecture provides drift detection that ensures the model's continued performance by enabling it to be quickly modified to changing data distributions. Through comprehensive testing, the performance of the proposed algorithm was compared with existing methods, and the results demonstrate its superiority in terms of classification accuracy, precision, and recall and drift detection capabilities.

Keywords-data stream; mining; classification; challenges; concept drift; ensemble

I. INTRODUCTION

A large amount of real-time data streams are produced by technical advancements in Internet of Things (IoT) devices, smart gadgets, dynamic networks, trade data, and more [1]. These diverse streaming data sources provide real-time data that need quick processing. The transmission of continuous data streams to a processing framework is called online data stream processing. This framework analyses and extracts meaningful information instantly, allowing quick decision-making and actions in sensor networks, automated trading systems, real-time monitoring, communications, and alerting applications. Traditional databases first store data on available resources and then process it whenever needed. However, data streams require real-time processing because instances are not stored. Instances can be referred to as useful insights obtained from continuous infinite data. A data stream typically has the following qualities: volume, velocity, variety, and unboundedness. Velocity means fast data generation and real-time analysis. Since volume refers to large data generated continuously, data instances are created continuously. Variety indicates data heterogeneity, meaning that data can take many forms depending on their source. The biggest issue is that stream data are continuous and endless. It can also be assumed that there is a need to process limitless data. Concept drift, the statistical change in the target variable predicted by the model, is a major concern in data stream processing [2]. This shift can be caused by user behavior, changes in data generation processes, or changes in the external environment. Models must adapt to changing data patterns to remain accurate and relevant in real-world applications. Concept drift reduces model accuracy during historical instance learning because the distributions in the prediction object and the learning samples might fluctuate. Concept drift can be progressive, steady, or rapid, depending on the pace of change [3]. A sudden drift occurs when the data distribution changes abruptly, generally due to unplanned events. Gradual drift is the slow and steady adjustment of change variables over time, sometimes owing to seasonal change or other reasons. Thus, models must be able to identify incremental change and transition without sudden changes. This is similar to progressive drift, except that it involves a fast shift that always causes enduring data pattern modification and is always followed by a catch-up phase. This study aims to propose a new strategy to govern the first dramatic change and the slow process that follows. Positional drift causes performance to fall progressively when the classifier is out of sync with the data. However, rapid change will probably decline the models' predictive ability faster. When the data keeps changing, acute incremental drift causes a dramatic performance loss that oscillates between the stabilization factor and the decline.

This study focuses on ensemble strategies, the sliding-window method [4], and previous drift detection methods [5-9] to handle concept drift. A sliding-window method limits the classifier to learning from the latest examples, excluding prior samples as the window slides. This strategy mitigates stale data while retraining a classifier. A classifier with a small window size responds differently and more sensitively to concept drift, but its precision may be compromised. A classifier with a larger window size responds better and is more stable, but its flexibility to respond to rapid changes may be compromised. In [5, 7] interactive trigger systems were proposed for drift detection, offering window size heuristics to meet study objectives. This method directly measures concept drift by measuring classifier activity and data dispersion. When an alert level is reached, an immediate attempt is made to gather more instances to ensure that the model learns from the recognized concept change. Upon detecting a concept drift, the classifier renews itself using the latest data samples. Depending on data flow, ensemble approaches can be block-based or online [8]. Online ensembles dynamically alter the core classifier to react quickly to abrupt drifts for each instance. This allows the model structure and ensemble classifier to adapt to changes in the data stream. Through recurrent weighting, the block-based ensemble efficiently responds to progressive idea drift by grouping data into fixed-size blocks. Its response to drift inside blocks is restricted, hence block size calibration is necessary. Smaller blocks can react fast to abrupt changes, but they may slow performance and increase computing costs during stable times, since they consume old data blocks before drift. The sliding-window approach can track data streams and discover concept drift. This lets a block-based ensemble easily capture data patterns over time using a multilayer sliding window.

Classifying continuous or accelerated data streams by hand is expensive and difficult. Active learning is investigated to choose worthwhile examples for labeling in a cost-effective manner. Current algorithms [9-12] are inflexible because they have difficulty adjusting to changing data sources. Drift can be reduced by allocating greater funding for labeling to improve classification accuracy. Making the switch to an on-demand active learning strategy enables quicker labeling in reaction to concept drift, accelerating the absorption of new concepts and enhancing instance selection accuracy, all of which help to control the increasing labeling expenses. Some studies investigated ensemble classifiers to understand dynamic data streams [13-19]. Data stream ensemble learning is block-based or online. Each event is analyzed independently in incremental online ensemble learning [20]. Two key algorithms are online boosting and bagging [21, 22]. Class imbalance and concept drift management approaches evaluate classification performance in unbalanced and non-stationary environments [4]. In addition, block-based and online learning hybrid systems are examined, along with online bagging/boosting tactics and active learning ways to improve labeling. Diverse methods can improve model performance in dynamic contexts. In [2], an active learning architecture was proposed for stream data. The preliminary structure of the models in this study integrates both ensemble learning [23] and active learning strategies to optimally adapt to concept drift. The method ensures high classification performance when training examples change over time by selecting samples for labeling and adding or replacing models in the ensemble of active classifiers. In this study, the management of concept drift in streaming data is addressed by employing the most practical techniques for handling it. In [9], an investigation was conducted on class imbalance learning with concept drift. This study focused on the challenges posed by data streams and evolving concepts, presenting methods to adjust learning strategies. The objective was to improve classification performance by examining various methods. This study provides insight into managing the complex interplay between concept drift and class imbalance in online learning environments. Accuracy Updated Ensemble (AUE), which employs component classifiers and dynamic updates, outperformed Accuracy Weighted Ensemble (AWE) [4]. Using up-to-date data blocks, AUE combines block-based methods with the learning features of the Hoeffding tree, assigning weights to classifiers based on their error rates. Since the highest weight is given to the newest base classifier, which acts as the most recent knowledge base, AUE can quickly adjust its behavior in concept drift occurrences. With online incremental learning, on-rule learning, and block-based ensembles, AUE effectively tackles the problem of concept drift. Active learning can also prevent or reduce the problem of scarcity of labeled data and concept drift using knowledge from the past. In [22], a comparative study on single and ensemble classifier techniques was presented for data stream classification. This analysis highlighted the pros and cons of different approaches and offered insight into their relative effectiveness. By providing insightful information on the choice and application of classification methods in contexts with dynamic data streams, this study can help practitioners and researchers make well-informed decisions.

II. PROPOSED FRAMEWORK

Figure 1 shows the proposed Active Learning Ensemble (ALEnsemble) framework to develop data stream topologies. Its process is divided into five steps as follows:

- Step 1: Divide the data stream (S) into discrete data blocks ($X_1, X_2, \ldots, X_n$). Each data block represents one of the i occurrences ($i = 1, 2, 3, \ldots, n$) in stream data S . The creation of two sub-data streams occurs when the current data block is X_n (where $n \geq D$ represents the size/window of the data block). A sliding window examines and analyzes a subset of a data stream in a fixed-size window that travels over the stream to keep the focus on the latest data.
- Step 2: Unlabeled data is labeled using active learning. The most common active learning method is to select the most valuable data points from an unlabeled sample to categorize. The ensemble classifier is trained on labeled data.

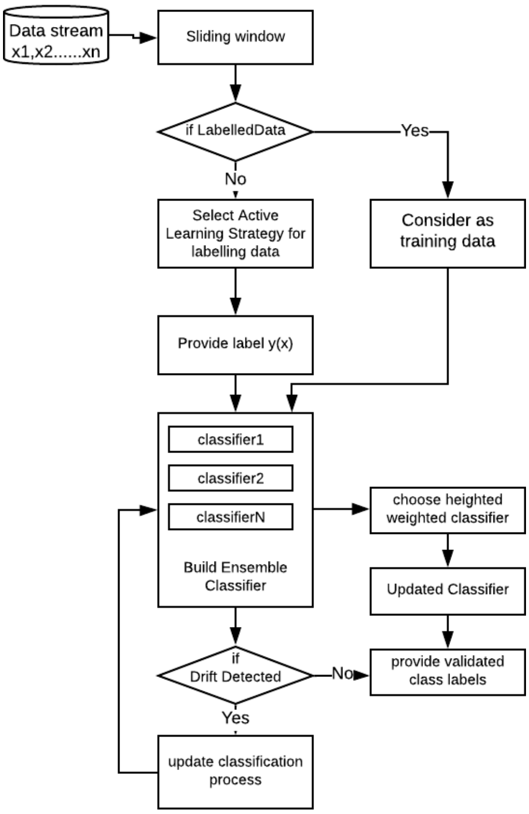
Data stream
x1,x2......xn
Sliding window
if LabelledData
Yes
No
Select Active Learning Strategy for labelling data
Consider as training data
Provide label y(x)
classifier1
classifier2
classifierN
Build Ensemble Classifier
choose heighted weighted classifier
Updated Classifier
if Drift Detected
No
provide validated class labels
Yes
update classification process

Fig. 1. Proposed framework.

- Step 3: An ensemble classifier uses the tagged data to predict future data. An ensemble classifier combines the predictions of many models to improve accuracy. ALEnsemble includes the Hoeffding tree classifier, the adaptive tree for concept drift, and OzaBag with ADWIN.
- Step 4: A highly weighted classifier is selected using an ensemble technique. This is more complicated than assigning weights to classifiers according to their height in a hierarchical ensemble, which awards classifiers with higher relevance at specific levels to improve prediction accuracy. After that, it labels the class and updates the classifier.
- Step 5: If concept drift is detected using various methods, such as DDM, EDDM, or ADWIN, the classifier is retrained with the most recent data, adapts its parameters to the new data distribution, or integrates new data into an incremental learning framework to maintain its accuracy.

III. EXPERIMENTAL STUDY

A. Dataset

The experimental study used datasets and various stream generators (Airlines, SEA, LED) [11]. Each of these datasets comprises a large number of instances with many attributes.

- Airlines is a real-world dataset consisting of 539,383 instances and eight attributes (Airline, Flight number, AirportFrom, AirportTo, DayOfWeek, Time, Length, and Delay).
- SEA is a synthetic dataset. With three real-valued qualities that fall between 0 and 10, this concept generator generates genuine abrupt concept drifts. There are 1,000,000 instances in the stream, 10,000 for each concept, having different concept function thresholds.
- LED is a synthetic dataset. This concept generator stimulates real, abrupt concept drifts. The stream consists of 1,000,000 instances, with 17 attributes.

B. Evaluation Metrics

Evaluating a model's performance in an experiment is essential to comprehend its efficacy and reliability. Various aspects of a model's performance are quantified using different evaluation measures. The key assessment metrics for experimental evaluation are as follows:

- Accuracy: When dealing with classification issues, accuracy refers to the total number of right predictions that a model has generated across all different kinds of predictions.

  $$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}$$

  where T_p is true positives, T_n is true negatives, F_p is false positives, and F_n is false negatives.

- Precision is the proportion of correct positive outcomes to the number of positive predictions made by the classifier.

  $$Precision = \frac{Tp}{Tp+Fp}$$

- Recall is determined by considering the count of valid positive results and then dividing it by the overall count of all samples that should have been recognized as positive.

  $$Recall = \frac{Tp}{Tp+Fn}$$

This section describes the experimental evaluation of the proposed ensemble data stream classification technique. This experiment addressed content drift through active learning and ensemble methods. The ALEnsemble was compared to the Heoffding tree, k-NN, naive Bayes, adaptive random forest, leverage bagging, and active learning. These algorithms were evaluated for accuracy, precision, and recall [13]. The Massive Online Analysis (MOA) open-source data stream mining software was used, which contains implemented data stream mining algorithms and assessment methods. The above-described datasets were used to evaluate the classification methods on concept-drifted data. MOA uses Brzezinski and Stefanowski programs [4] to produce synthetic data and offers real-time data downloads.

C. Results and Discussion

Figure 2 shows that the proposed ALEnsemble approach outperformed other algorithms. The accuracy of ALEnsemble was highest across the three datasets. Figure 3 shows the comparative results of ALEnsemble throughout real-time data streams with concept drift, indicating a little improvement in precision.

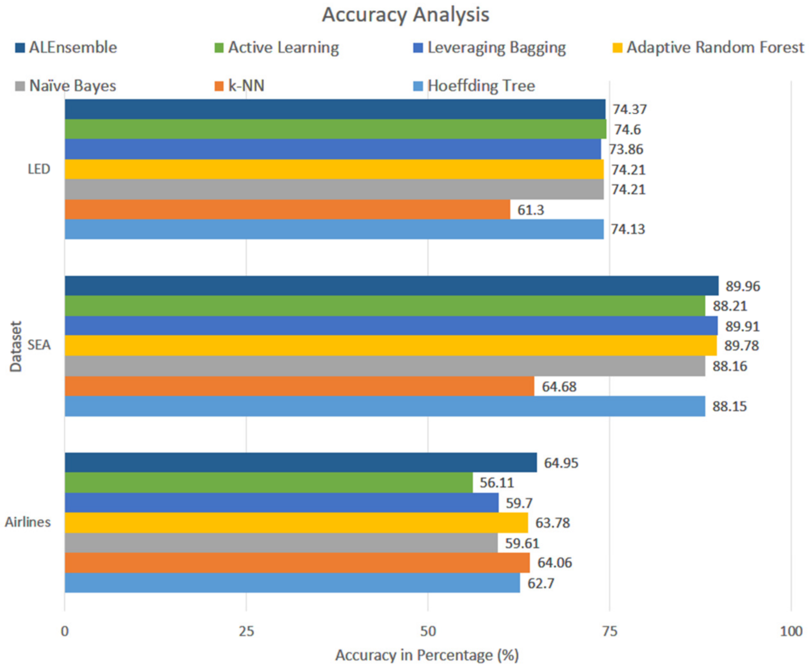

Fig. 2. Comparative analysis: Accuracy.

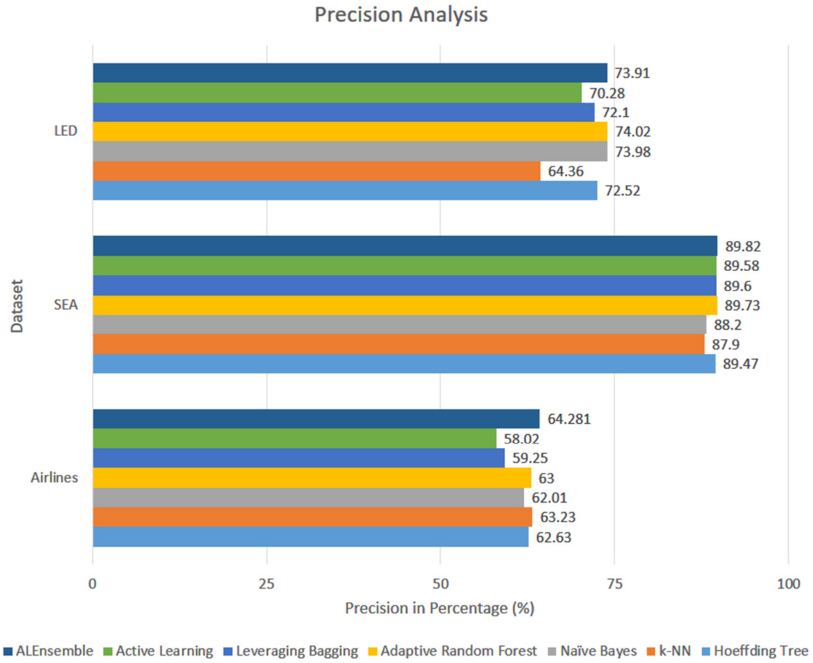

Fig. 3. Comparative analysis: Precision.

Figure 4 illustrates that, compared to other methods, the proposed ALEnsamble algorithm performed well for the recall metric. In addition, the adaptive random forest was effective with encouraging results.

IV. CONCLUSION AND FUTURE PROSPECTS

This paper proposed an ensemble method to identify concept drift across datasets and compare it with previous techniques. Three datasets were used to evaluate multiple categorization systems' accuracy, precision, and recall. The results imply that the proposed ALEnsemble is an effective approach and looks promising compared to other algorithms. Statistical tests, such as the Page-Hinkley and Friedman tests, can be used to verify the robustness of these algorithms. In addition, more methods should be investigated to make classifiers more resilient to uneven and noisy data streams, which are typical in real-world data contexts.

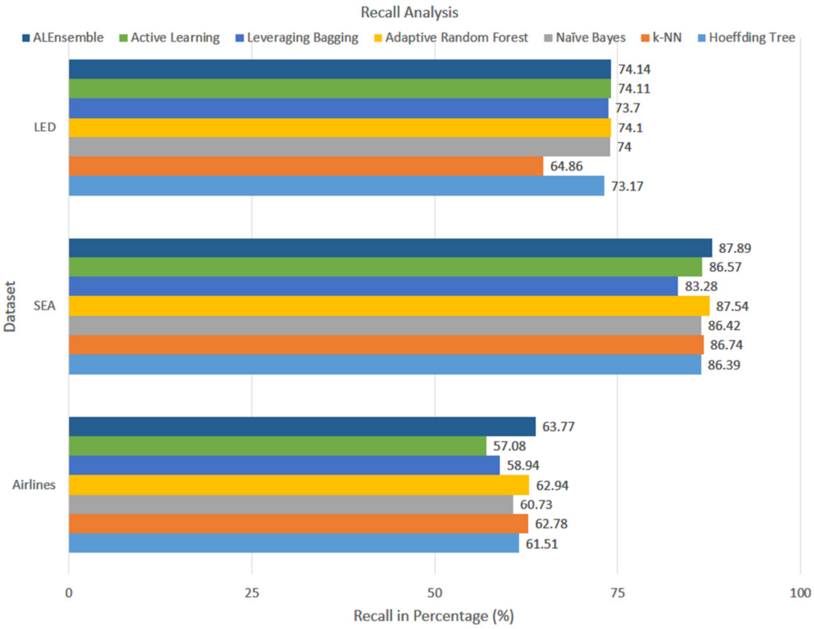

Fig. 4. Comparative analysis: Recall.

REFERENCES

[1] J. Gama, J. Aguilar-Ruiz, and R. Klinkenberg, "Knowledge discovery from data streams," *Intelligent Data Analysis*, vol. 12, no. 3, pp. 251–252, Jan. 2008, https://doi.org/10.3233/IDA-2008-12301.

[2] J. Shan, H. Zhang, W. Liu, and Q. Liu, "Online Active Learning Ensemble Framework for Drifted Data Streams," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 2, pp. 486–498, Oct. 2019, https://doi.org/10.1109/TNNLS.2018.2844332.

[3] I. Khamassi, M. Sayed-Mouchaweh, M. Hammami, and K. Ghédira, "Discussion and review on evolving data streams and concept drift adapting," *Evolving Systems*, vol. 9, no. 1, pp. 1–23, Mar. 2018, https://doi.org/10.1007/s12530-016-9168-2.

[4] D. Brzezinski and J. Stefanowski, "Reacting to Different Types of Concept Drift: The Accuracy Updated Ensemble Algorithm," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 1, pp. 81–94, Jan. 2014, https://doi.org/10.1109/TNNLS.2013.2251352.

[5] S. Wang, L. L. Minku, and X. Yao, "A Systematic Study of Online Class Imbalance Learning With Concept Drift," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4802–4821, Jul. 2018, https://doi.org/10.1109/TNNLS.2017.2771290.

[6] M. M. S. Shukla and M. K. R. Rathod, "Stream data mining and comparative study of classification algorithms," *Algorithms*, vol. 3, no. 1, pp. 163–168, 2013.

[7] M. Baena-Garcıa *et al.*, "Early Drift Detection Method ?," in *Proceedings of the 4th International Workshop in Knowledge Discovery Data Streams*, 2006.

[8] A. Masrani, M. Shukla, and K. Makadiya, "Empirical Analysis of Classification Algorithms in Data Stream Mining," in *International Conference on Innovative Computing and Communications*, Singapore, 2020, pp. 657–669, https://doi.org/10.1007/978-981-15-5113-0_53.

[9] D. Joshi and M. Shukla, "A Consolidated Study On Advanced Classification Techniques Used On Stream Data," in *2023 IEEE 11th Region 10 Humanitarian Technology Conference (R10-HTC)*, Rajkot, India, Oct. 2023, pp. 614–619, https://doi.org/10.1109/R10-HTC57504.2023.10461809.

[10] J. N. Adams, S. J. van Zelst, T. Rose, and W. M. P. van der Aalst, "Explainable concept drift in process mining," *Information Systems*, vol. 114, Mar. 2023, Art. no. 102177, https://doi.org/10.1016/j.is.2023.102177.

[11] A. Bifet, G. Holmes, and B. Pfahringer, "Leveraging Bagging for Evolving Data Streams," in *Machine Learning and Knowledge Discovery in Databases*, Barcelona, Spain, 2010, pp. 135–150, https://doi.org/10.1007/978-3-642-15880-3_15.

[12] S. G. T. de C. Santos, P. M. Gonçalves Júnior, G. D. dos S. Silva, and R. S. M. de Barros, "Speeding Up Recovery from Concept Drifts," in *Machine Learning and Knowledge Discovery in Databases*, Nancy, France, 2014, pp. 179–194, https://doi.org/10.1007/978-3-662-44845-8_12.

[13] I. Frías-Blanco, A. Verdecia-Cabrera, A. Ortiz-Díaz, and A. Carvalho, "Fast adaptive stacking of ensembles," in *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, Pisa, Italy, Apr. 2016, pp. 929–934, https://doi.org/10.1145/2851613.2851655.

[14] D. Brzezinski and J. Stefanowski, "Ensemble Classifiers for Imbalanced and Evolving Data Streams," in *Data Mining in Time Series and Streaming Databases*, vol. 83, World Scientific, 2017, pp. 44–68.

[15] D. Brzeziński and J. Stefanowski, "Accuracy Updated Ensemble for Data Streams with Concept Drift," in *Hybrid Artificial Intelligent Systems*, Wroclaw, Poland, 2011, pp. 155–163, https://doi.org/10.1007/978-3-642-21222-2_19.

[16] B. Krawczyk, B. Pfahringer, and M. Wozniak, "Combining active learning with concept drift detection for data stream mining," in *2018 IEEE International Conference on Big Data (Big Data)*, Seattle, WA, USA, Dec. 2018, pp. 2239–2244, https://doi.org/10.1109/BigData.2018.8622549.

[17] W. Fan, Y. Huang, H. Wang, and P. S. Yu, "Active Mining of Data Streams," in *Proceedings of the 2004 SIAM International Conference on Data Mining (SDM)*, Apr. 2004, pp. 457–461, https://doi.org/10.1137/1.9781611972740.46.

[18] I. Žliobaitė, A. Bifet, B. Pfahringer, and G. Holmes, "Active Learning With Drifting Streaming Data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 1, pp. 27–39, Jan. 2014, https://doi.org/10.1109/TNNLS.2012.2236570.

[19] Y. Wang, M. M. Rosli, N. Musa, and F. Li, "Multi-Class Imbalanced Data Classification: A Systematic Mapping Study," *Engineering, Technology & Applied Science Research*, vol. 14, no. 3, pp. 14183–14190, Jun. 2024, https://doi.org/10.48084/etasr.7206.

[20] A. S. Alkarim, A. S. A.-M. Al-Ghamdi, and M. Ragab, "Ensemble Learning-based Algorithms for Traffic Flow Prediction in Smart Traffic Systems," *Engineering, Technology & Applied Science Research*, vol.

14, no. 2, pp. 13090–13094, Apr. 2024, https://doi.org/10.48084/etasr.6767.

[21] W. Xu, F. Zhao, and Z. Lu, "Active learning over evolving data streams using paired ensemble framework," in *2016 Eighth International Conference on Advanced Computational Intelligence (ICACI)*, Chiang Mai, Thailand, Feb. 2016, pp. 180–185, https://doi.org/10.1109/ICACI.2016.7449823.

[22] B. Ghuse and S. Dongre, "Data Stream Classification for Anomaly Detection Using Ensemble of Classifiers," in *2023 Global Conference on Information Technologies and Communications (GCITC)*, Bangalore, India, Dec. 2023, pp. 1–6, https://doi.org/10.1109/GCITC60406.2023.10426312.

[23] H. M. Gomes, J. P. Barddal, F. Enembreck, and A. Bifet, "A Survey on Ensemble Learning for Data Stream Classification," *ACM Computing Surveys*, vol. 50, no. 2, pp. 1–36, Mar. 2018, https://doi.org/10.1145/3054925.