Archives available at journals.mriindia.com

International Journal on Advanced Computer Engineering and Communication Technology

ISSN: 2278-5140

Volume 14 Issue 03s, 2025

Feature Drift Aware Boosted Ensemble with Feature Set Management in Concept Drift Analysis for Academic Data

[1]Abhay A. Dande, [2]Dr. Mahendra A. Pund, [3]Ms. Amruta Deshmukh

[1,2] Computer Science and Engineering Department , PRMIT&R, Badnera, 444701, Amravati (M.H.), India

[3]Leader, IT Applications, Ciena Corporation

Email: [1]dandeabhay@gmail.com, [2]pundmukesh@gmail.com , [3]deshmukhamrutaphd@gmail.com

Peer Review Information

Submission: 05 Nov 2025

Revision: 25 Nov 2025

Acceptance: 17 Dec 2025

Keywords

Concept Drift, Feature drift, Drift detection and adaptation, Adaptive ensemble learning

Abstract

Predictive modelling in education has become a cornerstone of learning analytics, supporting student performance forecasting, early identification of at-risk learners, and adaptive curriculum design. However, these models often suffer degradation due to **concept drift**, the phenomenon where the joint distribution of features and outcomes evolves over time. In academic settings, drift arises from shifting curricula, new teaching modalities, grading policy changes, and evolving student behaviors, making drift adaptation a pressing research challenge. While much of the literature addresses global model retraining, recent advances highlight the importance of **feature drift**—distributional changes localized to specific input features or feature groups. Blind retraining can lead to unnecessary computational costs and "overfixing" of stable features, whereas feature-level management allows for more targeted, efficient adaptations. This paper reviews the past decade's progress in drift detection and adaptation, emphasizing feature-wise strategies in supervised learning models. We then propose a novel framework, **Feature Drift-Aware Boosted Ensemble for Education (FDABE-Edu)**, an ensemble-based architecture designed to detect, isolate, and adapt to drifting features in academic data streams. FDABE-Edu incorporates a continuous **Feature Drift Analyzer**, a **Weighted Ensemble** that prioritizes stable features while down-weighting volatile ones, and an **Update Manager** for iterative recalibration. The framework leverages statistical and performance feedback to dynamically adjust drift thresholds and retrain only where necessary. A pseudo-code implementation, system flowcharts, and architectural diagrams are provided to demonstrate feasibility. Comparative analysis situates FDABE-Edu alongside state-of-the-art drift handling algorithms, illustrating its novelty in academic contexts. Finally, we discuss emerging research avenues, including hierarchical drift detection, semi-supervised learning with delayed labels, fairness-aware adaptation, and causal inference integration. By bridging feature-level analysis with ensemble learning, FDABE-Edu advances the development of robust, adaptive, and transparent predictive systems for evolving educational environments.

Introduction

Educational institutions increasingly rely on predictive models to forecast student performance, identify at-risk learners, and optimize curriculum design. These models are typically trained on historical data under the assumption that the data distribution remains stationary. In practice, this assumption rarely holds: learning environments evolve continually as curricula are updated, grading policies shift, student demographics change, and external events such as sudden transitions to remote learning during a pandemic alter student behavior. Such changes give rise to concept drift [1, 2], the phenomenon wherein the joint distribution of features (X) and labels (Y) changes over time. If not addressed, concept drift can degrade a model's accuracy and lead to unfair or ineffective decisions, undermining the utility of learning analytics in education. Detecting and adapting to drift has therefore become a critical challenge in academic data mining.

Literature Review

Concept drift [1, 2] is formally defined as a change in the underlying data-generating process between different time periods. Let $P_t(X,Y)$ denote the joint distribution at time t. A concept drift occurs at a later time s if $P_t(X,Y) \neq P_s(X,Y)$. This change can manifest in several ways. In some cases, only the marginal distribution of features $P(X)$ changes while the conditional relationship $P(Y|X)$ remains stable. This phenomenon, often referred to as *covariate drift* or *virtual concept drift* [5, 2], reflects shifts in the feature space without altering the target concept. In educational settings, virtual drift may arise if the proportion of certain student demographics or the frequency of online course interactions changes over time, while their relationship to outcomes (e.g., grades) remains unchanged.

Alternatively, the relationship between features and labels may itself change. For instance, the adoption of a new teaching method could alter how study time translates into exam performance. This scenario constitutes *real drift* (also known as *concept shift*), where $P(Y|X)$ changes while $P(X)$ remains fixed, necessitating adjustments to the model's decision boundaries to preserve accuracy. A more general situation is *mixed drift*, in which both $P(X)$ and $P(Y|X)$ evolve simultaneously, a common occurrence in real-world data streams. Furthermore, entirely new classes or outcome categories may emerge over time, requiring the model to recognize previously unseen states. For example, the introduction of a pass/fail grading option represents a novel class that could disrupt a Grade Point Average (GPA) prediction model.

In educational data, feature-specific drifts are quite common: for example, learning management system (LMS) interaction frequencies may fluctuate significantly with new technology or academic calendars, even as other features like prior GPA or demographics remain relatively stable. If the overall model is retrained blindly on all features whenever performance drops, we risk "overfixing" parts of the model that were not actually affected by the drift. This motivates approaches that perform feature set management – dynamically selecting, weighting, or adapting features – to isolate the impact of drift and avoid unnecessary retraining.

This paper presents an in-depth review of concept drift [1, 2] analysis methods from the past ten years, with a special focus on strategies for feature set management in supervised learning models.FDABE-Edu aims to maintain predictive performance with minimal retraining, even as student data streams evolve. We provide conceptual architecture diagrams, flowcharts, and a comparative analysis of algorithms to support the discussion. Implementation-level guidance, including pseudo-code for the proposed approach, is also given to outline how such a system can be constructed. Finally, we discuss open research directions, including the challenges of label delays, fairness under drift, cross-institution model transfer, and combining drift adaptation with causal inference.

Concept Drift And Feature Drift In Academic Data

Concept drift [1, 2] research addresses the changing joint distribution $P(X,Y)$ over time, but it is often useful to break this phenomenon down into components affecting features and labels separately. As introduced above, virtual drift [5, 2] refers to changes in the input feature distribution $P(X)$ (also called covariate drift or data drift) that do not on their own alter the target concept. In academic data, an example of virtual drift would be a shift in the distribution of high school GPAs among incoming freshmen over years – this might change $P(X)$ (the GPA feature), but if the relationship between high school GPA and college performance (label) stays constant, the predictive model's decision boundary might not need to change. Real drift, on the other hand, is a change in the conditional distribution $P(Y|X)$, meaning the mapping from features to outcomes has changed even if the feature statistics look similar. An example here is if, after introducing open-book exams, the correlation between time spent in the library (a feature) and exam score (the label) diminishes; even with the same range of library hours (so

$P(X)$ unchanged), the predictive importance of that feature drops because $P(Y|X)$ shifted. Real drift directly impacts the model's correctness, as it signifies that the model's learned concept is no longer valid.

In practice, these two forms of drift often occur together. A curriculum change can introduce mixed drift: say a new online assignment platform leads students to spend less time in physical libraries ($P(X)$ shifts) and the value of library time for predicting grades changes because online resources dominate ($P(Y|X)$ shifts). Moreover, educational time series can exhibit recurrent or seasonal patterns. For instance, at the start of every semester, student engagement might spike and then normalize – a recurring drift [5] that could be mistaken for a permanent concept change if not recognized as periodic. Identifying whether an observed performance drop stems from a temporary change or a lasting shift is crucial for choosing the right adaptation strategy. Temporary drifts might warrant a light-touch approach (e.g. buffering with a short-term model) whereas permanent drifts require more extensive model updates. Feature drift [9] zooms in on the feature space: it occurs when specific features or groups of related features experience distributional shifts, even if the overall target concept is largely stable. In other words, one subset of X might undergo virtual drift [5, 2]. Feature drift is particularly pertinent in educational domains because different data sources evolve at different paces. For example, consider a student success prediction model that uses both demographic data and LMS interaction logs. Demographic features like age or high school background might remain fairly stable year over year aside from minor shifts in incoming cohorts, whereas LMS interaction features could change rapidly with the adoption of new e-learning tools or changing student study habits. If the LMS features drift significantly say students stop using forums and start relying on video lectures, altering the distribution of the "forum posts" feature, a model might start to mis-predict outcomes due to this shift. However, retraining the entire model from scratch might be overkill if other features e.g. demographics, prior grades are still predictive and stable. Instead, detecting which feature or feature subset drifted – and possibly adjusting only that part of the model – can be a more efficient solution. Recent studies on learning analytics have indeed observed that models can remain accurate if based on stable features while their performance deteriorates mainly due to a few drifting features. For instance, Kustitskaya et al. (2025) found that digital profile data of students relatively static background information were stable over time and models built on them showed minimal drift, whereas models relying on digital footprint data like dynamic LMS activities degraded significantly.

IV. Feature Drift Aware Boosted Ensemble for Education(FDABE-Edu)

To address concept drift [1, 2] in academic data with fine-grained feature control, we propose an architecture called Feature Drift [9]-Aware Boosted Ensemble for Education (FDABE-Edu). This approach adapts an existing idea of feature drift-aware ensembles – originally applied in e-commerce contexts – to the academic domain. The key components of FDABE-Edu are: a Feature Drift Analyzer that continuously monitors each feature for signs of drift, a Weighted Ensemble Model that makes predictions using multiple base learners and adjusts for feature stability, and an Update Manager that orchestrates periodic updates, retraining, and threshold adjustments.

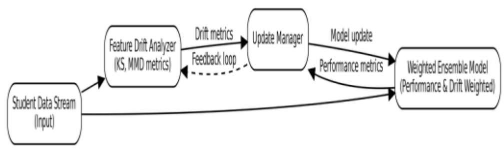

Figure 1: Proposed architecture of the FDABE-Edu system.

In this schematic, streaming student data (features and labels, when available) flow first into a Feature Drift [9] Analyzer. The analyser computes drift metrics for each incoming feature, for example by comparing the current distribution of the feature to its historical distribution using statistical tests or by measuring changes in feature importance. If a feature shows a significant drift beyond a predefined threshold, the analyser can flag it as "drifting."

The data with potentially some features marked or filtered out then feeds into the Weighted Ensemble Model. This ensemble consists of multiple base learners such as decision trees, gradient boosted models, or neural network predictors importantly, each learner can be specialized to a subset of features or the ensemble can adjust the contribution of each feature in its aggregate prediction. For instance, one base model might use all features but is trained with weights that emphasize stable features, whereas another base model might explicitly ignore a highly volatile feature to see if it improves robustness.

The ensemble combines the predictions of these base learners, weighting each learner's contribution by its recent performance models that have been accurate on recent data are given higher weight and by the stability of the features it relies on. Thus, a base learner that uses mostly stable features will tend to maintain good performance and therefore receive a higher vote in the ensemble's prediction.

The Update Manager module oversees the process over time: it periodically e.g., after each new batch or each semester recalibrates feature weights based on the latest drift analysis, decides if any base learner should be retrained or replaced for example, if one model has consistently underperformed due to drift in its features, the manager might replace it with a new model trained on the current data distribution, and adjusts the drift detection thresholds using feedback from the model's outcomes.

Notably, there is a feedback loop where the ensemble's performance information is fed back into the Feature Drift Analyzer – this can help adjust the sensitivity of drift detection. For example, if the ensemble handled a drift without much loss in accuracy, the analyser might raise thresholds to avoid flagging trivial drifts; if the ensemble struggled, the analyser might lower thresholds to become more sensitive in the future.

The FDABE-Edu architecture emphasizes continuous monitoring and adaptation. The Feature Drift [9] Analyzer and Weighted Ensemble work in tandem: the analyser tells the ensemble "which features to trust" at any given time, and the ensemble's success or failure informs the analyser whether its drift signals were valid.

This is particularly useful in education where ground truth labels like final course outcomes may only become available after a long delay , the ensemble's interim performance say on weekly quizzes or other proxy metrics can be used as feedback to validate or adjust drift detections.

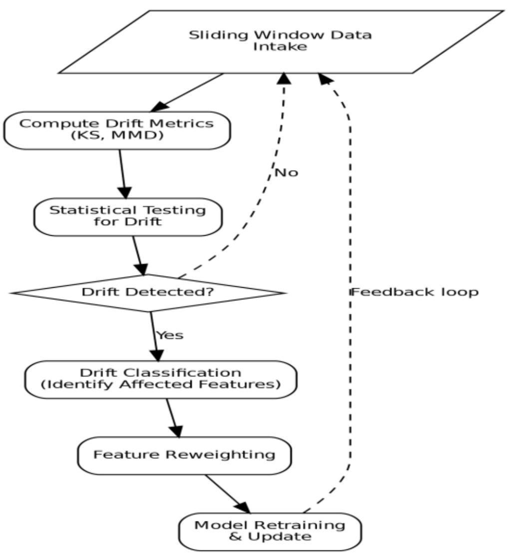

Figure 2: Flowchart of FDABE-Edu's feature drift detection and adaptation process.

The system operates in iterative cycles on streaming data. Initially, it starts by capturing an initial data window (this could be data from the first semester or the first N students, to set a baseline). The Feature Drift [9] Analyzer then computes drift metrics for each feature in a new incoming window of data and applies statistical tests to decide if any feature has drifted e.g., a null hypothesis "feature distribution has not changed" is tested for each feature. The process reaches a decision point: "Feature drift detected?" If the answer is Yes for one or more features, the system proceeds to update feature weights or remove drifting features. In practice, this means the ensemble will reduce the influence of those features on its predictions; the extreme case is dropping a feature entirely until it stabilizes. Next, if drift was detected (Yes branch), the system will retrain the ensemble on stable features – either retraining affected base learners or training new base learner(s) that do not use the drifting features – and adjust ensemble weights accordingly. If the drift decision was No, meaning all features appear stable, the system skips directly to the next step without retraining thus avoiding unnecessary computation when no drift is present. In either case, the data window is then slid forward – older data may be discarded or de-emphasized and new data included – representing the progression of time. The Update Manager then possibly adjusts drift thresholds or other parameters based on recent experience for example, if false alarms were triggered previously, it might raise thresholds; if a missed drift caused a performance drop, it might lower the thresholds to be more sensitive. Finally, the system moves to the next iteration with the updated model and parameters, ready to process the next incoming batch or time window of data.

Proposed Algorithm

Algorithm: FDABE-Edu Adaptive Learning Procedure

Input: Streaming labeled data (X_t, Y_t) arriving in time steps (or batches) $t = 1, 2, \dots$; initially trained base learners $h_1, h_2, \dots, h_M$ (ensemble) on first batch; initial feature drift thresholds $\Theta = \{\theta_i \text{ for each feature } X_i\}$.

Output: Continuously updated ensemble predictions $\hat{Y}_t$ for new data and an adaptive model.

1. Drift Analysis: For each feature X_i in the new data chunk, compare its distribution in recent data vs. past reference data (using statistical tests like KS, MMD [17], etc.) and compute a drift statistic D_i.

2. **Drift Detection:** If D_i exceeds its threshold θ_i (i.e., feature i shows

significant drift), mark feature i as drifting; otherwise, mark it as stable.

3. **Feature Weight Update:** For each base learner h_j in the ensemble, adjust its reliance on drifting features:
4. If h_j uses a drifting feature, reduce the weight of that feature in h_j (for example, modify the model's parameters or ignore that feature's input). Optionally, reduce the overall voting weight of h_j if it heavily relies on several drifting features.
5. **Drift Adaptation:** If any feature was marked drifting in step 2:
6. **Retraining:** Retrain the affected base learners h_j on the new data (excluding drifting features or with updated feature weights), or train a new base learner h_{new} using only stable features to replace a model that was degraded by drift.
7. **Ensemble Update:** Incorporate the retrained or new learners into the ensemble, replacing or augmenting the existing models. Re-compute the ensemble's combination weights (for example, assign weights to models inversely proportional to their recent error rates).
8. **Prediction:** Use the updated ensemble to make predictions $\hat{Y}_t$ for the current data (and optionally provide interpretability information such as which features were considered stable vs. drifting).
9. **Threshold Adjustment:** Using feedback (e.g., actual labels Y_t when available, or the ensemble's confidence), adjust the drift thresholds Θ:
10. If a false alarm was triggered (drift flagged but model accuracy did not suffer), increase the threshold θ_i for that feature (be less sensitive).
11. If a drift was missed (model performance dropped without any feature flagged), lower the relevant θ_i values (be more sensitive) for future detections.
12. **Slide Window:** Update the reference data to include the recent observations (and optionally discard the oldest data beyond a chosen window size). Proceed to the next time step ($t \leftarrow t+1$) and repeat from step 1.

This procedure ensures that FDABE-Edu continuously identifies drifts at the feature level, recovers model performance by focusing on stable features (retraining only what is necessary), and prevents degradation by preemptively down-weighting suspect features. It is a supervised framework, as it uses label feedback eventually for threshold tuning and model evaluation.

VI. FUTURE SCOPE

Our review underscores that significant progress has been made in detecting and adapting to concept drift [1, 2] but it also reveals gaps where further research—especially in the context of educational data—could be impactful. Below are several promising research directions building upon the idea of feature set management and beyond:

Dynamic Feature Grouping and Hierarchical Drift Detection: Thus far, we considered features individually, but features in education are often logically related for example, all features related to "online engagement" vs. those related to "academic history". A future extension is to perform drift detection at the group level. One could create a hierarchy of feature clusters e.g., demographic features, behavioral features, assessment features and apply tests like HSIC [18] or MMD [17] to group aggregates waterfutures.eu. If a whole group shows evidence of drift, it might indicate a systemic change e.g., a policy change affecting all behavioral metrics. Group-level drift detection can improve interpretability administrators can be told "student interaction patterns drifted" rather than "feature X drifted" and allow group-wise adaptation, where an entire module of the model related to that group is updated. Some initial ideas in this vein could involve hierarchical clustering of features with drift scores or using domain knowledge to pre-group features, then performing multivariate drift tests on those groups.

Label Scarcity and Proxy Supervision: In academic settings, labels like final grades or dropout outcomes are often delayed, arriving only after a semester or year. This leads to a dangerous blind spot where an early drift could go unnoticed until outcomes are realized. Semi-supervised approaches for drift could fill this gap. One idea is to incorporate proxy labels or interim assessments to continually update the model. For instance, weekly quiz performance might serve as a proxy for final exam performance; if the relationship between quiz scores and eventual success drifts perhaps quizzes become harder or easier, that can be detected earlier. FDABE-Edu could be extended to update feature weights using unsupervised metrics and then calibrate those updates once true labels arrive. Another approach is active learning: the system might selectively request human labels on a few new instances to check if a drift suspected from feature statistics is causing a real concept change. Research could explore how to optimally choose such instances e.g., focusing on borderline cases that would be most informative. Developing drift detection methods that remain reliable despite verification latency i.e. the delay in label availability is crucial for deployment in education, and techniques such as periodic backtesting i.e. checking model predictions once labels arrive and adjusting retrospectively are worth exploring.

Conclusion

Traditional approaches typically update or retrain models in entirety when drift is detected, but we highlighted literature that suggests more nuanced interventions: identifying which features have drifted can allow the model to adapt in a targeted way, conserving what still works and changing only what is necessary.

Feature set management in concept drift [1, 2], while not extensively studied in educational contexts before, shows great promise. By dynamically selecting and weighting features, models can isolate the impact of drifting signals, thereby maintaining accuracy longer and reducing retraining frequency. We proposed FDABE-Edu, a Feature Drift [9] Aware Boosted Ensemble tailored for academic data streams, which embodies these ideas. The architecture (Figure 1) and workflow (Figure 2) illustrate how incoming student data is continuously analysed for feature-level drifts, and how an ensemble model adjusts its focus toward stable features while down-weighting or retraining parts that rely on drifting ones. This approach can handle scenarios like a sudden shift to online learning or a slow change in study habits ,gradual drift in resource usage features in a principled manner.

In conclusion, concept drift [1, 2] remains a significant challenge for deploying machine learning in any dynamic domain, and education is no exception. However, the past decade's advancements provide a rich toolkit to detect, understand, and adapt to drift. By incorporating feature set management techniques essentially giving models a form of "peripheral vision" to notice when one input source changes , we can extend the longevity and fairness of predictive models amid evolving educational landscapes. Our proposed FDABE-Edu framework is a step in that direction, blending ideas from ensemble learning and feature monitoring to create a resilient model suited for academic analytics. The hope is that this review and the ideas within not only summarize what has been achieved in the literature but also inspire further research at the intersection of concept drift, feature engineering, and educational data mining. Robust, fair, and transparent adaptive learning systems will be key to ensuring that as education evolves through new technologies, policies, or global events, our analytical tools evolve with it, continuing to provide valuable and trustworthy insights.

References

J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," ACM Computing Surveys, vol. 46, no. 4, pp. 1–37, 2014.

J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 12, pp. 2346–2363, 2019.

G. Ditzler, M. Roveri, C. Alippi, and R. Polikar, "Learning in nonstationary environments: A survey," IEEE Computational Intelligence Magazine, vol. 10, no. 4, pp. 12–25, 2015.

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, 2018.

G. I. Webb, R. Hyde, H. Cao, H. L. Nguyen, and F. Petitjean, "Characterizing concept drift," Data Mining and Knowledge Discovery, vol. 30, no. 4, pp. 964–994, 2016.

A. S. Iwashita and J. P. Papa, "An overview on concept drift learning," IEEE Access, vol. 7, pp. 1532–1547, 2019.

R. N. Gemaque, J. P. Barddal, H. M. Gomes, F. Enembreck, and A. Bifet, "An overview of unsupervised drift detection methods," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 10, no. 6, e1381, 2020.

B. Halstead, Y.-S. Koh, P. Riddle, and T. Gedeon, "Recurring concept memory management in data streams: Exploiting data stream concept evolution to improve performance and transparency," Data Mining and Knowledge Discovery, vol. 35, no. 3, pp. 796–836, 2021.

J. P. Barddal, H. M. Gomes, F. Enembreck, and B. Pfahringer, "A survey on feature drift adaptation: Definition, benchmark, challenges and future directions," Journal of Systems and Software, vol. 127, pp. 278–294, 2017.

F. Hinder, V. Vaquet, and B. Hammer, "Feature-based analyses of concept drift," Neurocomputing, 2024 (in press).

A. J. Rabash, M. Z. A. Nazri, A. Shapii, and A. Al-Jumaily, "Stream learning under concept and feature drift: A literature survey," Journal of Artificial Intelligence, 2023.

D. Zhao and Y.-S. Koh, “Feature drift detection in evolving data streams,” in Proceedings of the 31st International Conference on Database and Expert Systems Applications (DEXA), Bratislava, Slovakia, 2020, pp. 335–349.

J. Gama, P. Medas, G. Castillo, and P. Rodrigues, “Learning with drift detection,” in Proceedings of the Brazilian Symposium on Artificial Intelligence (SBIA), São Luis, Brazil, 2004, pp. 286–295.

M. Baena-García, J. del Campo-Ávila, R. Fidalgo, A. Bifet, R. Gavaldà, and Ó. Alonso, “Early drift detection method,” in Proceedings of the 2nd International Workshop on Knowledge Discovery from Data Streams, 2006, pp. 77–86.

A. Bifet and R. Gavaldà, “Learning from time-changing data with adaptive windowing,” in Proceedings of the SIAM International Conference on Data Mining (SDM), Minneapolis, MN, 2007, pp. 443–448.

G. Hovakimyan and J. M. Bravo, “Evolving strategies in machine learning: A systematic review of concept drift detection,” Information, vol. 15, no. 12, Art. no. 786, 2024.

I. Goldenberg and G. I. Webb, “Survey of distance measures for quantifying concept drift and shift in numeric data,” Knowledge and Information Systems, vol. 60, pp. 591–615, 2019.

A. Gretton, K. Fukumizu, C.-H. Teo, L. Song, B. Schölkopf, and A. Smola, “A kernel statistical test of independence,” in Advances in Neural Information Processing Systems 20 (NIPS), Vancouver, Canada, 2008, pp. 585–592.

R. S. M. de Barros and S. G. T. de Carvalho Santos, “An overview and comprehensive comparison of ensembles for concept drift,” Information Fusion, vol. 52, pp. 213–244, 2019.

A. Bifet, G. Holmes, B. Pfahringer, and R. Gavaldà, “Leveraging bagging for evolving data streams,” in Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD), Barcelona, Spain, 2010, pp. 135–150.

H. M. Gomes, A. Bifet, J. Read, et al., “Adaptive random forests for evolving data stream classification,” Machine Learning, vol. 106, no. 9–10, pp. 1469–1495, 2017.

H. Yu, B. Hu, W. Zhang, D. Cao, and J. Cao, "Type-LDD: A Type-Driven Lite Concept Drift Detector for Data Streams," IEEE Transactions on Knowledge and Data Engineering, 2024.