On Handling Concept Drift, Calibration and Explainability in Non-Stationary Environments and Resources Limited Contexts

Sara Kebir[a] and Karim Tabia[b]
Univ. Artois, CNRS, CRIL F-62300 Lens, France

Keywords: Concept Drift, Lightweight Incremental Learning, Calibration, XAI, Feature Attribution.

Abstract: In many real-world applications, we face two important challenges: The shift in data distribution and the concept drift on the one hand, and on the other hand, the constraints of limited computational resources, particularly in the field of IoT and edge AI. Although both challenges have been well studied separately, it is rare to tackle these two challenges together. In this paper, we put ourselves in a context of limited resources and we address the problem of the concept and distribution shift not only to ensure a good level of accuracy over time, but also we study the impact that this could have on two complementary aspects which are the confidence/calibration of the model as well as the explainability of the predictions in this context. We first propose a global framework for this problem based on incremental learning, model calibration and lightweight explainability. In particular, we propose a solution to provide feature attributions in a context of limited resources. Finally, we empirically study the impact of incremental learning on model calibration and the quality of explanations.

1 INTRODUCTION

In some applications, data properties are not stationary (they may change over time) and shifts in the statistical properties of some classes may occur impacting negatively the performance of the used machine learning (ML) models. This is a well-known problem called concept drift (Lu et al., 2019) and its treatment consists in detecting such drifts then updating the used models with recent available data. In modern applications, it is no longer enough to have an accurate ML model, but to have complete confidence in these systems, it is also necessary to have well-calibrated models (providing good estimates of their predictive uncertainty) and explainable predictions. These problems are relatively well studied in the literature. However, this problem in a context of limited resources is very little explored. Indeed, if we consider the problem of anomaly detection in the case of a smart home where several sensors are used and where the detection is done in an egde AI fashion (locally and closer to the data collection sites), it is essential to take into account the changes and shifts that may occur over time (e.g. because people's habits change, the context too, etc.). It is also essential to provide a precise estimate of the confidence of the models and to have explanations when an alert is raised.

In an era where the Internet of Things (IoT) has rapidly permeated our lives, the promise of interconnected devices ushering in a new age of convenience and efficiency is undeniable. From smart homes that adjust lighting and temperature to intrusion detection systems keeping us safe, the applications of IoT streaming data are diverse. Nevertheless, the resource constraints in this context are a harsh reality. The IoT devices are designed to be power-efficient, often equipped with minimal processing power, memory, and energy resources (Cook et al., 2020). This inherent limitation forces us to consider innovative strategies for processing, analyzing, and acting upon the data they generate. One of the main challenges accompanying this context is the dynamic and evolving nature of the data analyzed in the ever-changing and non-stationary real environments. As the AI used systems rely on historical data and trained models, they struggle to maintain their accuracy and effectiveness when facing shifts in data distributions, new patterns, or changing user habits causing model degradation and detection failure.

In the realm of addressing this challenge of concept

[a] https://orcid.org/0000-0002-4471-9119
[b] https://orcid.org/0000-0002-8632-3980

Kebir, S. and Tabia, K.
On Handling Concept Drift, Calibration and Explainability in Non-Stationary Environments and Resources Limited Contexts.
DOI: 10.5220/0012382200003636
Paper published under CC license (CC BY-NC-ND 4.0)
In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 2*, pages 336-346
ISBN: 978-989-758-680-4; ISSN: 2184-433X
Proceedings Copyright © 2024 by SCITEPRESS – Science and Technology Publications, Lda.

drift, numerous models and techniques have emerged, showcasing the pressing demand for adaptive machine learning strategies. These methodologies span a spectrum of domains, including incremental learning algorithms like Hoeffding Trees (HTs) (Lu et al., 2019) and robust concept drift detection mechanisms such as ADWIN (Bifet, 2009). These models aim to accommodate the evolving nature of data distributions, allowing machine learning systems to maintain their predictive accuracy over time. Although efforts are numerous around this issue, there are few works that address the impact of concept drift and adaptive strategies on confidence, calibration and explainability in non-stationary contexts. In this work, we focus on problems where resources are limited and the context is non-stationary while trying to shed light on the trustworthiness of the evolutive AI systems. The main contributions of the paper are :

1. We first, propose a framework to treat the concept drift problem on stream data using a lightweight windowing ensemble model consuming less time and memory compared to the adaptive state-of-the-art methods;
2. We then, provide the first preliminary results on explainability in a lightweight context. The proposed scheme is designed to use very few resources and provides explanations as close to a standard explanation method, like SHAP, which are very demanding;
3. We finally, draw attention to the impact of concept drift and incremental learning on the calibration and the quality of explainabilty of the used model over time and open up the question to new perspectives.

2 RELATED WORKS

Before diving into the issue of handling concept drift in resource constrained environments, we present, through this section, the related works to the concept drift detection, adaptation and the potential impact on calibration and explainability of the used models over time.

2.1 Concept Drift Detection

Concept drift problems frequently arise in IoT data due to its non-stationary nature and the dynamic environments in which IoT systems operate. This can lead to the deterioration of the performance of ML models. Detecting concept drift in IoT data presents two primary challenges: the presence of numerous potential causing factors and multiple types of drifts (Agrahari and Singh, 2022; Lu et al., 2019). The most common type of drifts is sudden drift, where the data distribution changes abruptly, often due to external factors like a shift in user behavior or a change in the environment. Gradual drift, on the other hand, involves a more gradual and consistent change where a new concept gradually replaces an old one over a period of time, making it challenging to detect. Incremental drift occurs when the drift happens in small, incremental steps. Finally, recurring drift involves periodic changes, often influenced by seasonal or cyclical patterns in the data. Detecting and adapting to these various types of concept drift is essential for maintaining the accuracy and reliability of machine learning models.

To effectively address the issue of concept drift, various detection techniques can be employed. Among the commonly used approaches for this purpose are distribution-based and performance-based methods (Yang and Shami, 2023).

Distribution-based methods rely on the use of data buffers, which can either be fixed-sized sliding windows or adaptive windows, to monitor different concepts. These methods are specifically designed to detect concept drift by observing changes within these windows. ADWIN, a well-known distribution-based approach, utilizes an adaptable sliding window to identify concept drift (Bifet, 2009). It does so by comparing key characteristic values of old and new data distributions, like mean and variance values. A significant alteration in these characteristic values over time serves as an indicator of a drift occurrence. ADWIN is particularly effective at dealing with gradual drifts and long-term changes. However, it can sometimes generate false alarms and unnecessary model updates.

Performance-based methods, on the other hand, assess model performance over time to recognize concept drift. These methods gauge the rate of degradation in model performance. Early Drift Detection Method (EDDM) (Baena-García et al., 2006), a widely-used performance-based technique, tracks changes in model performance based on the rate of change in a learning model's error rate and standard deviation. By employing drift and warning thresholds, EDDM can effectively identify instances of model performance degradation and the occurrence of concept drift, particularly sudden drift. However, it may not be as proficient as distribution-based methods in detecting gradual drift. Performance-based methods can effectively detect the drifts that cause model degradation, but they require the availability of ground-truth labels (Yang and Shami, 2021).

2.2 Concept Drift Adaptation

Once concept drift is detected, it is crucial to effectively address it to enable the learning model to adapt to the new data patterns. Several techniques can be used to handle concept drift.

Incremental learning methods involve the partial updating of the learning model whenever new samples arrive or concept drift is identified. This allows the model to adapt incrementally to changing data. Hoeffding Trees (HTs) (Lu et al., 2019) represents a fundamental incremental learning technique that utilizes the Hoeffding inequality to calculate the minimum required number of data samples for each split node. This allows it to update nodes and adapt to new data samples. Extremely Fast Decision Tree (EFDT) (Manapragada et al., 2018) is a cutting-edge incremental learning approach and an enhanced version of HTs. It promptly selects and deploys node splits as soon as they attain a confidence value indicating their utility. EFDT excels at adapting to concept drift more accurately and efficiently compared to HTs. Online Passive-Aggressive (OPA) (Crammer et al., 2006) is another incremental learning algorithm that treats drifts by passively reacting to correct predictions and aggressively responding to errors. Numerous incremental approaches have been developed by leveraging conventional machine learning algorithms. For instance, K-Nearest Neighbors with the ADWIN drift detector (KNN-ADWIN) (Losing et al., 2016) represent an enhanced iteration of the traditional KNN model designed for real-time data analysis. KNN-ADWIN, incorporates an ADWIN drift detector into the conventional KNN model and employs a dynamic window to determine which samples should be retained for model updates.

Ensemble online learning models represent advanced methods for adapting to concept drift, integrating the predictions of multiple base learners to enhance performance. Leverage Bagging (LB) (Bifet et al., 2010), is a fundamental ensemble technique that creates and combines multiple base learners, such as Hoeffding Trees (HTs), using bootstrap samples and a majority voting strategy. Adaptive Random Forest (ARF) (Gomes et al., 2017) and Streaming Random Patches (SRP) (Gomes et al., 2019) are two sophisticated ensemble online learning approaches that utilize multiple HTs as base models and incorporate a drift detector, such as ADWIN, for each HT to handle concept drift. ARF employs the local subspace randomization strategy to construct trees, while SRP utilizes global subspace randomization to generate random feature subsets for model learning. The use of global subspace randomization enhances SRP's learning performance but comes at the cost of increased model complexity and longer learning times. Multi-Stage Automated Online Network Data Stream Analytics (MSANA) is a framework that has been proposed by (Yang and Shami, 2023) for IoT Systems where they use a window-based strategy and select lightweight base learners with greater computational speeds to build the ensemble model. Their framework consists of dynamic data pre-processing, drift-based dynamic feature selection, dynamic model selection, and online model ensemble using a novel W-PWPAE approach (Yang et al., 2021). While ensemble online learning techniques generally surpass incremental learning methods in the realm of dynamic data stream analysis, they often come with a significant computational cost (Yang and Shami, 2023).

2.3 Classifier Calibration and Explainability

In this context of adaptive learning, where models are continuously updated with new data, concept drift can have a profound impact on the model's predictive performance. Some of the crucial aspects that may also be affected by concept drift are the confidence, quality of calibration and explainability of the used models.

Calibration measures how well a model's predicted probabilities align with the actual likelihood of an event occurring (Vaicenavicius et al., 2019). When concept drift occurs and the model is not regularly updated to account for it, the calibration may deteriorate over time, leading to unreliable and misleading probability estimates. To maintain the quality of calibration in incremental learning, models must be regularly monitored and adapted to evolving data distributions, ensuring that predictions remain trustworthy and valuable for decision-making. As with the measurement of classifier efficiency, there are various metrics for measuring calibration. Some commonly used measures are Expected Calibration Error (ECE), Average Calibration Error (ACE) and Maximum Calibration Error (MCE). Miscalibration measures assess errors by classifying samples according to their confidence level, and then evaluating the accuracy within each class. For example, MCE is simply the weighted average difference between the classifier's confidence and the observed accuracy (on a test set) in each bin. Similarly, the maximum calibration error (MCE) simply gives the maximum deviation (Naeini et al., 2015). Negative log likelihood (NLL) can also be used to indirectly measure model calibration since it penalizes high probability scores assigned to incorrect labels and low probability scores assigned to correct labels (Ashukha et al., 2020). Calibration quality improves as these metrics decrease.

Explainability in machine learning is crucial for understanding model decisions. XAI post-hoc explanation methods like SHAP (Lundberg and Lee, 2017) and LIME (Ribeiro et al., 2016) are well-known feature attribution methods. Although proficient at offering insights into the used models' predictions, they are often resource-intensive, especially when dealing with complex models or high-dimensional datasets (Van den Broeck et al., 2021). This computational intensity can pose considerable limitations in applications where real-time decision-making is crucial, such as the detection of emergency situations affecting the elderly in smart homes. In addition, their high computational demands can lead to increased energy consumption and costs, which may prove unsustainable in resource-constrained environments. There is therefore a compelling need for lightweight XAI systems that strike a balance between explainability and real-time feasibility.

3 PROPOSED FRAMEWORK

In response to the possible fluctuations in data distribution over time driven by concept drift, it becomes crucial to adopt an incremental learning strategy. This involves continually training and updating the ML model as new data becomes available. The process of labeling data in such cases can be quite demanding and resource-intensive, often requiring the involvement of domain experts in real-world applications. Our framework, illustrated in Figure 1 and described in this section, summarizes the different steps of the process going from an initial offline training of the used model to its deployment in a resource-constrained environment and update over time when receiving the stream data. This will be followed by the proposed lightweight explanation approach.

3.1 Global Overview of the Solution

As illustrated on Figure 1, the proposed framework is composed of three basic building blocks each ensuring an important functionality, namely, i) incremental learning to adapt to the concept drift, ii) model calibration to provide a good estimate of the model's confidence, and finally, ii) the explanation of the model predictions (to verify and trust the predictions). Of course, this solution is specially designed for constrained environments and limited in computational resources.

When deployed in a resource-constrained environment and when new data becomes available, this system makes predictions by combining the votes from the base classifiers within the ensemble model to determine both the target value and the level of confidence. After this prediction phase, an explanation is generated. This explanation, when coupled with the predicted class label and the machine learning model's confidence level, enhances trust in the prediction and aids the expert in assigning the most precise label. We will focus in the following on each building bloc.

3.2 Model Offline / Adaptive Learning

To meet the constraints of limited resources, we chose an incremental learning scheme based on windowing ensemble models composed of a few lightweight basic classifiers, as they offer a good compromise between adaptation to the concept drift and low resource consumption.

First, we initiate the training of the windowing ensemble model during an offline phase, utilizing a training dataset ensuring that all its base classifiers are initially trained on the same data at this stage.

Since the combination of initial training data and the incoming data stream is effectively endless, making online approaches inefficient in terms of time and computational resources, we opt for the use of a window to store the most recent incoming data until the next training iteration. The window size can be dynamic or fixed in advance depending on the nature of the data. Pre-selecting the data to be saved in this window can also be adopted to limit recurring occurrences that would not have much impact on retraining. Finally, the incremental and adaptive re-training is exclusively applied on the least efficient base classifier during each iteration to effectively handle potential concept drift in a less resource-intensive way, all while preserving the accumulated knowledge to maintain the continuous performance of the ensemble model. To further meet the challenge of limited resources, this incremental re-training can be triggered in two ways. The first is by setting up a concept drift detector based on either the data contained in the window, or the model's performance, or a combination of both. The second is to perform this re-training periodically when the length of the window storing new data reaches its threshold.

3.3 Lightweight Prediction Explanation

To ensure more efficient and lightweight XAI systems, we propose a shift from traditional, resource-

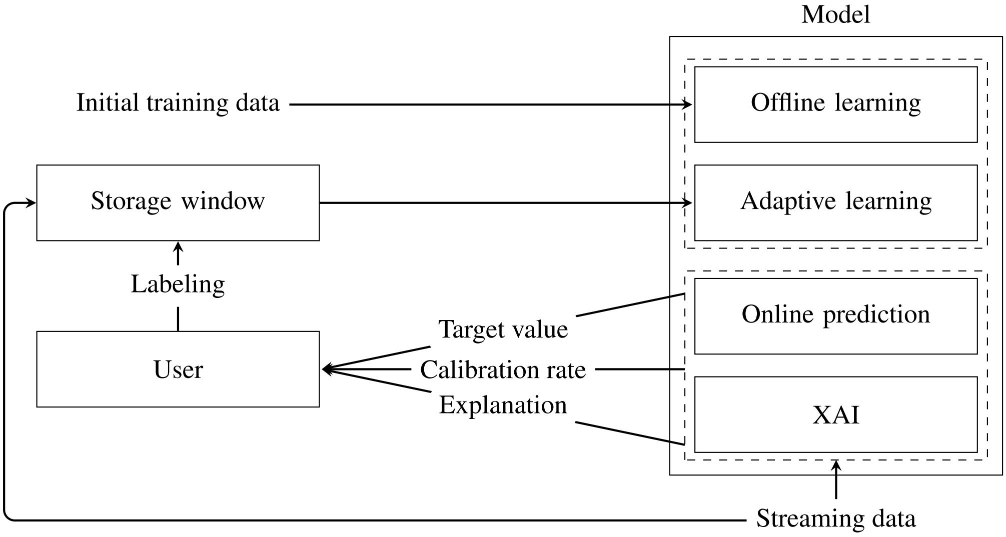

Figure 1: Concept drift adaptation proposed framework.

intensive XAI models to the use of lightweight[1] ML models, such as linear regressors and decision trees for example. Our fundamental concept centers around reducing the computational overhead linked to generating explanations by replacing them with regressions using ML predictive models which are much lighter in terms of prediction time and memory consumption. Fig. 2 illustrates how we generate lightweight explanations.

Instead of using an E explanation as a resource -overly greedy shap, to explain a predicition y for a data sample x, we propose to replace (more precisely to approximate) the explainer E by a set of regression models f_{x_1} ... f_{x_n} where each model f_{x_i} tries to predict $E(x_i)$ the feature attribution computed by E for the attribute x_i. The advantage of doing so is to learn offline models (in an environment that is resource-constrained) and then deploy these models in the constrained environment. These models which are supposed to approximate the Explainer E are of light size and above ensure feature attribution with a minimum of resources. To set up this solution, we have to

- **Build an explanation dataset:** to be able to approximate an explaner E which provides for a data sample $x = (x_1, .., x_n)$ and a prediction y made by the model to be explained, a vector of feature attributions $(f_{x_1}, .., f_{x_n})$ of scores where each score f_{x_i} tells how much the feature X_i was influential in the prediction y for x. Thus, to learn models that approximate E, we build a dataset composed of data samples x, their predictions by the model to be explained y as well as the feature attribution vectors computed by the explainer E as illustrated on the Table 1. This dataset can be easily built by taking up the dataset D which is used to train the model to be explained, the model's predictions on D and the explanations provided by the Explainer chosen E for data sample in D.
- **Build regressors to provide feature attributions:** Once explanation dataset has been built, we can train regression models (or only one model with several outputs in case of neural network-based regression) . If we build a model by feature f_{x_i}, then we have to train the regression model on the data play D and the corresponding explanation column f_{x_i} only.

4 EXPERIMENTAL STUDY

4.1 Experimental Settings

To assess the effectiveness of the proposed framework, four public IoT datasets related to anomaly detection and human activity recognition in a smart home environment facing concept drifts, are used: NSL-KDD, CICIDS2017, IoTID20 and Aruba.

- NSL-KDD is a balanced benchmark dataset for concept drift and network intrusion detection (Liu et al., 2020). For this dataset, it is known that there is a sudden drift when transitioning from the training set to the test set (Yang and Shami, 2021).
- CICIDS2017 is a dataset provided by the Canadian Institute of Cybersecurity (CIC), involving

[1] By lightweight ML models we mean models with low complexity (determining model size) and prediction time.

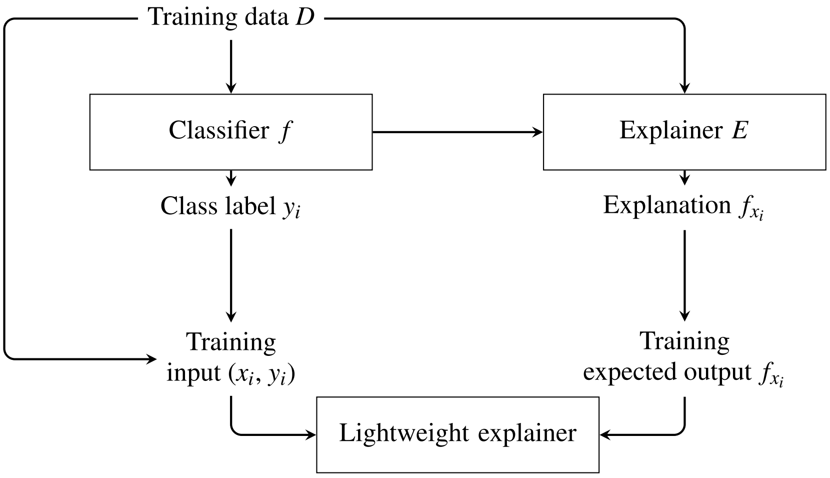

Figure 2: Lightweight explanation proposed approach.

Table 1: Illustration of the explanations dataset.

Input features X						Prediction		Feature attributions by an explainer E					
X_1	X_2	...	...	...	X_n	Y		f_{X_1}	f_{X_2}	...	...	...	f_{X_n}
12	”B”	...	...	...	”SF”	**1**		.34	.001	...	...	...	.12
8	”C”	...	...	...	”UG”	**0**		.05	.21	...	...	...	.02
55	”A”	...	...	...	”PS”	0		.07	.31	...	...	...	0
...	...	...	...	...	...	**...**		...	...	...	...	...	...
...	...	...	...	...	...	**...**		...	...	...	...	...	...
...	...	...	...	...	...	**...**		...	...	...	...	...	...
100	”B”	...	...	...	”SF”	**1**		.02	0	...	...	...	.05

cyber-attack scenarios. As different types of attacks were launched in different time periods to create this dataset, the attack patterns in the dataset change over time, causing multiple concept drifts (Sharafaldin et al., 2018).

- IoTID20 is an IoT traffic dataset with unbalanced data samples (94% normal samples versus 6% abnormal samples) for abnormal IoT device detection (Ullah and Mahmoud, 2020).
- Aruba is a dataset collected within the CASAS smart home project (Cook, 2012). This dataset collected different data sources in the home of an adult volunteer. The resident of the house was a woman who received visits from her children and grandchildren regularly between 4 November 2010 and 11 June 2011. Two data sources gave rise to the information, the first source was binary and was made up of movement and contact sensors, and the second source was made up of temperature sensors.

For the purpose of this work, a representative IoTID20 subset with 6,252 records and a sampled CICIDS2017 subset with 14,155 records, as well as a reduced Aruba dataset that has 200,784 records combining the firsts and lasts months of the experiment are used for the model evaluation.

These datasets allow us to see how our framework performs in binary classification for anomaly detection and in multi-class classification for activity recognition. They also allow us to observe how it performs in dealing with class imbalance, which is quite prevalent in the IoTID20 and CICIDS2017 datasets. Two validation methods, namely hold-out and prequential, are utilized for evaluation. In the hold-out evaluation, the initial 20% of the data is utilized for the initial model training, while the remaining 80% is reserved for online testing. In prequential validation, also known as test-and-train validation, each input sample from the online test set serves a dual purpose: first, it tests the learning model, and then it contributes to model training and updating (Yang et al., 2021).

The windowing ensemble model tested is composed of three decision trees as base classifiers. The model is updated in an incremental way by retraining the least efficient base classifier using windows of length 2000, 50, 500 and 10000 records for the NSL-KDD, CICIDS2017, IoTID20 and Aruba datasets, respectively.

The evaluation of the proposed framework's performance relies on different metrics linked to prediction quality, model calibration and lightweight in terms of total time and memory used during learning (accuracy, F1-score, NLL, ECE, MCE, total training time as well as occupied and peak memory).

4.2 Lightweight Adaptive Learning

Tables 2, 3, 4, 5 show the performance comparison of the proposed framework with other state-of-the-art drift adaptive approaches, including ARF (Gomes et al., 2017), SRP (Gomes et al., 2019), EFDT (Manapragada et al., 2018), KNN-ADWIN (Losing et al., 2016), OPA (Crammer et al., 2006), LB (Bifet et al., 2010) and MSANA (Yang and Shami, 2023). We can see that the ensemble model used without re-training (Baseline) was impacted by the various concept drifts contained in the four datasets tested, which explains its poorer performance compared to the online models. However, after the adaptive training phase (L-Ens), the model's performance improved on all tested datasets. The obtained results are better than those achieved with state-of-the-art methods on NSL-KDD and IoTID20 and almost similar on CICIDS2017 and Aruba. Figures 3 and 4 illustrate the continuous variation in accuracy of all the methods tested, and we can visually confirm that our approach adapts very well after the onset of concept drifts, which is not the case with the baseline model used without retraining. Furthermore, according to the calibration measurements (NLL, ECE, MCE) in Tables 2, 3, 4, 5, we can say that with the online training performed by the state-of-the-art methods, the calibration deteriorated over time, which is not the case with our ensemble method even after periodic re-training. Regarding the lightweight nature of the proposed framework, our approach required much less training time and used less memory than all the other state-of-the-art tested methods. This proves its effectiveness and good adaptation to concept drift, while answering the lightweight challenge.

4.3 Lightweight Explanation

Our proposal, although preliminary, has been tested with the four datasets presented above, by first, using as main classifier f a LightGBM (Jin et al., 2020), which is a fast and powerful ML model based on the ensemble of several decision trees, then, SHAP tree-Explainer (Lundberg et al., 2020) as the basic explanation model, and finally, a decision tree model as the classifier $f^{'}$ generating the lightweight explanations.

Table 6 shows the results of our preliminary approach (L-exp) compared with those obtained using SHAP. The evaluation measures used are related, on the one hand, to the quality of the explanations generated, and on the other, to the time and memory occupied when predicting a series of 1000 instances. The quality of the explanations was analyzed on different levels. The first is the reconstruction error mse, describing the gap between the explanations generated by our approach (through regression) and the true explanations (set of test instances generated by SHAP). The second level is linked to the predicted features. For a given explanation, we check on the first 2, 5 and 10 features, the rate of those that are common with the SHAP explanation, among which we note the rate of series where the features are identical, as well as the rate linked to the order of their appearance. From the results obtained, whether on the first 2, 5 or 10 features among the total of 41, 77, 31 and 38 features of the tested datasets, respectively NSL-KDD, CICIDS2017, IoTID20 and Aruba, we note that the SHAP explanations are almost similar to those generated with our approach, while meeting the lightweight criterion. Indeed, our approach (L-exp) takes much less time and memory than SHAP, especially on multi-class sets where XAI consumes much more memory than our framework.

4.4 Impact of Incremental Learning on Model Calibration and Explainability

It is also important to highlight the impact of concept drift on the explanations generated, as illustrated in Figures 5 and 6 for the NSL-KDD and CICIDS2017 datasets respectively. Indeed, we can see at the concept drift detection points in the dataset an increase in the mse error related to the quality of the generated explanations. Furthermore, based on the calibration measurements (NLL, ECE, MCE) in Tables 2, 3, 4 and 5, it can be observed that the online training conducted by state-of-the-art methods led to a degradation in their calibration over time. Consequently, the issue of adaptation not only in relation to the performance of the classifiers, but also to that of the calibration and quality of the explanation of the techniques currently in use over time, is put into perspective.

5 DISCUSSION AND CONCLUDING REMARKS

Through this work, we have highlighted the importance of taking into account the occurrence of concept drift and its impact on the ML models perfor-

Table 2: Performance comparison of our approach and state-of-the-art methods on NSL-KDD.

Model	Acc%	F1%	NLL	ECE%	MCE%	Training time (s)	Occupied mem (Kb)	Peak mem (Kb)
ARF-ADWIN	97.45	97.36	0.08	1.92	21.95	303.51	4977.42	8641.43
ARF-EDDM	98.05	97.99	0.07	1.14	18.34	465.56	10462.17	23007.39
SRP	97.93	97.86	0.08	1.41	22.62	1869.27	11779.89	19309.99
EFDT	90.50	89.96	1.13	3.27	12.27	6097.24	3399.01	4454.46
KNN-ADWIN	91.93	91.43	0.48	5.09	23.62	45.55	137.82	314.48
OPA	90.47	90.17	0.24	0.52	2.56	19.24	91.87	145.67
LB	97.53	97.44	0.09	0.57	12.70	1400.55	24131.05	24167.58
MSANA	92.90	92.49	0.17	2.13	6.34	2313.64	22522.52	74407.18
Baseline	95.99	95.75	1.43	0.01	3.98	0.78	57.88	6685.43
L-Ens	**98.17**	**98.10**	**0.28**	**0.02**	**15.08**	**2.04**	**60.52**	**6708.88**

Table 3: Performance comparison of our approach and state-of-the-art methods on CICIDS2017.

Model	Acc%	F1%	NLL	ECE%	MCE%	Training time (s)	Occupied mem (Kb)	Peak mem (Kb)
ARF-ADWIN	98.55	93.58	0.08	0.59	18.71	33.46	943.91	1869.14
ARF-EDDM	98.73	94.34	0.08	0.74	15.12	33.08	596.64	971.64
SRP	98.98	95.53	0.10	0.58	13.80	272.39	2125.76	7027.66
EFDT	95.01	80.33	1.13	0.99	11.72	332.31	1244.13	1532.5
KNN-ADWIN	98.77	94.72	0.16	0.68	9.30	12.95	138.86	383.19
OPA	98.26	92.43	0.11	6.05	16.81	4.92	43.70	147.05
LB	98.13	91.86	0.14	0.45	15.40	397.88	2460.46	5667.54
MSANA	98.96	95.40	0.07	2.87	15.86	562.71	2519.62	12394.82
Baseline	86.58	0.13	4.84	0.00	13.42	0.11	12.60	1052.09
L-Ens	**97.27**	**87.74**	**0.38**	**0.00**	**1.37**	**2.43**	**77.38**	**1067.70**

Table 4: Performance comparison of our approach and state-of-the-art methods on IoTID20.

Model	Acc%	F1%	NLL	ECE%	MCE%	Training time (s)	Occupied mem (Kb)	Peak mem (Kb)
ARF-ADWIN	98.26	99.08	0.09	1.44	21.17	13.78	1053.30	1369.63
ARF-EDDM	98.00	98.95	0.10	1.65	26.32	14.86	1691.27	1702.45
SRP	98.72	99.32	0.10	1.37	19.15	64.27	3002.57	3075.86
EFDT	96.02	97.92	0.40	0.94	1.39	102.42	566.49	772.45
KNN-ADWIN	95.92	97.85	0.30	3.45	25.37	2.88	83.90	243.33
OPA	93.74	96.69	0.19	4.32	11.40	1.00	75.96	115.20
LB	98.06	98.98	0.09	0.74	9.21	95.72	2482.13	2553.32
MSANA	98.58	99.25	0.06	2.12	10.49	323.75	5008.29	6888.13
Baseline	99.26	99.61	0.27	0.00	0.74	0.05	9.14	247.97
L-Ens	**99.26**	**99.61**	**0.11**	**0.00**	**0.19**	**0.20**	**38.50**	**249.79**

mance. To address the challenges posed by this concept drift, we explored the benefits of adaptive learning in resource-constrained environments, while assessing its impact on model performance and calibration, as well as the quality of explanations provided over time. Our proposed approach is based on the use of a lightweight windowing ensemble model that is incrementally updated. It also includes preliminary work related to the generation of lightweight explanations. The results obtained on binary and multi-class datasets demonstrate its effectiveness over time, while maintaining very low resource costs. These results also raise questions about the quality of calibration after this concept drift adaptation stage, and how to generate high-quality, adaptive explanations over time. As a future direction, the exploration of uncertainty and reliability in incremental AI facing concept drift involves enhancing calibration and the quality of explainability approaches while considering various classifiers and XAI methods. This exploration

Table 5: Performance comparison of our approach and state-of-the-art methods on Aruba.

Model	Acc%	F1%	NLL	ECE%	MCE%	Training time (s)	Occupied mem (Kb)	Peak mem (Kb)
ARF-ADWIN	96.10	82.47	0.22	2.45	24.48	305.86	453.07	1687.51
ARF-EDDM	96.51	83.82	0.19	3.25	24.22	316.47	831.43	1171.09
SRP	97.51	89.77	0.16	2.13	25.30	2057.65	3583.76	9093.03
EFDT	94.33	73.17	0.78	2.93	15.56	10526.64	6333.47	6501.39
KNN-ADWIN	98.06	93.77	0.15	1.64	58.47	85.15	93.86	268.95
OPA	2.12	4.01	35.26	97.26	98.59	31.52	72.71	134.77
LB	96.24	87.69	0.34	1.45	19.95	1791.96	1184.22	2547.22
Baseline	80.35	50.81	7.08	0.00	19.65	1.92	37.36	15241.17
L-Ens	**89.77**	**60.00**	**2.27**	**0.00**	**5.69**	**3.61**	**52.81**	**15242.80**

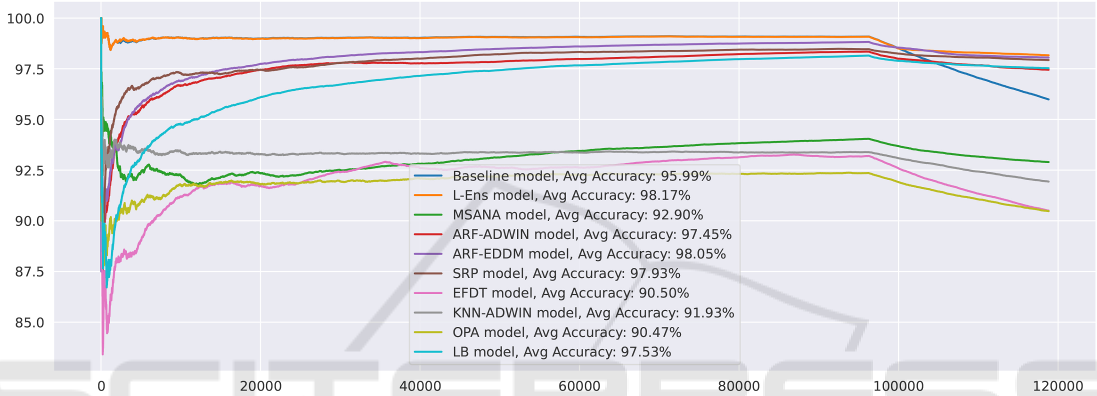

Figure 3: Accuracy variation on NSL-KDD.

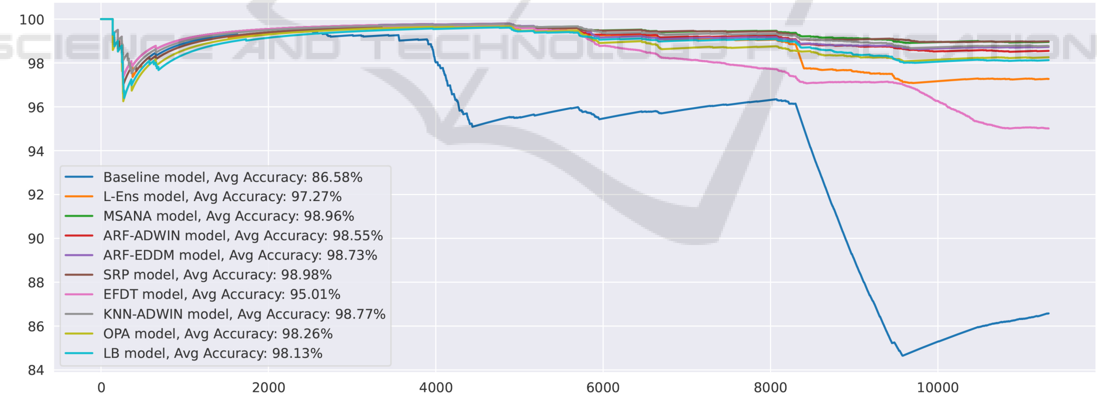

Figure 4: Accuracy variation on CICIDS2017.

Table 6: Evaluation of the quality of generated lightweight explanations.

Dataset	Explanation quality										Time (s)		Memory (Kb)	
	Mse	Shared features (%)			Same set (%)			Same order (%)			SHAP	L-exp	SHAP	L-exp
		2	5	10	2	5	10	2	5	10				
NSL-KDD	0.0004	95.59	96.34	97.31	91.84	85.57	81.35	97.51	92.33	91.74	0.21	0.002	339.71	322.41
IoTID20	0.04	99.04	99.33	97.60	98.16	96.90	83.01	99.04	96.93	94.51	5.62	0.002	276.46	243.85
CICIDS2017	9.55e-05	97.59	98.38	98.59	96.23	93.06	87.90	99.97	93.27	95.25	0.086	0.002	619.82	603.19
Aruba	84.19	93.68	93.67	95.05	90.37	80.30	75.37	98.27	92.63	87.56	8.06	0.002	3406.64	299.29

should also encompass the assessment of the influence of data pre-processing and balancing on a continual basis.

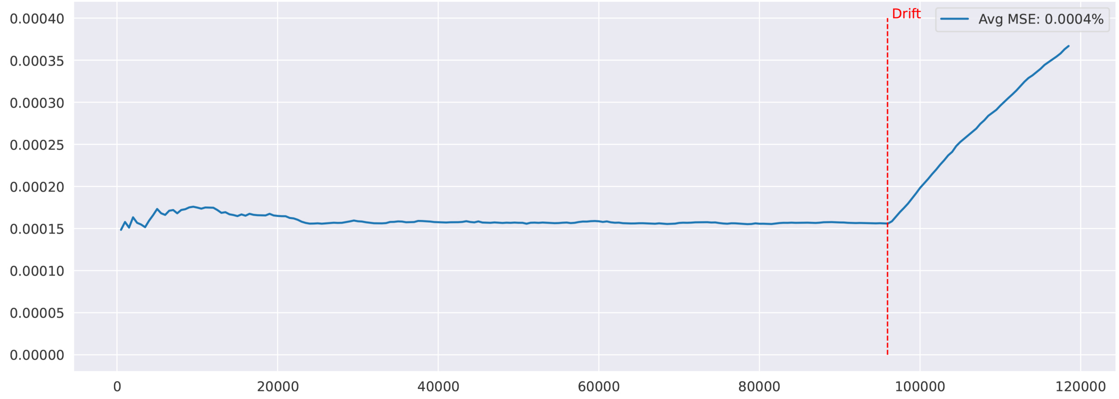

Figure 5: Impact of concept drift on the mse error associated with NSL-KDD explanations.

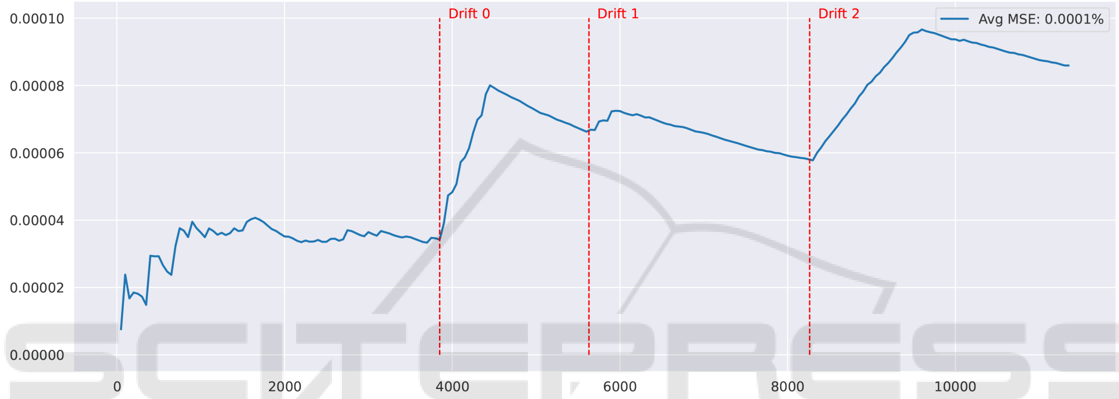

Figure 6: Impact of concept drift on the mse error associated with CICIDS2017 explanations.

ACKNOWLEDGEMENTS

This work has been supported by the Vivah project 'Vers une Intelligence artificielle à VisAge Humain' supported by the ANR.

REFERENCES

Agrahari, S. and Singh, A. K. (2022). Concept drift detection in data stream mining : A literature review. *Journal of King Saud University - Computer and Information Sciences*, 34(10, Part B):9523–9540.

Ashukha, A., Lyzhov, A., Molchanov, D., and Vetrov, D. P. (2020). Pitfalls of in-domain uncertainty estimation and ensembling in deep learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Baena-García, M., Campo-Ávila, J., Fidalgo-Merino, R., Bifet, A., Gavald, R., and Morales-Bueno, R. (2006). Early drift detection method.

Bifet, A. (2009). Adaptive learning and mining for data streams and frequent patterns. *SIGKDD Explor.*, 11:55–56.

Bifet, A., Holmes, G., and Pfahringer, B. (2010). Leveraging bagging for evolving data streams. In Balcázar, J. L., Bonchi, F., Gionis, A., and Sebag, M., editors, *Machine Learning and Knowledge Discovery in Databases*, pages 135–150, Berlin, Heidelberg. Springer Berlin Heidelberg.

Cook, A. A., Mısırlı, G., and Fan, Z. (2020). Anomaly detection for iot time-series data: A survey. *IEEE Internet of Things Journal*, 7(7):6481–6494.

Cook, D. (2012). Learning setting-generalized activity models for smart spaces. *IEEE Intelligent Systems*, 27(1):32–38.

Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., and Singer, Y. (2006). Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585.

Gomes, H., Read, J., and Bifet, A. (2019). Streaming random patches for evolving data stream classification. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 240–249, Los Alamitos, CA, USA. IEEE Computer Society.

Gomes, H. M., Bifet, A., Read, J., Barddal, J. P., Enembreck, F., Pfharinger, B., Holmes, G., and Abdessalem, T. (2017). Adaptive random forests for evolving data stream classification. *Machine Learning*, 106(9):1469–1495.

Jin, D., Lu, Y., Qin, J., Cheng, Z., and Mao, Z. (2020). Swiftids: Real-time intrusion detection system based on lightgbm and parallel intrusion detection mechanism. *Computers & Security*, 97:101984.

Liu, J., Kantarci, B., and Adams, C. (2020). Machine learning-driven intrusion detection for contiking-based iot networks exposed to nsl-kdd dataset. In *Proceedings of the 2nd ACM Workshop on Wireless Security and Machine Learning*, WiseML '20, page 25–30, New York, NY, USA. Association for Computing Machinery.

Losing, V., Hammer, B., and Wersing, H. (2016). Knn classifier with self adjusting memory for heterogeneous concept drift. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 291–300.

Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., and Zhang, G. (2019). Learning under concept drift: A review. *IEEE Transactions on Knowledge and Data Engineering*, 31(12):2346–2363.

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., and Lee, S.-I. (2020). From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, 2(1):56–67.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA. Curran Associates Inc.

Manapragada, C., Webb, G. I., and Salehi, M. (2018). Extremely fast decision tree. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, page 1953–1962, New York, NY, USA. Association for Computing Machinery.

Naeini, M. P., Cooper, G. F., and Hauskrecht, M. (2015). Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 2901–2907. AAAI Press.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 1135–1144, New York, NY, USA. Association for Computing Machinery.

Sharafaldin, I., Lashkari, A. H., and Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *International Conference on Information Systems Security and Privacy*.

Ullah, I. and Mahmoud, Q. H. (2020). A scheme for generating a dataset for anomalous activity detection in iot networks. In *Advances in Artificial Intelligence: 33rd Canadian Conference on Artificial Intelligence, Canadian AI 2020, Ottawa, ON, Canada, May 13–15, 2020, Proceedings*, page 508–520, Berlin, Heidelberg. Springer-Verlag.

Vaicenavicius, J., Widmann, D., Andersson, C., Lindsten, F., Roll, J., and Schön, T. (2019). Evaluating model calibration in classification. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 3459–3467. PMLR.

Van den Broeck, G., Lykov, A., Schleich, M., and Suciu, D. (2021). On the tractability of SHAP explanations. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*.

Yang, L., Manias, D. M., and Shami, A. (2021). Pwpae: An ensemble framework for concept drift adaptation in iot data streams. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pages 01–06.

Yang, L. and Shami, A. (2021). A lightweight concept drift detection and adaptation framework for iot data streams. *IEEE Internet of Things Magazine*, 4:96–101.

Yang, L. and Shami, A. (2023). A multi-stage automated online network data stream analytics framework for IIoT systems. *IEEE Transactions on Industrial Informatics*, 19(2):2107–2116.