Computer Science (CO-SCIENCE)
Volume 6 Issue 1 July 2026
Accreditation **Sinta 3** No. SK : 295/C/C3/KPT/2026

Integrating MLOps for Monitoring, Drift Detection, and Adaptive Retraining in Sensor Systems

Novia Heriyani[1], Nita Merlina[2*]

[1, 2]Universitas Nusa Mandiri,
Jl Marginda No 545, Depok, Jawa Barat, Indonesia

e-mail: [1]14230034@nusamandiri.ac.id, [2]nita@nusamandiri.ac.id

(*) Corresponding Author

Article Info: Received: 09-03-2026 | Revised : 12-07-2026 | Accepted : 16-07-2026

Abstracts - *Machine learning models deployed in sensor-based production environments are prone to performance degradation due to evolving data distributions, commonly known as data drift. In gas sensor systems, such drift often arises from environmental variability and sensor aging, which can significantly reduce predictive reliability if left unaddressed. This study presents an integrated Machine Learning Operations (MLOps) framework that combines performance monitoring, distribution-based drift detection, and adaptive retraining within a unified production pipeline. Experiments were conducted using the Gas Sensor Array Drift Dataset, organized into sequential batches to emulate real-world deployment conditions. Data drift was quantified using the Population Stability Index (PSI) and Kullback–Leibler (KL) divergence, which served as decision criteria for adaptive retraining. The proposed adaptive retraining strategy was systematically compared with baseline (no retraining) and periodic retraining approaches. Experimental results show that the proposed framework achieved an average accuracy of up to 0.74 and an F1-score of 0.72, outperforming the baseline strategy while maintaining performance comparable to periodic retraining. Because all evaluated batches consistently exceeded the predefined drift thresholds, adaptive retraining was triggered throughout the evaluation, demonstrating the effectiveness of the proposed event-driven MLOps pipeline under continuous sensor drift. Additionally, the use of containerization and experiment tracking ensures reproducibility and supports full lifecycle traceability. Overall, this study demonstrates that integrating monitoring, statistical drift detection, and adaptive retraining within an MLOps framework improves the operational robustness of sensor-based machine learning systems in dynamic environments.*

Keywords : *Adaptive Retraining, Data Drift Detection, Gas Sensor Data, Machine Learning Operations (MLOps), Production Machine Learni*

INTRODUCTION

Machine learning (ML) has become a core component in industrial sensor systems, where it supports operational efficiency and enables data-driven decision-making. One notable application is the electronic nose (E-nose), which employs arrays of gas sensors for chemical classification and air quality monitoring (Li et al., 2026). Despite achieving strong performance during training, ML models often experience a decline in accuracy once deployed in real-world environments. A primary factor behind this issue is data drift, defined as changes in the statistical distribution of input data over time (Wörner et al., 2025).

In the context of metal-oxide gas sensors, drift arises from multiple sources. Material aging leads to first-order drift (Müller & Sberveglieri, 2022), while environmental factors such as temperature and humidity variations contribute to second-order drift (Abdullah et al., 2022). These shifts violate the independent and identically distributed (i.i.d.) assumption on which most ML models rely, ultimately reducing predictive performance.

To address this challenge, Machine Learning Operations (MLOps) introduces continuous monitoring within production systems (Balasubramanian, 2020). Drift detection is typically performed using statistical techniques such as the Kolmogorov Smirnov test (Mohan Raja Pulicharla, 2019), which enables the system to identify distributional changes automatically. When drift is detected, adaptive retraining can be triggered, allowing models to update only when necessary. This approach is event-driven retraining than periodic retraining, which updates models at fixed intervals regardless of actual performance changes.

Previous studies have proposed various approaches to mitigate gas sensor drift, including incremental learning (Lorwongtrakool & Meesad, 2020), domain adaptation (Ma et al., 2018), ensemble learning (Das et al., 2020; Dong et al., 2020), and deep learning-based methods (Zhang et al., 2021). These approaches have successfully improved drift compensation and classification performance; however, they are predominantly evaluated as standalone algorithmic solutions under offline experimental settings. As a result, these components are commonly evaluated independently rather than being integrated into a unified operational workflow capable of supporting long-term machine learning deployment. This observation is reinforced by the comparative analysis summarized in Table 1, which indicates that existing studies predominantly address drift compensation from an algorithmic perspective, whereas production-oriented MLOps capabilities remain largely unexplored.

The Gas Sensor Array Drift Dataset proposed by Kumar and Chouksey (Kumar & Chouksey, 2023) serves as a relevant benchmark for analyzing drift in E-nose systems. The dataset was collected over a 36-month period using 16 metal-oxide semiconductor sensors across six types of gases, capturing gradual signal changes caused by both sensor aging and environmental variability. Its temporal structure makes it particularly suitable for evaluating production-oriented machine learning systems under continuously evolving data distributions.

Motivated by these limitations, this study proposes an integrated MLOps framework that unifies performance monitoring, statistical drift detection, adaptive retraining, and deployment management within a single production pipeline. Unlike previous studies that primarily improve individual algorithms or evaluate isolated components, the proposed framework focuses on operational integration across the complete machine learning lifecycle. Rather than introducing a new classification algorithm, this work emphasizes how monitoring, drift detection, retraining decisions, and deployment can be systematically orchestrated to maintain model reliability under continuously evolving sensor data. To validate the proposed framework, three retraining strategies, baseline, periodic retraining, and adaptive retraining, are comparatively evaluated using the Gas Sensor Array Drift Dataset. This comparison provides empirical evidence of how integrated MLOps practices improve model robustness and support long-term operational sustainability in dynamic sensor environments.

Accordingly, the primary contribution of this study lies in operationalizing MLOps for sensor-based machine learning by integrating monitoring, drift detection, adaptive retraining, and deployment management into a unified production-oriented framework, rather than proposing a new predictive algorithm.

RELATED WORK

Previous studies on gas sensor drift compensation can generally be categorized into four research directions: incremental learning, domain adaptation, ensemble learning, and deep learning-based approaches. These methods consistently report substantial improvements in classification accuracy and drift compensation when evaluated using benchmark datasets such as the Gas Sensor Array Drift Dataset. However, despite differences in modeling techniques, most studies share a common characteristic: they primarily focus on algorithmic optimization under offline experimental settings rather than on the operational management of machine learning models after deployment.

Table 1. Literature Review

Researcher	Title	Method / Topic	Contribution	Research Gap
(Kumar & Chouksey, 2023)	Gas Sensor Array Drift in an E-Nose System: A Dataset for Machine Learning Applications	Dataset Gas Sensor Array, PCA, ANN	Provides a comprehensive gas sensor drift dataset and drift pattern analysis, widely used for validating drift detection and compensation techniques.	Only provides a dataset; does not discuss production pipeline implementation or long-term model sustainability.
(Lorwongtrakool & Meesad, 2020)	Correlation-based incremental learning network for gas sensors drift compensation classification	Incremental Learning with Correlation Distance & Gaussian MF	Achieved 98.96% accuracy; preserves prior knowledge while adapting to new patterns.	Focuses on incremental learning algorithms; lacks integration of performance monitoring in production environments.
(Ma et al., 2018)	Weighted Domain Transfer Extreme Learning Machine and Its Online Version for Gas Sensor Drift Compensation in E-Nose Systems	WDTELM, online learning, clustering-aided sample selection	Reduces labeled data requirements by up to 20% ($R^2 \approx 0.95$, high relative accuracy); enables online learning with constant time complexity and comparable offline accuracy.	Focuses solely on domain adaptation; does not support a continuous feedback loop.
(Manna, 2020)	Drift Compensation for Electronic Nose by Multiple Classifiers System with Genetic Algorithm Optimized Feature Subset	Ensemble SVM, Genetic Algorithm, Feature Selection	Feature subset optimization significantly improves classification accuracy under 3-year drift conditions. Accuracy improved from 67.87% to 69.27% (Ensemble + GA).	Does not address real-world production sustainability; limited to offline evaluation..
(Dong et al., 2020)	Research on Recognition Model with Random Forest and Entropy Weight for Chemical Gas Sensor Array	RF, Entropy Weight, Bootstrap Aggregating	Robust against imbalanced distributions and temporal drift; achieved high accuracy (>95%) across multiple UCI dataset batches.	No production pipeline integration; limited to dataset-based experiments.
(Das et al., 2020)	Gas Sensor Drift Compensation by Ensemble of Classifiers Using Extreme Learning Machine	ELM, Ensemble Learning, Majority Voting	Proposes ELM ensemble to enhance drift stability. Achieved 92.6% accuracy (Batch 5, UCI drift dataset, ensemble of 5 ELMs), with average accuracy above 90% across other batches.	Does not consider real-time model performance monitoring.

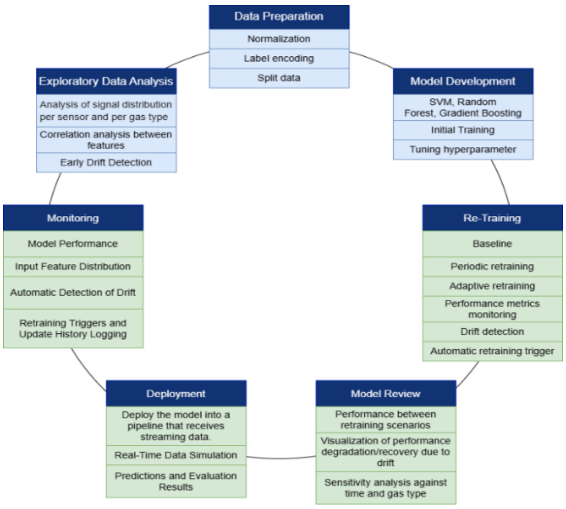
(Zhang et al., 2021)	A Novel Gas Recognition and Concentration Estimation Model for an Artificial Olfactory System with a Gas Sensor Array	Deep Learning, BiLSTM, GRU, Time-Frequency Feature Extraction	Achieves highly accurate gas concentration estimation in mixed gases; R^2 up to 95.3% with low MAE; outperforms 8 classical algorithms.	Focused on high accuracy; does not consider long-term sustainability.
(Ajayi et al., 2024)	Voting Ensemble Learning Model (VELM) for Harmful Gas Detection in Environmental Applications	Voting Ensemble Learning Model, RF, SVM, Logistic Regression	Achieved 99.46% accuracy; stable under various conditions; suitable for field implementation.	Accuracy-oriented; lacks automatic drift detection mechanisms.
(Dong et al., 2021)	Online inertial machine learning for sensor array long-term drift compensation	Online Self-training, Inertial ML, Data Cleaning, SVM	Extends sensor effectiveness by 4–10 months; prediction time <300 ms; suitable for real-world applications.	Does not integrate model performance monitoring into a production lifecycle.

Source: Research Result (2026)

As summarized in Table 1, although previous studies employ different modeling strategies, they converge on a common objective of improving predictive accuracy or compensating for sensor drift. Comparatively less attention has been devoted to operational aspects such as continuous performance monitoring, automated drift detection, retraining decision-making, and deployment lifecycle management. Consequently, existing solutions remain largely algorithm-centric and provide limited support for maintaining model reliability in long-term production environments. This common limitation establishes the motivation for the present study, which focuses on integrating these operational capabilities within a unified MLOps pipeline rather than proposing another drift compensation algorithm.

RESEARCH METHOD

The study adopts a quantitative experimental approach grounded in system engineering principles to design and evaluate an adaptive machine learning system capable of maintaining performance under dynamic data conditions in production environments. The proposed framework follows the MLOps paradigm, structuring the workflow into a continuous and iterative cycle consisting of Exploratory Data Analysis (EDA), Data Preparation, Model Development, (Re-)Training, Model Review, Deployment, and Monitoring, as illustrated in Figure 1. This design emphasizes not only model development but also long-term maintenance through integrated performance monitoring, automated drift detection, and metric-driven retraining, forming a closed feedback loop that sustains predictive reliability over time(Ruf et al., 2021). MLOps is particularly relevant in this context because it extends beyond static model training to include post-deployment lifecycle management (Ikram & Tabassam, 2023). It enables automated detection of performance degradation and adaptive responses through retraining, ensuring that the model remains effective despite evolving data distributions. This is essential for gas sensor-based classification tasks, where non-stationary data is inherent due to environmental changes and sensor aging (Ahmed et al., 2025).



Source: Research Result (2026)

Figure 1. Proposed Adaptive MLOps Pipeline

Figure 1 depicts the iterative nature of the proposed system. The circular workflow highlights how each stage is interconnected, allowing continuous updates based on incoming data. Models including Support Vector Machine (SVM), Random Forest (RF), and Gradient Boosting (GB) are initially trained using the earliest available batch and subsequently evaluated on sequential batches to simulate a streaming environment (Zhao et al., 2019). During deployment, monitoring continuously evaluates both predictive performance and input data distribution. Adaptive retraining is triggered whenever predefined drift or performance thresholds are exceeded, after which the updated model is redeployed and evaluated on subsequent batches to maintain predictive reliability under evolving data distributions (Sahiner B et al., 2023). The proposed pipeline was implemented in Python using Scikit-learn for model development, MLflow for experiment tracking, Joblib for model persistence, and Docker for containerized deployment.

1. Dataset

The dataset used in this study is the Gas Sensor Array Drift Dataset introduced by Kumar and Chouksey (2023). It was collected over 36 months using 16 metal-oxide semiconductor (MOS) sensors to classify six types of gases. Each observation reflects real-world sensor responses, capturing both environmental variability and sensor aging effects. The dataset inherently contains concept drift and data drift, making it well-suited for evaluating adaptive MLOps systems. To simulate real-world deployment, the dataset is organized chronologically into temporal batches, where the earliest temporal batch (Batch 1) is used for initial model training, while Batches 2–10 are sequentially processed to simulate streaming data in a production environment.

2. Exploratory Data Analysis

During the EDA phase, the study examines data distribution, inter-feature relationships, and temporal stability across batches. Visualization techniques such as histograms, boxplots, and temporal plots are used to identify noise, inconsistencies, and early signs of drift. Principal Component Analysis (PCA) is applied to reduce dimensionality and provide a visual indication of distributional shifts over time. These insights guide preprocessing decisions and the design of drift simulation scenarios.

3. Data Preparation

The data preparation stage involves preprocessing steps such as handling missing values, outlier treatment, normalization, and label encoding. Data is structured chronologically and split into training, validation, and streaming test sets. This chronological partition preserves the temporal characteristics of the dataset and avoids information leakage from future batches into the training process. Normalization techniques such as Z-score standardization or min-max scaling are applied to ensure feature comparability, which is critical for models sensitive to scale. Label encoding transforms categorical gas labels into numerical representations suitable for classification algorithms.

4. Model Development

Model development focuses on supervised learning algorithms, specifically GB, SVM and RF, selected for their balance between performance and computational efficiency (Ajala et al., 2025). SVM is effective in handling high-dimensional and non-linear data (Rahmi, 2020), while RF provides robustness against noise and overfitting (Hossain Lipu et al., 2023). GB is included to evaluate ensemble boosting performance under drift conditions, given its ability to sequentially correct residual errors (Michael Ehiedu Usiagwu et al., 2025). Hyperparameter tuning is conducted using cross-validation to identify optimal configurations. The use of classical models is intentional, as the study prioritizes lifecycle management and system integration over proposing new model architectures. Accordingly, deep learning architectures are beyond the scope of this study, as the objective is to evaluate the proposed MLOps lifecycle and operational workflow rather than compare different predictive model architectures. Additionally, prior studies indicate that these models perform competitively on this dataset.

5. Model Training and Retraining

The training and retraining phase evaluates three strategies: baseline (no retraining), periodic retraining (fixed intervals), and adaptive retraining (triggered by drift detection). Drift is detected using Population Stability Index (PSI) and Kullback–Leibler (KL) divergence because they provide complementary measures of distributional change. PSI measures the magnitude of feature distribution shifts, whereas KL divergence quantifies the divergence between reference and incoming probability distributions. Adaptive retraining is triggered when PSI exceeds 0.25, KL divergence exceeds 0.30, or model accuracy decreases by more than 10% relative to the baseline performance. During adaptive retraining, the updated model is trained using the original reference training data together with the current incoming batch to preserve previously learned patterns while adapting to newly observed data.

6. Model Review

Drift and performance thresholds are determined empirically using baseline validation performance on early batches. The drift threshold is defined based on statistical divergence distribution observed during stable periods, while the performance threshold is set relative to initial validation accuracy. Candidate models generated during retraining are compared with the currently deployed model, and only models demonstrating improved validation performance are promoted for deployment. This ensures that retraining decisions are data-driven rather than arbitrarily defined.

7. Deployment

In the deployment stage, the selected model is integrated into a simulated production pipeline that processes data in a streaming manner. Each batch represents incoming real-world data, and predictions are generated sequentially. The system logs prediction results and evaluation metrics for longitudinal analysis.

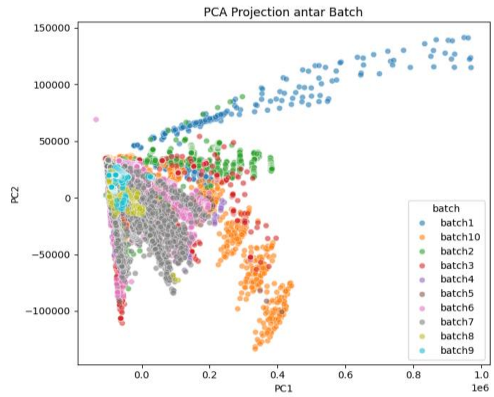
***8.* Monitoring**

Finally, the monitoring component serves as the core of the adaptive system. It continuously tracks model performance metrics and input data distributions. Drift detection mechanisms operate alongside performance monitoring to identify both explicit and latent changes in data. When predefined thresholds are exceeded, retraining is triggered automatically. All monitoring outputs, including drift events, retraining decisions, and model performance over time, are logged and visualized to support longitudinal analysis and decision-making.

Overall, system performance is evaluated by comparing the three retraining strategies using standard classification metrics and temporal analysis. Evaluation is conducted per batch to capture the impact of drift and the effectiveness of each adaptation strategy. This approach enables a clear assessment of how quickly the system detects degradation, initiates retraining, and recovers performance. This evaluation framework enables objective comparison of retraining strategies under controlled drift conditions. The experimental protocol consists of training the initial model using early batches, followed by sequential evaluation on subsequent batches under streaming simulation. For each retraining strategy, performance is recorded per batch, enabling longitudinal comparison of stability, recovery time after drift, and variance across time. Comparative effectiveness is assessed based on average performance across batches, performance variability, recovery following drift events, and retraining frequency.

RESULTS AND DISCUSSION

The initial stage of analysis focuses on understanding the structure, characteristics, and distributional patterns across the ten observed data batches. This step is essential, as variations between batches may introduce distribution shifts that directly affect the stability and generalization of predictive models.



Source: Research Result (2026)

Figure 2. PCA Projection across Batches

PCA is employed to examine the structural relationships between batches in a lower-dimensional space. The two-dimensional projection shown in Figure 2. PCA Projection across Batches indicates that each batch forms a relatively distinct cluster. Although partial overlaps exist, several batches particularly Batch 1, Batch 3, and Batch 6 exhibit noticeably different distributions compared to others. This separation suggests that the feature distributions are not homogeneous across time. Furthermore, the shift in cluster positions along the PC1–PC2 axes, especially between earlier and later batches, reflects the presence of temporal drift. This drift is likely influenced by changing environmental conditions or sensor degradation. As a result, models trained on earlier batches may experience reduced generalization when applied to subsequent data without proper adaptation. To quantify the extent of distributional changes, the PSI is used as a divergence metric between reference and observed distributions (Azeem, 2025). PSI is calculated as:

$$PSI = \sum_{i=1}^{B} (P_i - Q_i)\, ln\left(\frac{P_i}{Q_i}\right) \qquad (1)$$

Description of symbols:
P_i = represents the observed proportion in bin i
Q_i = denotes the reference proportion,
B = is the total number of bins.

In practice, PSI values below 0.1 indicate stable distributions, values between 0.1 and 0.25 suggest moderate drift, and values above 0.25 indicate significant drift that may require retraining. Mean PSI values well above the 0.25 threshold, confirming substantial distributional shifts over time. In several cases, the PSI values are extremely high, indicating that the observed distributions diverge significantly from the reference. This implies that a model trained solely on Batch 1 is highly likely to suffer performance degradation when applied to later batches. PSI values exceeding 7 further suggest that the underlying data distribution has fundamentally changed, violating the stationarity assumption required by most machine learning models.

Table 2. PSI values of each batch relative to the reference (Batch 1) and Rolling PSI calculation results (Batch_t vs Batch_(t-1))

Batch	Mean PSI	Batch	Mean Rolling PSI
Batch 2	0.83	Batch 2	0.92
Batch 3	1.88	Batch 3	0.59
Batch 4	5.67	Batch 4	1.92
Batch 5	7.6	Batch 5	3.17
Batch 6	3.1	Batch 6	1.75
Batch 7	2.78	Batch 7	0.48
Batch 8	6.47	Batch 8	3.95
Batch 9	7.53	Batch 9	2.19
Batch 10	1.23	Batch 10	1.23

Source: Research Result (2026)

To better capture the temporal dynamics of drift, a rolling PSI analysis was performed, comparing each batch with its immediate predecessor. Table 2 shows that drift does not increase steadily over time but occurs in irregular bursts. Some batch transitions exhibit sharp PSI spikes, while others remain relatively stable. For example, Batch 7 presents a low rolling PSI, indicating minimal change from the previous batch, despite a significant difference from the initial reference. This pattern suggests that drift in this dataset is episodic, with intermittent periods of relative stability. Comparing PSI values relative to the reference with rolling PSI results reveals an important insight. Although cumulative drift from the initial batch is consistently high, incremental changes between consecutive batches are often moderate. This indicates that distributional shifts progress in stages, accumulating into substantial overall deviations. Figure 2 supports this observation, showing that spikes in rolling PSI occur at specific points rather than following a smooth upward trajectory, reflecting non-linear drift behavior.

Experiments were replicated across three models SVM, RF, and GB using fixed thresholds of 0.25 for PSI, 0.30 for KL divergence, and 0.10 for performance degradation. Evaluation per batch employed accuracy and F1-score to measure both stability and recovery under drift. In the baseline scenario Table 3, models trained only on Batch 1 were tested on subsequent batches without updates. Results show a consistent performance decline across all models as batches progressed. While Batch 2 achieved near-perfect performance, sharp degradation occurred from Batch 3 to Batch 9. Average accuracy dropped to 0.46 for SVM, 0.55 for RF, and 0.52 for GB, with F1-scores following a similar trend. These results highlight the models' inability to generalize under shifting data distributions, confirming the violation of the stationarity assumption. This aligns with prior drift analyses, where distributional changes directly affected model performance

Table 3. Baseline And Periodic Result

Baseline	**GB**		**RF**		**SVM**		**Periodic**	**GB**		**RF**		**SVM**	
Batch	**ACC**	**F1**	**ACC**	**F1**	**ACC**	**F1**	**Batch**	**ACC**	**F1**	**ACC**	**F1**	**ACC**	**F1**
2	0.99	0.99	0.99	0.99	0.99	0.99	2	0.37	0.29	0.54	0.47	0.54	0.43
3	0.73	0.73	0.74	0.75	0.67	0.69	3	0.73	0.74	0.75	0.76	0.71	0.73
4	0.65	0.59	0.71	0.66	0.54	0.49	4	0.89	0.89	0.92	0.92	0.69	0.61
5	0.68	0.58	0.53	0.5	0.4	0.4	5	0.66	0.68	0.79	0.76	0.71	0.76
6	0.35	0.24	0.43	0.34	0.24	0.21	6	0.89	0.9	0.83	0.82	0.9	0.9
7	0.46	0.37	0.49	0.38	0.43	0.34	7	0.35	0.33	0.4	0.45	0.47	0.45
8	0.39	0.3	0.38	0.27	0.36	0.27	8	0.66	0.57	0.69	0.61	0.75	0.71
9	0.17	0.08	0.22	0.1	0.2	0.12	9	0.82	0.79	0.85	0.83	0.89	0.88
10	0.22	0.13	0.43	0.33	0.34	0.2	10	0.4	0.41	0.93	0.93	0.95	0.95
Mean	**0.52**	**0.45**	**0.55**	**0.48**	**0.46**	**0.41**	**Mean**	**0.64**	**0.62**	**0.74**	**0.73**	**0.73**	**0.71**

Source: Research Result (2026)

In contrast, the periodic retraining scenario Table 3 demonstrates a marked improvement. By updating the model at every batch, average accuracy increases to 0.73 (SVM), 0.74 (RF), and 0.64 (GB). Beyond the improved averages, the models exhibit a recovery pattern after each retraining step, although some fluctuations remain, particularly in Batch 7. This suggests that periodic retraining is effective in mitigating performance degradation caused by drift. However, the approach remains static, as it does not account for the actual state of the data distribution. Consequently, retraining is triggered for every batch, introducing potential computational inefficiencies due to unnecessary updates.

Table 4. Adaptive Result

Batch	**GB Acc**	**GB F1**	**GB Time(s)**	**RF Acc**	**RF F1**	**RF Time(s)**	**SVM Acc**	**SVM F1**	**SVM Time(s)**	**PSI**	**KL**
2	0.15	0.15	79.63	0.28	0.22	2.24	0.31	0.27	0.87	1.12	0.71
3	0.85	0.84	35.77	0.86	0.84	0.84	0.91	0.91	0.74	2.05	1.39
4	0.9	0.9	62.77	0.93	0.93	1.25	0.68	0.57	1.27	4.55	1.07
5	0.83	0.85	30.03	0.84	0.82	0.65	0.76	0.8	0.12	1.44	1.05
6	0.85	0.83	5.61	0.93	0.94	0.24	0.91	0.91	0.03	3.3	1.69
7	0.52	0.55	55.57	0.51	0.55	1.25	0.43	0.41	0.18	2.68	1.31
8	0.68	0.6	146.64	0.67	0.59	3.3	0.75	0.71	0.8	0.31	0.14
9	0.85	0.82	91.78	0.85	0.83	2.17	0.9	0.9	0.52	2.04	1.69
10	0.67	0.63	15.54	0.8	0.77	0.36	0.78	0.72	0.06	8.2	3.46
Mean	**0.7**	**0.69**		**0.74**	**0.72**		**0.72**	**0.69**			

Source: Research Result (2026)

To address this limitation, the adaptive retraining scenario Table 4 introduces a conditional update mechanism, where retraining is triggered only when significant drift is detected. The results indicate that this approach maintains performance at a level comparable to periodic retraining, with average accuracy values of 0.72 (SVM), 0.74 (RF), and 0.70 (GB). In early stages (e.g., Batch 2), lower performance coincides with elevated PSI and KL divergence values, which then trigger retraining and lead to improved performance in subsequent batches. This pattern is consistently observed across all models, indicating that the drift detection mechanism functions effectively as a trigger for performance recovery. From an efficiency perspective, notable differences in retraining duration are observed across models. GB exhibits higher and more variable computational cost, whereas RF and SVM remain relatively stable and faster. Within an MLOps context, this distinction is critical, as retraining frequency and duration directly influence system latency and operational cost. Therefore, RF and SVM provide a more favorable trade-off between performance and efficiency. Besides predictive performance, computational cost is also an important consideration for production deployment. As shown in Table X, Random Forest consistently required substantially shorter retraining time than Gradient Boosting while achieving comparable predictive performance. This indicates that RF provides a favorable balance between predictive accuracy and computational efficiency for adaptive MLOps deployment.

Table 5. Comparison of Average Performance of Three Retraining Strategies

Model	Baseline Acc	Periodic Acc	Adaptive Acc	Baseline F1	Periodic F1	Adaptive F1	Periodic Retrain	Adaptive Retrain
SVM	0.46	0.73	0.72	0.41	0.71	0.69	9	9
RF	0.55	0.74	0.74	0.48	0.73	0.72	9	9
GB	0.51	0.64	0.70	0.45	0.62	0.68	9	9

Source: Research Result (2026)

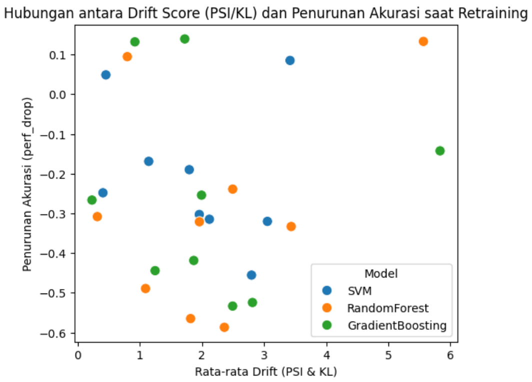
The aggregate comparison in Table 5 confirms that both periodic and adaptive retraining consistently outperform the baseline, while achieving comparable average performance. Although adaptive retraining resulted in the same number of retraining sessions as periodic retraining, this outcome reflects the characteristics of the Gas Sensor Array Drift Dataset rather than a limitation of the proposed framework. As shown in Table 2, the average PSI values of all incoming batches exceeded the predefined threshold (PSI > 0.25), indicating persistent distributional shifts throughout the data stream. This finding is consistent with the PCA visualization in Figure 2, which shows continuous distribution changes across temporal batches without prolonged stable periods. Consequently, adaptive retraining was triggered for every batch, resulting in the same retraining frequency as the periodic strategy. Therefore, the advantage of the proposed adaptive approach lies not in reducing retraining frequency, but in its event-driven decision mechanism, where retraining is performed only when statistically significant drift or performance degradation is detected. Under production environments with intermittent rather than continuous drift, this mechanism is expected to avoid unnecessary retraining while maintaining predictive performance.

Table 6. Results of Controlled Drift Experiment

	batch	*type*	*accuracy*	f1	psi	kl	drift	retrain
1	batch2	*low_drift*	0.997753	0.997752	0.04787	0.012185	FALSE	FALSE
2	batch3	*high_drift*	0.204494	0.128164	3.705605	0.257858	TRUE	TRUE
3	batch4	*low_drift*	0.188764	0.139855	0.047762	0.010468	FALSE	FALSE
4	batch5	*high_drift*	0.18427	0.161348	3.724571	0.259016	TRUE	TRUE
5	batch6	*low_drift*	0.159551	0.140521	0.047866	0.01163	FALSE	FALSE

Source: Research Result (2026)

Additional experiments under controlled drift conditions Table 6 further evaluate system sensitivity. In the low-drift scenario, PSI values remain below the defined threshold, and no retraining is triggered, indicating that the system does not overreact to minor fluctuations. Conversely, under high-drift conditions, PSI increases significantly, followed by performance degradation and consistent retraining activation. These results confirm that the pipeline can distinguish between minor and substantial distributional shifts and respond proportionally. Overall, the results presented in Tables 3 to 6 demonstrate that integrating monitoring, drift detection, and adaptive retraining within an MLOps framework effectively sustains model performance in dynamic streaming data environments. Compared to static approaches, the adaptive strategy not only maintains competitive performance but also improves operational efficiency, making it more appropriate for real-world production systems.



Source: Research Result (2026)

Figure 3. Relationship Between Distribution Drift and Model Performance Degradation

The relationship between distribution drift and model performance degradation was examined using two complementary metrics: PSI and KL. The relationship is illustrated in Figure 3, which presents a scatter plot comparing the average drift score (PSI and KL) against the drop in accuracy before retraining occurs. The visualization indicates that high drift values do not always correspond to severe performance degradation. In several batches, PSI and KL values increase substantially while the model maintains relatively stable accuracy after retraining. This suggests that the adaptive retraining mechanism can partially mitigate the impact of distribution shifts. However, some points in Figure 3 reveal a different pattern: extremely high drift values tend to coincide with a sharp decline in model accuracy prior to retraining. This observation indicates that substantial distribution changes can undermine prediction reliability if not addressed promptly. The analysis therefore reinforces the importance of drift detection as a critical component within the MLOps architecture, particularly for sensor-based systems where data distributions evolve over time.

Table 7. Summary of Prediction Results and Model Evaluation at the Deployment Stage

stream_idx	batch_ name	label_ available	accuracy_ deployed	f1_ deployed	psi_ mean	kl_ mean	drift_ flag	pending_ candidate	eval_batch_ used
1	batch2	No	NaN	NaN	2.464	2.63	TRUE	FALSE	–
2	batch3	Yes	0.674	0.691	1.634	1.387	TRUE	TRUE	batch2
3	batch4	Yes	0.678	0.574	1.413	1.673	TRUE	FALSE	batch3
4	batch5	Yes	0.553	0.491	1.787	2.450	TRUE	TRUE	batch4
5	batch6	Yes	0.716	0.673	1.837	1.127	TRUE	FALSE	batch5
6	batch7	Yes	0.60	0.634	1.84	0.674	TRUE	TRUE	batch6
7	batch8	Yes	0.75	0.71	2.798	3.332	TRUE	FALSE	batch7
8	batch9	Yes	0.81	0.784	2.038	1.636	TRUE	TRUE	batch8
9	batch10	Yes	0.78	0.718	2.61	4.824	TRUE	FALSE	batch9

Source: Research Result (2026)

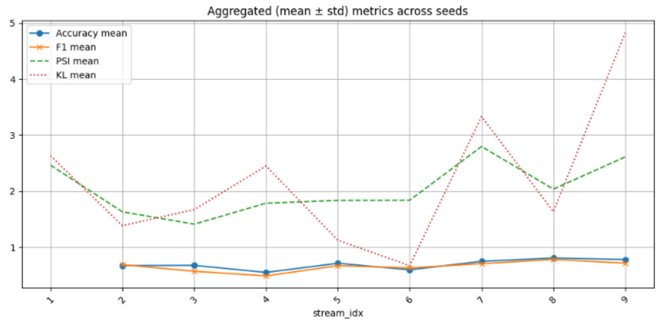
Model performance during the deployment phase is summarized in Table 7, which reports prediction results for each streaming batch along with label availability, drift measurements, and candidate model evaluation status. Each row represents a single batch-processing cycle and includes accuracy and F1-score from the currently deployed model, as well as PSI and KL values used to detect distribution shifts. Additional fields such as *drift flag*, *pending candidate*, and *evaluation batch used* describe the decision process implemented in the proposed MLOps pipeline. The results in Table 7 show that several batches triggering retraining specifically batches 3, 5, 7, and 9 are followed by noticeable performance improvements in subsequent batches. This pattern indicates that the pipeline can detect distribution changes in a timely manner and generate candidate models better aligned with the updated data distribution. The consistently high PSI and KL values across many batches also suggest that the gas sensor dataset exhibits persistent drift, a behavior consistent with the phenomenon of *sensor aging* in Metal Oxide Semiconductor sensor systems.

Table 8. Results of Drift Monitoring, Retraining, Candidate Evaluation, and Model Performance per Batch

Batch	PSI	KL	Drift?	Retraining Triggered?	Candidate Evaluated?	Candidate Promoted?	Acc/F1 Deployed (at Current Batch)	Acc/F1 Candidate
2	2.464	2.633	Yes	No	No	No	0 / 0	–
3	1.634	1.387	Yes	Yes (drift=2)	Yes	Yes (promoted)	0.542 / 0.487	0.678 / 0.574
4	1.413	1.673	Yes	No	No	No	0.678 / 0.574	–
5	1.787	2.45	Yes	Yes (drift=2)	Yes	Yes (promoted)	0.660 / 0.556	0.716 / 0.673
6	1.836	1.127	Yes	No	No	No	0.716 / 0.673	–
7	1.84	0.674	Yes	Yes (drift=2)	Yes	Yes (promoted)	0.521 / 0.500	0.750 / 0.709
8	2.798	3.332	Yes	No	No	No	0.750 / 0.709	–
9	2.038	1.636	Yes	Yes (drift=2)	Yes	Yes (promoted)	0.555 / 0.511	0.783 / 0.718
10	2.612	4.824	Yes	No	No	No	0.783 / 0.718	–

Source: Research Result (2026)

Continuous monitoring of model and data quality during deployment is further detailed in Table 8. This table provides a comprehensive overview of drift values for each batch, adaptive retraining decisions, candidate model evaluations, and performance comparisons between deployed and candidate models. The results reveal a repeated drift pattern occurring in batches 3, 5, 7, and 9, each of which triggers adaptive retraining. In these cases, the newly generated candidate model consistently outperforms the previously deployed model and is subsequently promoted to replace it. Conversely, batches showing only a single drift indication do not immediately trigger retraining. This behavior demonstrates that the pipeline is not only responsive to distribution changes but also capable of avoiding unnecessary model updates. As a result, the adaptation strategy remains selective and context-aware, ensuring that retraining occurs only when justified by the data conditions.



Source: Research Result (2026)

Figure 4. Aggregate trends of performance and drift metrics

Aggregate trends of performance and drift metrics throughout the streaming process are shown in Figure 4, which plots the average values of accuracy, F1-score, PSI, and KL divergence for each batch. The figure shows that accuracy and F1-score tend to decline when drift increases, but improve again after retraining is applied. Meanwhile, PSI and KL values remain relatively high throughout the streaming process, indicating continuous distribution changes in the sensor data. The sharp increase in drift observed in certain batches particularly batches 7 and 9 highlights the strong drift characteristics of the Gas Sensor Array dataset, which are associated with gradual sensor degradation. Despite these conditions, the trend in Figure 4 demonstrates that the adaptive retraining mechanism integrated within the pipeline can maintain relatively stable prediction quality even under significant drift.

Table 9. Performance Improvement over Baseline Model

Model	Baseline	Periodic	Adaptive	Δ Adaptive vs Baseline	Δ Periodic vs Baseline
GB	0.513	0.642	0.699	0.186	0.128
RF	0.547	0.744	0.742	0.196	0.19
SVM	0.465	0.735	0.715	0.25	0.27

Source: Research Result (2026)

A summary of performance improvements relative to the baseline scenario is presented in Table 9. The results show that all models achieve substantial accuracy gains under retraining scenarios. The Gradient Boosting model improves by 0.186 in the adaptive scenario and 0.128 in the periodic scenario. Random Forest shows a balanced improvement of 0.196 (adaptive) and 0.190 (periodic). Meanwhile, the SVM model records the largest absolute improvement, with accuracy gains of 0.250 in the adaptive scenario and 0.270 in the periodic scenario. Overall, the experimental results indicate that retraining strategies both periodic and adaptive consistently improve model performance compared with the baseline scenario without updates. Although periodic retraining occasionally produces slightly higher performance, the adaptive approach remains more efficient because model updates occur only when distribution shifts or significant performance degradation are detected. These findings support the use of drift-driven adaptive retraining as an effective strategy for MLOps systems operating in dynamic sensor data environments.

CONCLUSION

This study designed and implemented an adaptive MLOps pipeline for a gas classification system based on metal oxide semiconductor sensors operating in a dynamic streaming environment. The experimental results demonstrate that integrating performance monitoring, statistical drift detection, and adaptive retraining within a unified MLOps pipeline helps maintain predictive performance under evolving data distributions caused by sensor drift. The proposed framework enables automated monitoring, drift detection, and model updating while preventing data leakage during retraining, thereby supporting more reliable deployment of machine learning models in non-stationary environments. Rather than proposing a new classification algorithm, the primary contribution of this work lies in integrating these operational components into a production-oriented machine learning lifecycle. This study has several limitations. First, the evaluation was conducted using a single publicly available gas sensor drift dataset and three conventional machine learning models. Second, because the dataset exhibits substantial drift across nearly all temporal batches, adaptive retraining was triggered throughout the evaluation, limiting the assessment of potential reductions in retraining frequency under less dynamic conditions. Furthermore, deep learning models were not included because the objective of this study was to evaluate the proposed MLOps lifecycle rather than compare predictive architectures. Previous studies have also shown that conventional machine learning models remain competitive on the Gas Sensor Array Drift Dataset, making them appropriate for evaluating the operational aspects of the proposed framework. Future work may evaluate the proposed framework using datasets with different drift characteristics, investigate additional drift detection techniques, incorporate deep learning models, and explore continual learning approaches to improve adaptation efficiency across a broader range of production scenarios.

REFERENCES

Abdullah, A. N., Kamarudin, K., Kamarudin, L. M., Adom, A. H., Mamduh, S. M., Mohd Juffry, Z. H., & Bennetts, V. H. (2022). Correction Model for Metal Oxide Sensor Drift Caused by Ambient Temperature and Humidity. *Sensors*, *22*(9), 3301. https://doi.org/10.3390/s22093301

Ahmed, A., del Rio-Chanona, E. A., & Mercangöz, M. (2025). Comparative Study of Machine Learning and System Identification for Process Systems Engineering Dynamics. *Industrial & Engineering Chemistry Research*, *64*(8), 4450–4478. https://doi.org/10.1021/acs.iecr.4c03264

Ajala, A. A., Adeoye, O. L., Salami, O. M., & Jimoh, A. Y. (2025). An examination of daily CO2 emissions prediction through a comparative analysis of machine learning, deep learning, and statistical models. *Environmental Science and Pollution Research*, *32*(5), 2510–2535. https://doi.org/10.1007/s11356-024-35764-8

Ajayi, O. F., Udosen, A. A., Ajayi, W., Ohwo, B. O., & Amusa, A. I. (2024). Voting Ensemble Learning Model (VELM) for Harmful Gas Detection in Environmental Applications. *Asian Journal of Electrical Sciences*, *13*(2), 45–50. https://doi.org/10.70112/ajes-2024.13.2.4252

Azeem, M. (2025). *Population Stability Index (PSI)*.

Balasubramanian, A. (2020). End-to-end model lifecycle management: An MLOPS framework for drift detection, root cause analysis, and continuous retraining. *International Journal of Multidisciplinary Research and Growth Evaluation*, *1*(1), 92–102. https://doi.org/10.54660/.IJMRGE.2020.1.1-92-102

Das, P., Manna, A., & Ghoshal, S. (2020). *Gas Sensor Drift Compensation by Ensemble of Classifiers Using Extreme Learning Machine*. IEEE.

Dong, X., Han, S., Wang, A., & Shang, K. (2021). Online inertial machine learning for sensor array long-term drift compensation. *Chemosensors*, *9*(12). https://doi.org/10.3390/chemosensors9120353

Dong, X., Qi, X., Cui, J., Xu, X., & Wan, A. (2020). *Research on Recognition Model with Random Forest and Entropy Weight for Chemical Gas Sensor Array*. IEEE.

Hossain Lipu, M. S., Hannan, M. A., Hussain, A., Ansari, S., Rahman, S. A., Saad, M. H. M., & Muttaqi, K. M. (2023). Real-Time State of Charge Estimation of Lithium-Ion Batteries Using Optimized Random Forest Regression Algorithm. *IEEE Transactions on Intelligent Vehicles*, *8*(1), 639–648. https://doi.org/10.1109/TIV.2022.3161301

Ikram, A., & Tabassam, U. (2023). *MLOps: A Step Forward to Enterprise Machine Learning*.

Kumar, J. R. R., & Chouksey, P. (2023). Gas Sensor Array Drift in an E-Nose System: A Dataset for Machine Learning Applications. *International Journal on Recent and Innovation Trends in Computing and Communication*, *11*(6), 167–171. https://doi.org/10.17762/ijritcc.v11i6.7343

Li, M., Wu, C., Wang, Z., Wu, Z., Huang, W., Chen, J., Yu, K., Wen, T., Yin, H., & Wang, Z. (2026). Enhanced Gas Classification in Electronic Nose Systems Using an SMOTE-Augmented Machine Learning Framework. *Sensors*, *26*(2), 714. https://doi.org/10.3390/s26020714

Lorwongtrakool, P., & Meesad, P. (2020). Correlation-based incremental learning network for gas sensors drift compensation classification. *Advances in Science, Technology and Engineering Systems*, *5*(4), 660–666. https://doi.org/10.25046/AJ050479

Ma, Z., Luo, G., Qin, K., Wang, N., & Niu, W. (2018). Weighted Domain Transfer Extreme Learning Machine and Its Online Version for Gas Sensor Drift Compensation in E-Nose Systems. *Wireless Communications and Mobile Computing*, *2018*(1). https://doi.org/10.1155/2018/2308237

Manna, A. (2020). *Drift Compensation for Electronic Nose by Multiple Classifiers System with Genetic Algorithm Optimized Feature Subset*. IEEE.

Michael Ehiedu Usiagwu, Mayowa Timothy Adesina, & Johnson Chinonso. (2025). Advanced machine learning models for real-time decision making in dynamic data environments. *International Journal of Science and Research Archive*, *14*(2), 852–865. https://doi.org/10.30574/ijsra.2025.14.2.0441

Mohan Raja Pulicharla. (2019). Detecting and addressing model drift: Automated monitoring and real-time retraining in ML pipelines. *World Journal of Advanced Research and Reviews*, *3*(2), 147–152. https://doi.org/10.30574/wjarr.2019.3.2.0189

Müller, G., & Sberveglieri, G. (2022). Origin of Baseline Drift in Metal Oxide Gas Sensors: Effects of Bulk Equilibration. *Chemosensors*, *10*(5), 171. https://doi.org/10.3390/chemosensors10050171

Rahmi, N. S. (2020). Ensemble-support vector machine-random undersampling: Simulation study of multiclass classification for handling high dimensional and imbalanced data. *Journal of Physics: Conference Series*, *1613*(1), 012064. https://doi.org/10.1088/1742-6596/1613/1/012064

Ruf, P., Madan, M., Reich, C., & Ould-Abdeslam, D. (2021). Demystifying mlops and presenting a recipe for the selection of open-source tools. *Applied Sciences (Switzerland)*, *11*(19). https://doi.org/10.3390/app11198861

Sahiner B, Chen W, Samala RK, & Petrick N. (2023). Data drift in medical machine learning: implications and potential remedies. *Br J Radiol*.

Wörner, J., Eimler, J., & Pein-Hackelbusch, M. (2025). Long-term drift behavior in metal oxide gas sensor arrays: a one-year dataset from an electronic nose. *Scientific Data*, *12*(1), 1628. https://doi.org/10.1038/s41597-025-05993-8

Zhang, W., Wang, L., Chen, J., Bi, X., Chen, C., Zhang, J., & Hans, V. (2021). A Novel Gas Recognition and Concentration Estimation Model for an Artificial Olfactory System with a Gas Sensor Array. *IEEE Sensors Journal*, *21*(17), 18459–18468. https://doi.org/10.1109/JSEN.2021.3091582

Zhao, X., Li, P., Xiao, K., Meng, X., Han, L., & Yu, C. (2019). Sensor drift compensation based on the improved lstm and svm multi-class ensemble learning models. *Sensors (Switzerland)*, *19*(18). https://doi.org/10.3390/s19183844