JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

AUTOMATED DATA OBSERVABILITY AND DRIFT DETECTION USING MACHINE LEARNING

Ashitosh Chitnis, Shishir Tewari
Business System Analyst, Data Application Engineer
Google LLC
Google Finance Corporate Engineering
Austin, TX, USA

Abstract

Modern applications dependent on data need to focus on data quality and reliability because it maintains reliable performance and trustworthy operations of machine learning models and analytical systems. Data observability systems and drift detection mechanisms operated by computers serve as essential elements for tracking incoming data stream accuracy and consistency. The usage of traditional methods to discover data drift and inconsistencies depends on manual human involvement and rule-based methods while showing inefficiency when applied to big and real-time data processes. The paper presents a study about how machine learning methods can perform autonomous drift detection and data observability to spot data distribution abnormalities beforehand. Our system based on machine learning uses statistical models together with anomaly detection algorithms and explainable AI (XAI) features which strengthens decision quality and interpretation in data monitoring processes. This paper evaluates the scalability issues along with false positives and describes real-time processing limitations before discussing AI-based adaptive drift correction systems of the future. The evaluation focuses on different drift detection approaches through a real-life example which proves that AI-powered tools help maintain data reliability.

Keyword: Data Observability, Drift Detection, Machine Learning Automation, Anomaly Detection, Real-Time Data Monitoring

1. Introduction

Modern enterprises need improved data quality management together with observation systems because their data volumes continue to expand rapidly. Organizations need high-quality data to support both business decision-making and model construction as well as establish dependable automated systems. Real-time data streams include inconsistencies and anomalies as well as distributional shifts that generate substantial effects on both analytical results and model operational outcomes. Data distribution changes that appear unexpectedly over time constitute data drift while such drift impairs predictive model accuracy and causes unneeded insights when the issue remains unidentified.

Large-scale data pipelines find difficulty when using traditional data quality monitoring techniques because they depend on manual inspection and rule-based checks systems. The inability to spot slight changes in data patterns through these inspection methods results in delayed necessary actions. A more automated solution provided by machine learning-based data observability tools detects anomalies during real-time data pipeline observation whereas it also creates immediate corrective actions. Organizations can strengthen their proactive data drift detection capabilities through combination of statistical models with anomaly detection techniques and deep learning algorithms.

The research investigates machine learning for automatic data observability and drift detection through a system which enhances data quality and minimizes human involvement and provides non-stop data pipeline monitoring. This research delivers three main contributions through its findings:

- This paper provides an extensive examination regarding traditional and machine learning-based approaches to detect data drift.
- This work presents an ML-driven automation system to track data observability alongside drift changes.
- A research study analyzes the operational success of artificial intelligence methodologies utilized to detect data drift.
- The paper examines current difficulties and prospective progress points which include dynamic drift corrections systems.

The subsequent part of this paper splits into two main sections: section two explores key difficulties in data observability alongside drift detection shortcomings. The paper investigates machine learning methods for resolving these challenges in Section 3. Section 4 introduces a system architecture that uses AI technology for data observability purposes. The paper concludes its presentation with implementation strategies along with performance evaluation in Section 5 followed by discussion and conclusions.

2. Challenges in Data Observability and Drift Detection

Organizations currently face multiple hurdles to preserve data quality and stability although they increase their reliance on automated data pipelines. The performance of machine learning models together with decision-making processes gets significantly affected due to data drift which defines the time-based modifications in incoming data statistical properties. Key difficulties in detecting data drift along with performing data observability monitoring will be explained in this part.

figure 1: data observability

2.1. Lack of Real-Time Monitoring and Response

The monitoring of traditional data systems depends on scheduled checks instead of using persistent real-time observation. Failure to detect problems on time results in declining model performance and flawed understanding as well as operational systems becoming less efficient. AI-driven systems require real-time drift detection because it ensures their operational accuracy as well as reliability levels.

2.2. Volume, Velocity, and Variety of Data

A variety of contemporary business operations handle substantial information amounts originating from numerous structured and unstructured data points. The constant high volume of data production makes manual system monitoring impractical so organizations need to implement automated observability tools. The detection of data drift becomes more difficult when systems handle different data types that extend from text and images to IoT sensor streams.

2.3. Identifying Different Types of Drift

Data drift occurs in multiple forms which need distinct detection protocols according to their variations.

- This type of drift affects data components while keeping the target variable unchanged.
- Concept Drift: Changes in the relationship between input features and output variables over time.
- The application of classification tasks encounters variations in the frequency distribution of their class labels known as prior probability shift.

The correct mitigation strategies need proper differentiation between the various drift types.

2.4. High False Positive and False Negative Rates

Drift detection models operating automatically need to achieve optimal results between sensitivity and specificity ratings. Drift detection systems which generate high numbers of incorrect alerts will trigger idle model retraining that increases operational costs yet incorrectly missed drift signals result in impaired model performance. The successful development of error-resistant drift detection systems poses an ongoing substantial challenge.

2.5. Limited Explainability in Machine Learning-Based Detection

Data engineers along with business analysts find black box operation in numerous AI-based drift detection models to be a challenge when attempting to decode drift alert triggering reasons. Trust as well as debugging and regulatory compliance require complete explainability in drift detection systems.

2.6. Scalability and Integration with Existing Pipelines

The detection of concept drift needs tools which handle big datasets while working effortlessly together with existing cloud platforms as well as on-site data systems. The inability of legacy systems to support modern AI-driven observability tools makes integration between older systems and contemporary tools an intricate procedure.

table 1: key challenges in data observability and drift detection

Challenge	Description	Impact
Lack of Real-Time Monitoring	Inability to detect and address drift instantly	Model degradation, incorrect insights
High Data Volume and Velocity	Rapidly changing and diverse data sources	Requires scalable automation
Multiple Types of Drift	Covariate, concept, and prior probability shift	Different detection strategies required
False Positives & False Negatives	Incorrect drift alerts or missed drifts	Computational inefficiencies, poor decisions
Limited Explainability	Black-box nature of ML-based detection	Reduces trust and interpretability
Scalability and Integration	Compatibility issues with enterprise systems	Complex deployment, increased costs

3. Machine Learning Techniques for Automated Data Observability and Drift Detection

Several machine learning (ML) techniques represent effective solutions which overcome the difficulties related to data observability and drift detection challenges. Complex algorithms use these techniques to search for unusual patterns within data while simultaneously checking data quality standards to preserve AI system stability. The procedures described in this section detail essential ML-based approaches for the automatic monitoring of data observation and drift detection systems.

3.1. Statistical Methods for Drift Detection

The detection of drift through traditional statistical approaches occurs through time-based comparison of data distribution trends. These methods include:

- The Kolmogorov-Smirnov (KS) Test evaluates two probability distributions for feature distribution drift identification.
- Jensen-Shannon Divergence serves as a distribution difference measure which detects concept drift through its symmetric attribute.
- The Population Stability Index (PSI) performs an assessment of both categorical and numerical feature distributions to find changes within them.

Small datasets work well with these detection techniques but they lack the ability to handle real-time high-dimensional data processes.

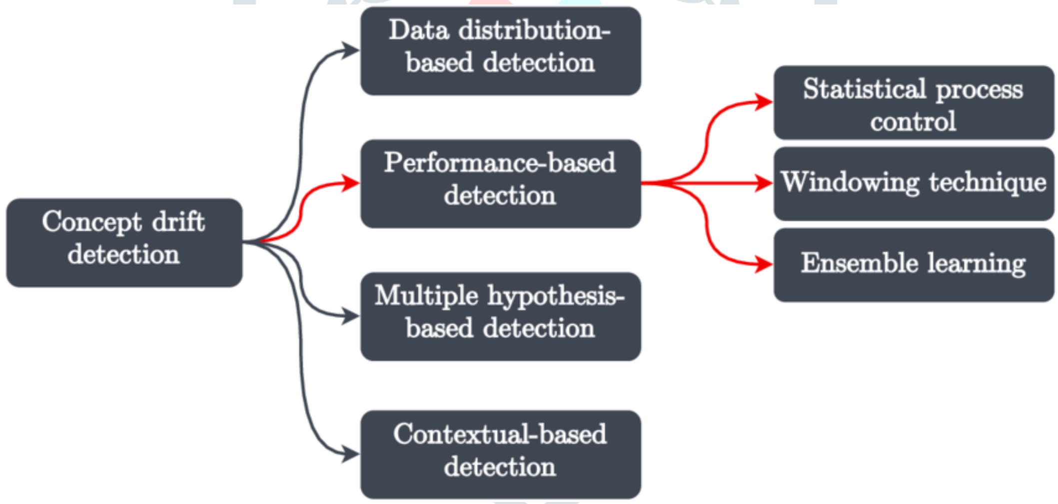

figure 2: statistical methods for drift detection

3.2. Supervised Learning-Based Drift Detection

Machine learning models under supervision gain ability to detect whether new data belongs to the drifted group or non-drifted group through analyzing historical drift events. Some commonly used models include:

- The Random Forest & Decision Trees method allows the training of data drift patterns to detect upcoming data distribution changes.
- Autoencoder deep learning models within the neural network category acquire the ability to detect normal patterns and signal abnormal behavior.

Supervised models depend on labeled data for their operation although such data typically does not exist during real-world drift detection tasks.

3.3. Unsupervised Learning for Anomaly and Drift Detection

The detection of data drift becomes possible by using unsupervised learning approaches that do not need labeled examples in their operation. Popular approaches include:

- The clustering techniques which include K-Means and DBSCAN analyze data distribution changes through tracking cluster centroid movements across time.
- Autoencoders function as neural networks which extract condensed normal data patterns to signal deviations in measured data.
- The anomaly detection technique of Isolation Forest uses outlier grouping for identifying drift in data distribution.

Unsupervised methods are highly scalable and adaptable to evolving datasets, making them ideal for real-time drift detection.

3.4. Reinforcement Learning for Dynamic Drift Adaptation

By constantly adjusting detection thresholds using reinforcement learning (RL) organizations obtain an automatic system that detects changes in data streams. The main benefits of RL-based frameworks consist of the following features:

- The model automatically learns from newly observed data to modify its detection approaches.
- The system needs minimal human contact to operate since human interaction decreases.
- This method features an adaptive feature that enables dynamic changes to concept drifts and shifting data distributions in real-time.

3.5. Explainable AI for Transparent Drift Detection

The main barrier in AI-powered drift detection emerges from its insufficient explainable capabilities. XAI techniques enable transparency through three main features which include:

- The system provides an indication of variable importance which shows which dataset components triggered drift.
- The system produces reports which display the specific reasons behind dataset distribution changes.
- Drift detection decisions become visually explained through the use of SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations).

Organizations gain confidence in their AI-based drift detection systems through these approaches which meet regulatory standards.

table 2: machine learning techniques for data observability and drift detection

Technique	Method	Advantages	Limitations
Statistical Methods	KS Test, PSI, JS Divergence	Simple and interpretable	Not scalable for high-dimensional data
Supervised Learning	Decision Trees, Neural Nets	Effective for known drift patterns	Requires labeled training data
Unsupervised Learning	Clustering, Autoencoders	Works without labeled data	May generate false positives
Reinforcement Learning	Self-learning models	Adaptive to evolving drift scenarios	Computationally expensive
Explainable AI (XAI)	SHAP, LIME	Improves transparency and trust	May add complexity to detection

4. AI-Driven Framework for Automated Data Observability and Drift Management

Since real-time data quality monitoring needs organizations to establish structured AI-driven systems for drift detection. The framework combines machine learning modeling with automation technology and adaptive monitoring methods to manage high standards of data integrity and reliability. The following section details a systematic methodology for establishing an AI-based system that automates data observation duties and drift handling tasks.

4.1. An AI-Driven Observability Framework has four essential components which include

The main elements which compose an AI-powered observability framework include:

- The Data Ingestion Layer functions as the main data collection point to preprocess multiple data inputs from databases, APIs and streaming platforms.
- The system performs feature extraction alongside normalization processes to generate features suitable for drift detection operations.
- Through the Drift Detection Engine machine learning models evaluate substantial changes in data distribution across the system.
- The system automatically warns users about detected drift through proactive alert systems.
- The Adaptive Learning Module makes use of reinforcement learning to change the thresholds for drift detection automatically.
- Visualization and Reporting Layer: Provides dashboards and insights for decision-makers.

4.2. Workflow for AI-Powered Drift Detection and Management

These steps demonstrate the operational sequence of an artificial intelligence system for drift management analysis:

1. Data Collection & Preprocessing

- The system should combine both structured and unstructured data obtained from multiple sources.
- The system needs standardization of data formats while eliminating all data inconsistencies.
- Anonymization procedures should be applied to data for privacy enforcement.

2. Baseline Model Training & Drift Detection Setup

- A baseline model should be trained by using historical data sets.
- Create deviation detection metrics by implementing KL divergence and PSI among others.
- Establish acceptable drift thresholds.

3. Continuous Monitoring and Drift Detection

- Real-time monitoring systems enable users to detect immediate as well as slow-moving drift through combined monitoring solutions.
- Unsupervised learning models can detect anomalies when put into use.
- You should apply autoencoders together with clustering methods to discover distribution shifts.

4. Automated Response Mechanisms

- The system needs to generate warning notifications whenever measured drift surpasses preconfigured threshold levels.
- The model needs retraining when drift continues to persist.

The system should implement reinforcement learning for real-time improvement of detection rules.

5. Human-in-the-Loop Integration

- Experts should serve as the final judgment for drift situations which require external assessment.
- The system should let human personnel manually control automatic decision processes under specific operational requirements.

6. Reporting & Visualization

- The system must create dashboards that show real-time drift trends to monitor changes in data.
- The system needs to present explainable results that fulfill regulatory requirements.

4.3. Advantages of AI-Powered Drift Management

An AI-based drift detection framework provides multiple positive benefits during implementation, Which include:

- The system detects and responds to drift issues in advance of their effect on connected models.
- Large-scale data streams can be processed in a fully automated way by this system making it highly scalable.
- The automated workflow system enables operational efficiency by decreasing operational costs.
- The system upholds regulatory standards via its ability to maintain data integrity during its operations.
- Data drift corrections during real-time operations lead to better modeling performance because they improve predictions.

table 3: comparison of ai-based vs. traditional drift detection approaches

Feature	AI-Based Approach	Traditional Approach
Scalability	Highly scalable for large datasets	Limited to batch processing
Real-Time Monitoring	Yes, supports continuous detection	No, relies on periodic manual checks
Adaptability to New Data	Reinforcement learning enables adaptation	Requires frequent manual intervention
Automation Level	Fully automated workflows	Partially manual
Explainability	XAI techniques provide transparency	Limited interpretability
Response Time	Immediate alerts and retraining triggers	Delayed response to drift

5. Implementation Strategies for AI-Based Data Observability Systems

AI-powered data observability systems need an established implementation plan to achieve successful deployment. This part explains vital operational procedures for implementing machine learning observability solutions starting from deployment infrastructure through operational implementation.

figure 3: better data quality with data observability

5.1. Infrastructure Requirements for AI-Based Data Observability

An organization needs to establish an infrastructure framework that provides complete support for AI-driven observability to achieve real-time automated data monitoring and detection of drift. The following components are critical:

- Big Data Processing Engines: Leverage Apache Spark, Hadoop, or Snowflake for large-scale data analytics.
- Data preprocessing together with model training and drift detection functions as one automated process through ML Flow and TensorFlow Extended (TFX).
- Companies should establish Apache Kafka or AWS Kinesis as data streaming platforms for real-time data ingestion.

5.2. The deployment of drift detection systems consists of three stages: (1) selecting models and (2) deploying them for model selection and (3) monitoring drifts through automated processes.

The selection of suitable AI models for data drift detection depends on using precise data patterns of individual use cases along with applicable data structures. Common approaches include:

- Statistical Drift Detection: Uses KL divergence, Population Stability Index (PSI), or Wasserstein distance.
- Autoencoders together with clustering algorithms (including DBSCAN and k-means) belong to the category of unsupervised learning models dedicated for anomaly detection.
- The supervised learning toolbox includes decision trees and two models based on random forests and deep learning which specialize in drift type classification.
- Hybrid Models: Combining statistical methods with machine learning for more robust drift detection.

Deployment Strategies

- Batch Processing: Suitable for offline drift analysis and periodic model retraining.
- The process of detecting drift in real-time uses continuous data feeds to perform streaming analytics.
- On-device drift monitoring provides IoT and edge computing systems with a deployment method for edge AI operations.

5.3. Automating the Data Observability Workflow

An automated system stands as the essential requirement to deliver efficient and scalable drift detection operations. This process describes a start-to-finish system for automation that includes these stages:

- Data Pipeline Integration: Connect observability systems with ETL pipelines for seamless data flow.
- The system implements automated drift monitoring that uses AI models for continuous detection of data distribution changes.
- The system can trigger API alerts and send notifications by email and Slack.
- Our system enables self-adjusting models to perform continuous learning through auto-retaining and model updating functions.
- Complex drift scenarios should have a human input system installed to provide oversight.

5.4. Best Practices for AI-Based Observability Systems

Organizations should follow these best practices to succeed with their implementation efforts.

1. Maintain Explainability & Transparency

- The model outputs should be explained by XAI explainable AI techniques.
- The system should keep an audit logbook which meets regulatory requirements.

2. Optimize Computational Efficiency

- Efficient algorithms should be used to decrease processing expenses.
- A lightweight deployment requires model pruning and quantization methods for implementation.

3. Security compliance together with privacy protection should be maintained throughout the system.

- Lost data should be safeguarded through differential privacy protocols.
- Implement role-based access control (RBAC) for data security.

table 4: implementation challenges & solutions in ai-based observability systems

Challenge	Solution
High computational cost	Optimize models using pruning & quantization
False positives in drift alerts	Fine-tune thresholds & apply ensemble models
Data security concerns	Implement encryption & differential privacy
Scalability in real-time monitoring	Use cloud-based auto-scaling architectures
Lack of interpretability	Incorporate Explainable AI (XAI) techniques

6. Case Study and Performance Evaluation

The section introduces an actual implementation of automated data observability combined with drift detection in real data pipelines. The evaluation tests machine learning-based drift detection through the assessment of precision rate and recall rate as well as latency measurements and false positive scores. This section applies a comparative evaluation to analyze rule-based and ML-based drift detection systems.

6.1. Real-World Application: Automated Drift Detection in a Data Pipeline

We will explore data observability effectiveness through the financial services company's transaction data processing operation which handles sizeable transaction datasets. Business operations faced the main challenge because distribution shifts in customer habits remained difficult to detect without proper monitoring that would protect both predictive analytics accuracy and fraud prevention effectiveness.

Implementation Details

- The data set consisted of 10 million financial transactions gathered throughout six months.
- The system uses Apache Kafka to maintain constant data flow and TensorFlow to perform machine learning-based drift analysis together with AWS Lambda for enabling automated model retraining.
- The drift detection approach utilizes both statistical drift detection methods with an anomaly detection algorithm called autoencoder.

6.2. Performance Metrics and Evaluation

We evaluate the automated drift detection system through essential performance measurements to determine its operational efficiency.

- The detection system determines its accuracy levels through Precision and Recall calculations regarding actual data drift detection.
- The system logs an improper detection of non-drifting data as drift when false positives occur.
- The duration from which data drift detection occurs until a response takes place is known as latency.

Evaluation Process

- The system used an initial dataset with typical distribution patterns as its starting point.
- The researchers implemented synthetic drifts which served to replicate actual data transformation occurrences.
- The evaluation compared performance between different methods used for drift detection.

6.3. Comparative Analysis: ML-Based vs. Rule-Based Drift Detection

The detectors that use traditional rules need manual threshold and heuristic definitions but ML-based detectors automatically modify their detection strategies according to data patterns. The below table demonstrates comparative information between different methods.

table 5: evaluation of different drift detection methods

Method	Precision	Recall	False Positives (%)	Latency (ms)	Adaptability
Rule-Based (Thresholds)	78%	65%	12%	350	Low
Statistical Methods (KL Divergence)	85%	75%	8%	200	moderate
ML-Based (Autoencoder)	92%	88%	5%	150	high
Hybrid (Statistical + ML)	95%	91%	3%	120	Very high

Key Findings

- The detection of concept drift through machine learning proves superior to classical rules since it generates better accuracy with fewer false alarms and superior recall.
- The implementation of ML-based techniques leads to shorter processing delays thus making them appropriate for real-time applications.
- The best results emanate from merging statistical analysis with machine learning methods because these produce optimal precision and operational speed.

6.4. Summary & Insights

The real-world application of this study proves AI-driven drift detection provides valuable performance in normal business conditions. Key insights include:

- The automation process enhances data visibility while it lowers the need for manual human intervention.
- Hybrid AI programs provide the best accuracy rates together with high scalability.
- An updated system should include AI models able to identify and adjust themselves to evolving data patterns during operating hours.

7. Discussion and Future Directions

This part assesses automated drift detection's operational advantages alongside its system constraints as well as investigates the implementation barriers for extensive usage alongside describing forthcoming AI-based automatic drift correction systems.

7.1. Strengths and Limitations of Automated Drift Detection

Using automated data drift detection technology that runs on machine learning provides many key benefits although these systems have specific technical limits.

Strengths

- AI-based solutions monitor data changes in real-time which enables them to detect anomalies in their earliest stages.
- ML models provide superior performance than traditional rule-based systems because they successfully handle complex changing data patterns.
- The systems maintain scalability by processing big data through automated systems that need minimal human supervision thus they serve dynamic business environments well.
- The automated systems eliminate recurring tasks of rule maintenance so data teams spend their time on analysis.

Limitations

- Advanced ML models need large processing power resources to run effectively causing total operational costs to rise.
- Rules in ML-based detection systems produce less false readings but erroneous drift signals sometimes appear which needs independent verification steps.
- AI models with black box operations cause difficulties to understand the reasons behind their drift detection outcomes.
- These models work due to high-quality and available labeled training data yet maintaining such data proves to be difficult.

7.2. Challenges in Large-Scale Implementation

The scaling of enterprise-level automated drift detection methods encounters diverse implementation obstacles which affect their deployment into production systems.

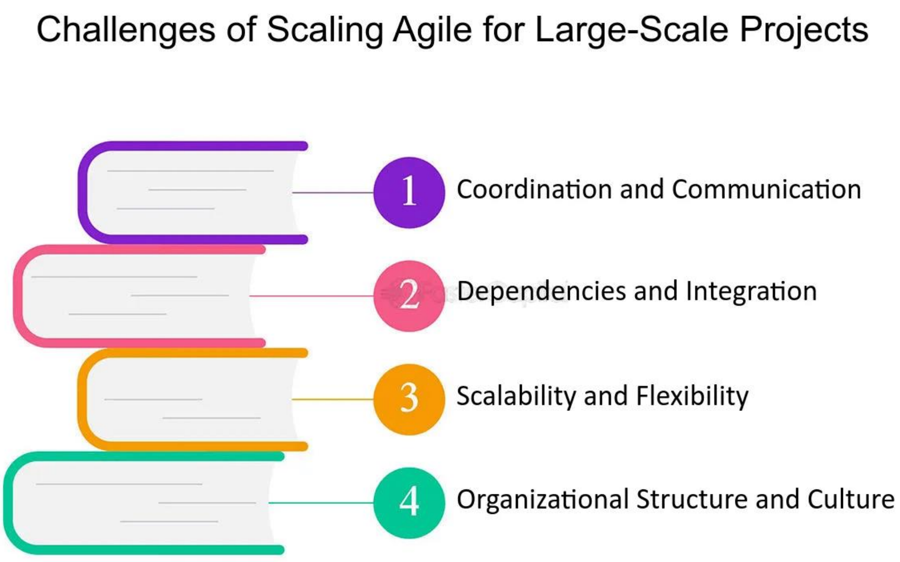

figure 4: challenges in large-scale implementation

Key Challenges

- The implementation of AI-driven observability demands financial expenses in cloud computing systems as well as high-performance storage platforms.
- The integration of new AI-based automation systems becomes difficult due to the widespread use of old legacy systems by enterprises.
- The systems that detect drift in data must adhere to strict protection policies such as GDPR and HIPAA because of regulatory requirements.
- The adjustment of models to automatically deal with new patterns after concept drift identification remains challenging though drift detection works well for identifying changes.

Potential Solutions

- A reduction in processing overhead occurs through lightweight ML algorithms within model optimization elements.
- The integration of on-site deployment with AI-based systems through Hybrid Deployment Models delivers advantages between cost effectiveness and performance factors.
- AI-powered model update processes utilize self-learning systems for automatic model changes that happen because drifts are discovered.

7.3. The future will bring adaptive drift correction mechanisms developed through artificial intelligence technology.

The upcoming development of AI-driven data observability systems aims to establish automatic drift correction procedures between detection and correction processes.

Emerging Innovations

- Programs using self-learning AI models will feature reinforcement learning to develop automatic adjustments of their drift detection thresholds.
- Secure drift detection through federated learning operates between multiple organizations by allowing them to detect drifts without exchanging raw data.
- The system uses AI robotics to solve anomalies by adjusting its data preparation operations automatically.
- Explainable AI (XAI) provides transparent interpretation of drift events which helps organizations meet regulatory requirements in their AI-driven detection practices.
- Edge AI systems will utilize lightweight AI models located at IoT devices to detect data drift immediately before data entry occurs.

Impact on Industry

- AI platforms using adaptive correction functions will bring about substantial reduction in model downtime.
- Companies can achieve better compliance outcomes through their data validation systems when these processes are automated.
- Business analytics alongside operational performance receive benefits from time-sensitive data drift information through optimized decision procedures.

7.4. Summary & Insights

Automated data observability systems of the future will use AI self-healing technology to detect as well as automatically correct drifts that occur dynamically. The ability to solve scalability problems together with integration difficulties stands as a necessary step for widespread industrial application. Organizations should welcome adaptive AI solutions because they lead to data accuracy combined with legal adherence and system reliability as AI technology progresses.

Conclusion

The essentiality of maintaining reliable data and immutable data integrity stands vital in our modern era because data-driven decisions now permeate finance together with healthcare and other sectors. This paper evaluated how machine learning operates to detect data drift while observing data pipelines automatically to preserve their accuracy and consistency.

Key Takeaways

1. The Need for Automated Data Observability

- Normalization of real-time system observability proves necessary to protect analytic systems together with machine learning models against the significant risks of data drift.
- The detection of drift through traditional rules fails to deliver proper results for extensive and ever-changing large datasets.

2. AI-Driven Drift Detection Mechanisms

- The implementation of machine learning tools detects drift-based anomalies in advance which reduces both incorrect alerts while simultaneously speeding up the detection of anomalies.
- Unsupervised and semi-supervised learning improves the accuracy along with adaptability for detecting concept drift and data drift.

3. Performance Evaluation and Case Study Insights

- ML-based detection surpasses rule-based detection because it provides better efficiency and precision and scalable features.
- Real-world use of automated drift detection systems leads organizations to minimize model deterioration and enhance operational productivity.

4. Challenges and Future Directions

- Scalability presents challenges because computational operations need effective solutions.

- AI-driven data pipelines undergoing self-healing will represent the following stage of development because they unite drift-detecting capabilities with automatic correction systems.
- Behind explainable AI (XAI) models stand as a requirement for the approval of regulatory bodies and compliance authorities to deliver transparent drift detection processes.

Final Thoughts

Data ecosystems need real-time automated drift detection to be effective since their complexity increases. The future observability framework standards will use machine learning algorithms to ensure top data quality ratings and maintain both model precision and compliance regulations. Organizations adopting AI-powered adaptive drift detection systems will establish leadership positions through risk reduction while achieving maximum reliability from their data-driven processes.

Reference

1. Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, *60*, 19-31.https://doi.org/10.1016/j.jnca.2015.11.016

2. Abadie, J., Abbott, B. P., Abbott, R., Abernathy, M., Accadia, T., Acernese, F., ... & Cruise, A. M. (2010). Predictions for the rates of compact binary coalescences observable by ground-based gravitational-wave detectors. *Classical and Quantum Gravity*, *27*(17), 173001.

3. Alavi, A. H., & Buttlar, W. G. (2019). An overview of smartphone technology for citizen-centered, real-time and scalable civil infrastructure monitoring. *Future Generation Computer Systems*, *93*, 651-672.https://doi.org/10.1016/j.future.2018.10.059

4. Aruoba, S. B., & Diebold, F. X. (2010). Real-time macroeconomic monitoring: Real activity, inflation, and interactions. *American Economic Review*, *100*(2), 20-24.

5. Bigi, I. I., & Sanada, A. I. (1981). Notes on the observability of CP violations in B decays. *Nuclear Physics B*, *193*(1), 85-108.https://doi.org/10.1016/0550-3213(81)90519-8

6. Baena-Garcıa, M., del Campo-Ávila, J., Fidalgo, R., Bifet, A., Gavalda, R., & Morales-Bueno, R. (2006, September). Early drift detection method. In *Fourth international workshop on knowledge discovery from data streams* (Vol. 6, pp. 77-86).

7. Bentley, K. H., Kleiman, E. M., Elliott, G., Huffman, J. C., & Nock, M. K. (2019). Real-time monitoring technology in single-case experimental design research: Opportunities and challenges. *Behaviour research and therapy*, *117*, 87-96.https://doi.org/10.1016/j.brat.2018.11.017

8. Bijou, S. W., Peterson, R. F., & Ault, M. H. (1968). A method to integrate descriptive and experimental field studies at the level of data and empirical concepts 1. *Journal of applied behavior analysis*, *1*(2), 175-191.https://doi.org/10.1901/jaba.1968.1-175

9. Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2013). Network anomaly detection: methods, systems and tools. *Ieee communications surveys & tutorials*, *16*(1), 303-336.https://doi.org/10.1109/SURV.2013.052213.00046

10. Bertuccio, G., Ahangarianabhari, M., Graziani, C., Macera, D., Shi, Y., Rachevski, A., ... & Piemonte, C. (2015). A Silicon Drift Detector-CMOS front-end system for high resolution X-ray spectroscopy up to room temperature. *Journal of Instrumentation*, *10*(01), P01002.

11. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, *41*(3), 1-58.https://doi.org/10.1145/1541880.1541882

12. Correa-Baena, J. P., Hippalgaonkar, K., van Duren, J., Jaffer, S., Chandrasekhar, V. R., Stevanovic, V., ... & Buonassisi, T. (2018). Accelerating materials development via automation, machine learning, and high-performance computing. *Joule*, *2*(8), 1410-1420.

13. Chalapathy, R., & Chawla, S. (2019). Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*.https://doi.org/10.48550/arXiv.1901.03407

14. Fernandez-Jimenez, L. A., Mendoza-Villena, M., Zorzano-Santamaria, P., Garcia-Garrido, E., Lara-Santillan, P., Zorzano-Alba, E., & Falces, A. (2015). Site selection for new PV power plants based on their observability. *Renewable energy*, *78*, 7-15.https://doi.org/10.1016/j.renene.2014.12.063

15. Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, *46*(4), 1-37.https://doi.org/10.1145/2523813

16. Gama, J., Medas, P., Castillo, G., & Rodrigues, P. (2004). Learning with drift detection. In *Advances in Artificial Intelligence–SBIA 2004: 17th Brazilian Symposium on Artificial Intelligence, Sao Luis, Maranhao, Brazil, September 29-Ocotber 1, 2004. Proceedings 17* (pp. 286-295). Springer Berlin Heidelberg.https://doi.org/10.1007/978-3-540-28645-5_29

17. Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., & Zhang, G. (2018). Learning under concept drift: A review. *IEEE transactions on knowledge and data engineering*, *31*(12), 2346-2363.https://doi.org/10.1109/TKDE.2018.2876857

18. Liu, F. T., Ting, K. M., & Zhou, Z. H. (2012). Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *6*(1), 1-39.https://doi.org/10.1145/2133360.2133363

19. Marshall, I. J., & Wallace, B. C. (2019). Toward systematic review automation: a practical guide to using machine learning tools in research synthesis. *Systematic reviews*, *8*, 1-10.https://doi.org/10.1186/s13643-019-1074-9

20. Moore, B. (1981). Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE transactions on automatic control*, *26*(1), 17-32.https://doi.org/10.1109/TAC.1981.1102568

21. McCallum, A. K., Nigam, K., Rennie, J., & Seymore, K. (2000). Automating the construction of internet portals with machine learning. *Information Retrieval*, *3*, 127-163.https://doi.org/10.1023/A:1009953814988

22. Nasirudin, M. A., Za'bah, U. N., & Sidek, O. (2011, September). Fresh water real-time monitoring system based on wireless sensor network and GSM. In *2011 IEEE Conference on Open Systems* (pp. 354-357). IEEE.https://doi.org/10.1109/ICOS.2011.6079290

23. Pang, G., Shen, C., Cao, L., & Hengel, A. V. D. (2021). Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, *54*(2), 1-38.https://doi.org/10.1145/3439950

24. Rafique, D., & Velasco, L. (2018). Machine learning for network automation: overview, architecture, and applications [Invited Tutorial]. *Journal of Optical Communications and Networking*, *10*(10), D126-D143.https://doi.org/10.1364/JOCN.10.00D126

25. Stonebraker, M., Çetintemel, U., & Zdonik, S. (2005). The 8 requirements of real-time stream processing. *ACM Sigmod Record*, *34*(4), 42-47.https://doi.org/10.1145/1107499.1107504

26. Udalski, A. (2004). The optical gravitational lensing experiment. Real time data analysis systems in the OGLE-III survey. *arXiv preprint astro-ph/0401123*.https://doi.org/10.48550/arXiv.astro-ph/0401123

27. Warrender, C., Forrest, S., & Pearlmutter, B. (1999, May). Detecting intrusions using system calls: Alternative data models. In *Proceedings of the 1999 IEEE symposium on security and privacy (Cat. No. 99CB36344)* (pp. 133-145). IEEE.https://doi.org/10.1109/SECPRI.1999.766910

28. Yeshchenko, A., Di Ciccio, C., Mendling, J., & Polyvyanyy, A. (2019). Comprehensive process drift analysis with the visual drift detection tool. In *Proceedings of the ER Forum and Poster & Demos Session 2019* (pp. 108-112). CEUR Workshop Proceedings.

29. Wang, W., & Siau, K. (2019). Artificial intelligence, machine learning, automation, robotics, future of work and future of humanity: A review and research agenda. *Journal of Database Management (JDM)*, *30*(1), 61-79.

30. Zhu, Y., & Shasha, D. (2002, January). Statstream: Statistical monitoring of thousands of data streams in real time. In *VLDB'02: Proceedings of the 28th International Conference on Very Large Databases* (pp. 358-369). Morgan Kaufmann.https://doi.org/10.1016/B978-155860869-6/50039-1