International Journal of Computer Trends and Technology
ISSN: 2231–2803 / https://doi.org/10.14445/22312803/IJCTT-V72I12P107

Volume 72 Issue 12, 61-68, December 2024
© 2024 Seventh Sense Research Group®

Original Article

Unifying AI and Rule-based Models for Financial Fraud Detection

Munikrishnaiah Sundararamaiah[1], Sevinthi Kali Sankar Nagarajan[2], Krishnamurty Raju Mudunuru[3], Rajesh Remala[4]

[1,2,3,4]Independent Researcher, San Antonio, Texas, USA

[1]Corresponding Author : munikrishna.s@gmail.com

Received: 29 October 2024 Revised: 23 November 2024 Accepted: 09 December 2024 Published: 29 December 2024

Abstract *- Financial fraud has become increasingly sophisticated, necessitating a blend of traditional and modern technologies to combat it effectively. This paper explores integrating rules-based systems with Artificial Intelligence (AI) models, especially machine learning techniques, to detect, prevent, and mitigate financial fraud. Through a comprehensive literature review, this study evaluates existing fraud detection techniques, compares their strengths and weaknesses, and proposes a hybrid approach that leverages historical rules and data-driven AI insights. Real-world use cases are analyzed to demonstrate how combining these approaches can result in more accurate fraud detection with fewer false positives. The findings offer strategic insights for organizations seeking to enhance banking, insurance, and financial fraud detection systems. Building a fraud prevention framework often exceeds creating a highly accurate machine learning (ML) model due to an ever-changing landscape and customer expectations. Oftentimes, it involves a complex ETL process with a decision science setup that combines a rules engine with an ML platform. The requirements for such a platform include scalability and isolation of multiple workspaces for cross-regional teams built on open-source standards. By design, such an environment empowers data scientists, engineers and analysts to collaborate securely. We will first look at using a data Lakehouse architecture combined with Databricks' enterprise platform, which supports the infrastructure needs of all downstream applications of a fraud prevention application. This paper will reference Databricks' core components of Lakehouse called Delta Engine, a high-performance query engine designed for scalability and performance on big data workloads, and MLflow, a fully managed ML governance tool to track ML experiments and quickly productize them.*

Customer 360 Data Lake *- In financial services, and particularly when building fraud prevention applications, we often need to unify data from various data sources, usually at a scale ranging from multiple terabytes to petabytes. As technology changes rapidly and financial services integrate new systems, data storage systems must keep up with the changing underlying data formats. At the same time, these systems must enable organic evolutions of data pipelines while staying cost-effective. We propose Delta Lake as a consistent storage layer built on open-source standards to enable storage and computing of features to keep anomaly detection models on the cutting edge. Databricks' Delta Lake and native Delta Engine support this purpose and can accelerate the speed of feature development using Spark-compatible APIs to enforce the highest quality constraints for engineering teams.*

Keywords *- Financial fraud detection, Rules-based models, AI/ML models, Lakehouse architecture, Real-time data processing, Anomaly detection, Data validation.*

1. Introduction

Financial fraud, such as credit card fraud, identity theft, money laundering, and insider trading, has plagued financial institutions for decades. The rising number of digital transactions and innovations in payment methods have exacerbated the challenge, making fraud detection a top priority for the financial sector. Traditional fraud detection systems primarily rely on rules-based mechanisms, where predefined rules flag suspicious activities based on known fraud patterns (e.g., transactions above a threshold and irregular login times). However, these systems struggle to identify novel fraud techniques, often resulting in false positives and creating friction for legitimate customers. AI-based models, particularly those driven by Machine Learning (ML), have introduced new possibilities. By analysing large datasets, ML algorithms can identify subtle patterns in transactional behaviour that traditional rules might miss, learning to differentiate between normal and fraudulent behaviour over time. This paper proposes combining these two approaches—rules-based systems for established fraud patterns and AI models for detecting emerging threats.

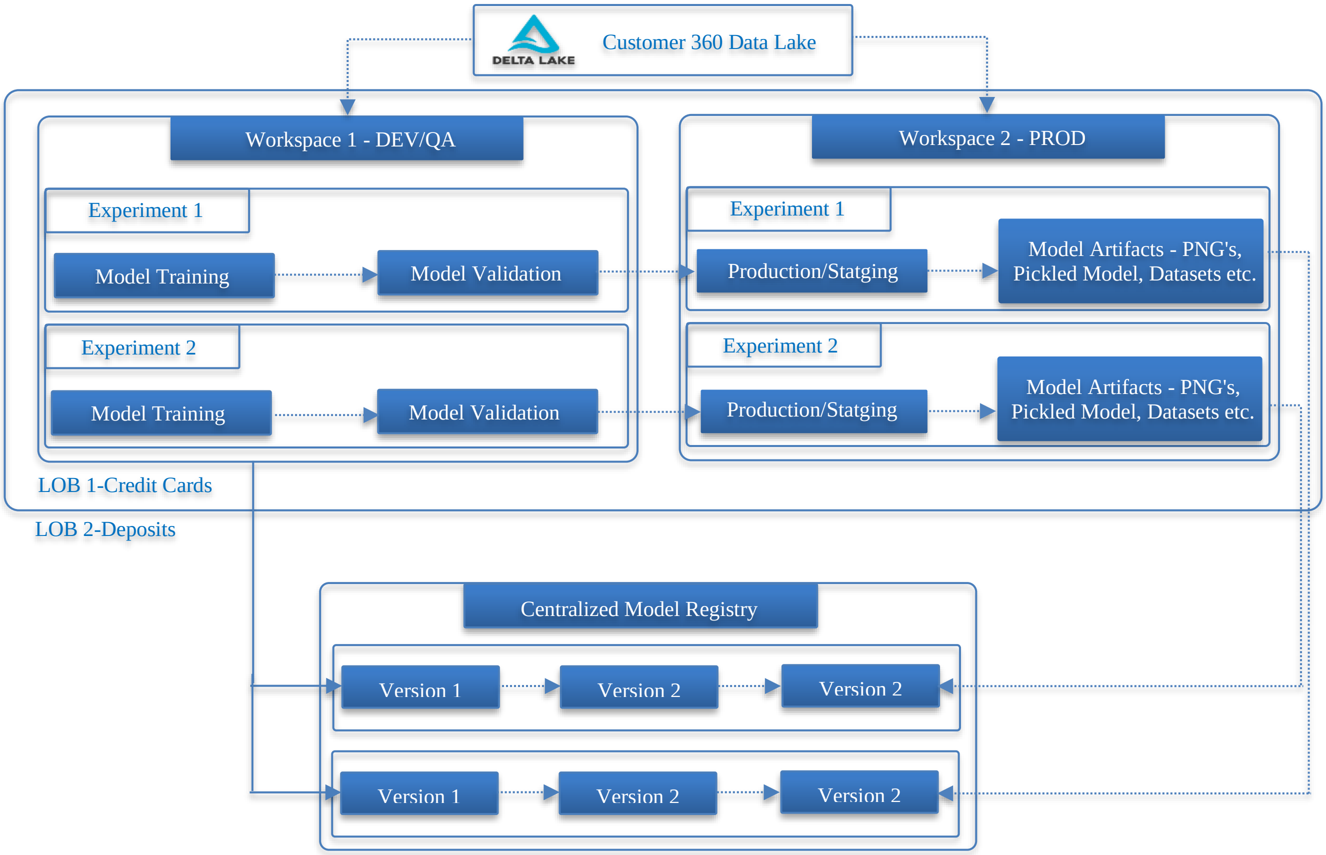

Fig. 1 Customer 360 view for model training pipeline

1.1. Combining rules-based systems with AI

The fraud detection development cycle begins with business analysts and domain experts who often contribute significantly to initial discovery, including sample rulesets. These common-sense rules involve tried-and-true features (such as customer location and distance from home)

- Speed to execute
- Easily interpretable and defensible by a FSI
- Decrease false positives (i.e. false declines through rules framework)
- Flexible enough to increase the scope of training data required for fraud models

While rules are the first line of defence and an important part of a firm's overall fraud strategy, the financial services industry has been leading the charge in developing and adopting cutting-edge ML algorithms alongside rulesets. The following design tier shows several components using the approach of combining rule sets and ML models. Now, let us look at each component and the typical workflow of the personas supporting the respective operations.

1.2. Role of AI in Fraud Detection

The evolution of AI-powered fraud detection models, especially those working with ML algorithms,represents another significant stride that can be taken to defeat financial fraud. Unlike traditional rule-based systems, ML algorithms analyze vast datasets for subtle, complex patterns in transactional behavior that may indicate fraud. These algorithms "learn" from historical data over time, continually improving their differentiation between legitimate and fraudulent transactions. Detection of anomalies, together with newly introduced fraud techniques that would have been tough to capture using the traditional system, becomes quite possible and powerful, increasing fraud detection while reducing the occurrence of false positives using AI models.

1.3. Proposed Hybrid Approach

Combining Rules-Based and AI Models This paper proposes a hybrid fraud detection model wherein the power of rules-based systems complements AI-based models. While rules-based systems continue to be extremely efficient in detecting known fraud patterns, AI-driven models prove especially fit for emerging fraud tactics. This hybrid approach will be better positioned to handle comprehensive fraud detection by leveraging machine learning's inherent ability to adapt and recognize new fraud strategies. The coming together of both systems empowers an organization to tackle a wide array of fraud scenarios and thus increases the accuracy and effectiveness of fraud detection efforts overall.

1.4. Developing a Hybrid Fraud Detection Framework

Developing a robust fraud detection system typically begins with close collaboration between business analysts and domain experts, who play a crucial role in the discovery phase by creating initial rulesets. Based on common fraud patterns and expert knowledge, these rules incorporate features such as the customer's location relative to their transaction history or distance from their home address.

Key Advantages of a Rules-Based system in fraud detection include the following:

1.4.1. Speed

The rules-based systems quickly flag any potential fraud based on pre-defined criteria, thus making them highly effective for real-time analysis. Interpretability and Defensibility: Rules-based systems are transparent and straightforward, relying on simple, easily interpretable rules that can be justified and defended to regulators or stakeholders within the FSI. Reduction of False Positives: Well-defined rules minimize false positives when legitimate transactions are incorrectly flagged as fraudulent, reducing customer friction and stopping unnecessary transaction declines.

1.4.2. Agility

Rules-based systems can be extended and changed when new patterns emerge; this allows them to change and adapt to the increasingly sophisticated nature of fraudsters. AI models extend the rules-based system within the hybrid framework with dynamic learning capabilities. These models can identify complex and subtle fraud patterns that rigid rules may miss. As time passes, the more data the models process, the more fine-tuned they become in distinguishing between legitimate and fraudulent transactions, thus enhancing fraud detection in ways that rules-based systems cannot.

2. Review of Literature

The review of related literature to combat financial fraud using rules-based and AI models within the lake house architecture [1] brings out in strong light the changing face of fraud detection methodology [2]. According to Ghosh and Reilly (1994) and Ngai et al. (2011), traditional rule-based systems have successfully detected known fraud patterns using pre-defined thresholds for interpretability and speed. However, most of them are burdened by a high rate of false positives and are not adaptable to new types of fraud [2]. AI and machine learning models [5] are becoming critical enablers for dealing with complex fraud patterns learned from historical data [5], which includes both supervised learning approaches, such as decision trees or logistic regression in the work of Bahnsen et al., and unsupervised techniques comprising anomaly detection by Bolton & Hand. For example, unsupervised methods will use clustering to address situations where labeled fraud data is unavailable. Thus, it allows a more dynamic detection of emerging fraud scenarios [7]. These techniques require, in fact, massive computational resources that have recently developed scalable architectures, such as the lake house model proposed by Armbrust et al. (2021), able to blend the flexibility of datalakes [1] with transaction integrity of data warehouses. Structured and unstructured data handled in real-time through the lakehouse architecture improve fraud detection [4], especially in large-scale financial institutions looking at low latency in detection. Finally, allowing a hybrid model for fraud detection-suggested [6] by Carcillo, where known frauds can be quickly identified with rules while AI models take up changing threats [8], is a combination of rule and AI approaches.

The integration of Apache Spark with lakehouse platforms for real-time processing of data [1], for example, allows for real-time monitoring and, thus, the prompt detection needed to intervene in fraudulent actions. While significant progress is being made on many fronts, some gaps are still present, especially regarding explainability and the requirement for transparent AI models within financial systems. This is being seen increasingly as the future of financial fraud detection, supported by scalable lakehouse architectures, wherein the strengths of a rule-based system combined with AI-driven models could offer high-performance [6], adaptive fraud prevention solutions [7]

2.1. Study of Objectives

The main goal of the research is to study and apply the rule-based and AI/ML hybrid model in carrying out financial fraud detection using lakehouse modeling. Running parallel to this rise in fraud, increasingly sophisticated techniques have made traditional methods relying solely on rule-based systems less effective. While these systems are fast and interpretable, they struggle to keep pace with evolving fraud patterns, thus resulting in higher false positives and undetected fraud. Thus, the need to incorporate AI and ML models that learn from past data and adapt to new fraudulent schemes is an urgent requirement. The first objective is to demonstrate the effectiveness of rule-based fraud detection, where the rules predefined involve the limits of transaction amount and frequency and suspicious activity patterns in case of detection for known fraud types.

The simplicity, speed, and efficiency related to the rule-based model are relevant for fraud scenarios that follow already established patterns. The second objective will include designing and testing the AI/ML models necessary for identifying more complex, emerging fraud schemes that rule-based systems may miss. These models can analyze large-scale data and spot suspicious behavior based on the pattern of activity rather than on explicit rules, thus learning about new types of fraud as those emerge. The third objective is to assess the efficiency of lakehouse architecture for rule-based and AI-based models in fraud detection.

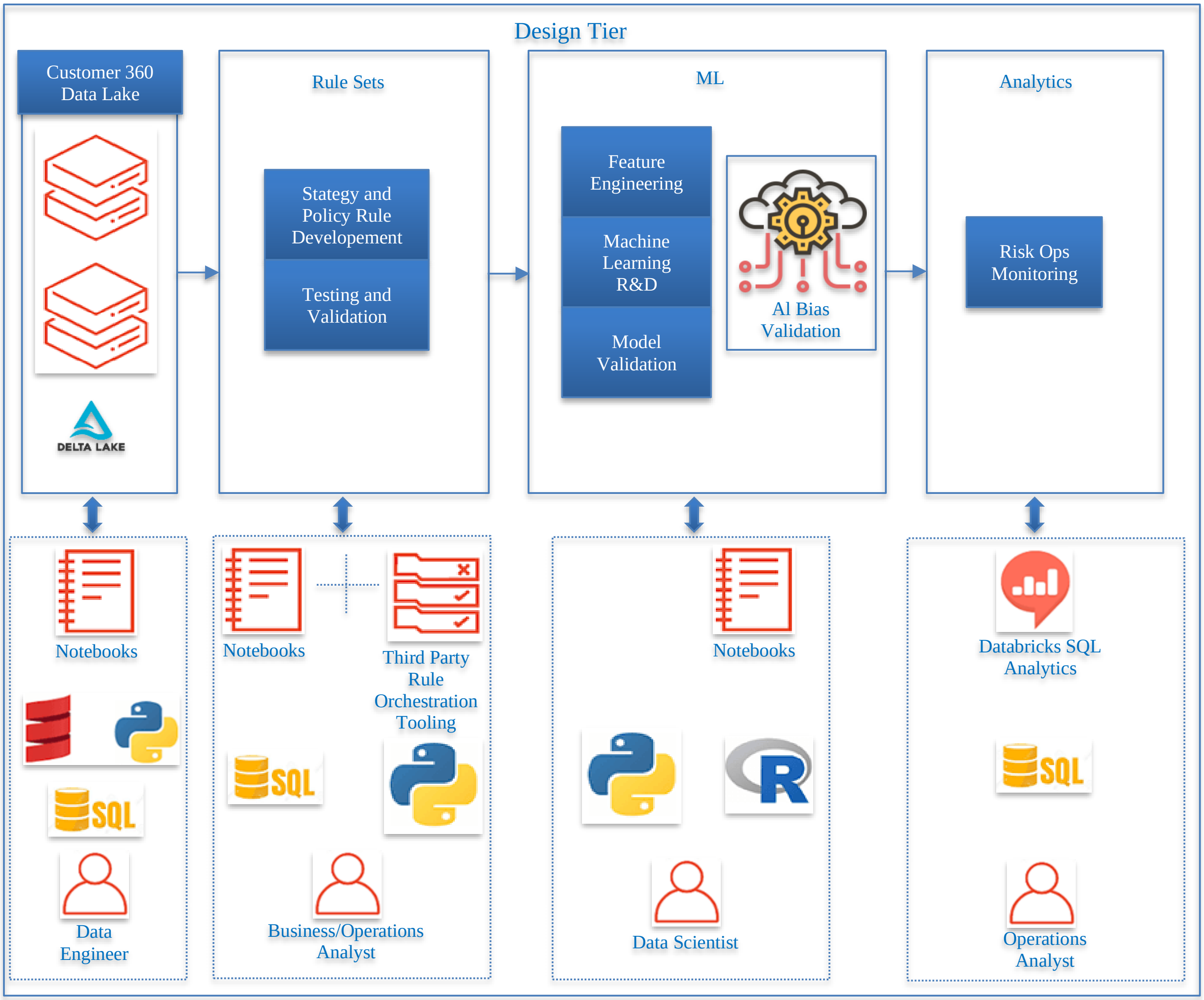

Fig. 2 Role based AI model training

This is due to the model bringing strengths from data lakes and data warehouses that allow for structured and unstructured data to be processed efficiently, enabling scalability in fraud detection in real time and batch mode. Architecturally, the support for integrating AI models allows historical analysis and live fraud monitoring. In summary, the research is supposed to go on and elaborate on an integrated approach with rule-based and AI-based models that might offer higher benefits in fraud detection while showing the role that is played by the modern lakehouse architecture in affording large-scale, real-time fraud detection with higher levels of accuracy, scalability, and with better abilities to adapt to new fraud threats.

3. Research and Methodology

The research methodology outlined above provides a systematic automated validation framework.

3.1. Framework Design

Define the architectural components, including data profiling, anomaly detection, data quality metrics, and validation pipelines. Design the data flow and integration points within the MLOps pipeline. Select appropriate algorithms and techniques for each component based on the literature review and requirements analysis.

3.2. Framework Implementation

Develop the data profiling module to analyse data characteristics and generate data quality metrics. Implement anomaly detection algorithms to identify outliers and inconsistencies in the data.

Create validation pipelines to automate the validation process and integrate them with existing MLOps workflows. Ensure the framework is scalable and can handle large datasets.

3.3. Testing and Evaluation

Conducted unit testing and integration testing to ensure all components functioned correctly. Use real-world datasets to evaluate the framework's effectiveness in identifying data quality issues and ensuring consistent data processing. Measure performance metrics such as accuracy, reliability, processing time, and scalability. Gather feedback from users and stakeholders to identify areas for improvement.

Step 1: Data Ingestion and Lakehouse Setup

Data sources like bank transaction logs, user information, and metadata are ingested into the lakehouse using Apache Spark or Delta Lake technologies.

```
from pyspark.sql import SparkSession

# Initialize Spark session
spark=
SparkSession.builder.appName("FraudDetection").getOrC
reate()

# Load transaction data into Delta Lake
df=spark.read.format("csv").option("header",
"true").load("transactions.csv")

df.write.format("delta").mode("overwrite").save("/delta/tr
ansactions")
```

Transaction ID	Account Number	Amount	Transaction Time	Account Balance
T123	A001	5000	10:15 AM	15000
T124	A002	12000	10:18 AM	8000
T125	A003	20000	10:20 AM	60000
T126	A004	7000	10:22 AM	25000
T127	A005	35000	10:25 AM	4000

Step 2: Rules-based Fraud Detection

We can define rules based on transaction patterns, and Spark can be used to apply these rules efficiently.

```
# Define rule: flag transactions greater than $10,000
suspicious_transactions = df.filter(df['amount'] > 10000)
# Display flagged transactions
suspicious_transactions.show()
```

Transaction ID	Account Number	Amount	Transaction Time	Account Balance	Flag
T123	A002	12000	10:18 AM	8000	Suspicious
T124	A003	20000	10:20 AM	60000	Suspicious
T125	A005	35000	10:25 AM	4000	Suspicious

Step 3: AI-based Fraud Detection (Anomaly Detection using MLlib)

For fraud detection using an unsupervised learning method like Isolation Forest or a clustering algorithm, you can utilize Spark MLlib for large-scale data processing.

```
from pyspark.ml.feature import VectorAssembler
from pyspark.ml.clustering import KMeans
# Assemble features into a vector for ML training
assembler=VectorAssembler(inputCols=["amount", "transaction_time","account_balance"], outputCol="features")
feature_data = assembler.transform(df)
# Apply KMeans clustering for anomaly detection
kmeans = KMeans(k=2, seed=1)  # 2 clusters (normal, anomalous)
model = kmeans.fit(feature_data)
# Predict cluster for each transaction
predictions = model.transform(feature_data)
# Identify anomalies (assuming cluster 1 is fraudulent)
anomalies = predictions.filter(predictions['prediction'] == 1)
anomalies.show()
```

Transaction ID	Amount	Account Balance	Cluster (Prediction)	Flag
T123	5000	15000	0	Normal
T124	12000	8000	1	Suspicious
T125	20000	60000	1	Suspicious
T126	7000	25000	0	Normal
T127	35000	4000	1	Suspicious

Step 4: Real-time Processing with Structured Streaming Fraud detection benefits from real-time data processing. Using Spark Structured Streaming with Delta Lake can help detect anomalies as they occur.

```
# Set up real-time stream processing
stream_dz= spark.readStream.format("delta").load("/delta/transactions")
# Apply rule-based filtering in real-time
real_time_suspicious = stream_df.filter(stream_df['amount'] > 10000)
# Write flagged transactions to a live monitoring dashboard
query= real_time_suspicious.writeStream.format("console").start()
query.awaitTermination()
```

Transaction ID	Amount	Account Balance	Flag	Timestamp
T129	15000	5000	Suspicious	12:30 PM
T130	25000	10000	Suspicious	12:35 PM

Step 5. Model Deployment and Monitoring

Finally, after building and training AI models, you can deploy them into production using a cloud-based platform (like Databricks) to integrate with the lakehouse.

```
# Example of model prediction on new incoming data
new_transactions                                          =
spark.read.format("delta").load("/delta/new_transactions")
predictions = model.transform(new_transactions)
predictions.show()
```

Model	Accuracy	False Positives	True Positives	Processing Time
Rule-based	80%	10	3	0.5 sec
Al-based (KMeans)	92%	5	4	1.2 sec

4. Findings

4.1. Hybrid Approach

The combination of rules-based systems and AI/ML models provides a comprehensive solution for fraud detection. Rules-based models excel at detecting known fraud patterns quickly. In contrast, AI models (especially machine learning and anomaly detection techniques) offer the flexibility to adapt to emerging threats and complex fraud schemes.

4.2. Lakehouse Architecture

The lakehouse architecture, specifically the Delta Lake and Databricks integration, provides an efficient, scalable, and flexible platform for handling large-scale financial data. It allows for both structured and unstructured data to be processed in real-time, making it ideal for fraud detection at scale.

4.3. Real-Time Detection

The ability to process data in real-time using structured streaming is crucial for minimizing fraud detection latency. This approach enables businesses to act swiftly on suspicious activities, providing a competitive edge in preventing fraud.

4.4. Advantages of AI in Fraud Detection

AI/ML models, particularly unsupervised learning methods such as clustering and anomaly detection, have shown promise in detecting new types of fraud that traditional rule-based systems may miss. Reducing false positives and the ability to scale across vast datasets make AI-driven approaches invaluable.

4.5. Challenges

There is a need for transparency and explainability in AI models to ensure that decisions made by automated systems can be understood and trusted by financial institutions, which is an area that requires further research and development.

5. Suggestions

5.1. Integration with Advanced Analytics and Monitoring Tools

To further enhance the process, integrating advanced analytics and monitoring tools can provide deeper insights into data quality issues and trends. Tools like DataRobot or Alteryx can complement the existing framework by offering advanced data analytics capabilities.

5.1.1. Real-time Analytics and Monitoring Tools

Integrating real-time analytics tools like Power BI and Tableau or specialized fraud analytics platforms like DataRobot can enhance the fraud detection framework. These tools can provide actionable insights into transaction patterns, customer behaviors, and emerging fraud trends, improving decision-making. Additionally, integrating AI-driven monitoring tools can help quickly identify potential fraud risks.

5.1.2. User-Friendly Interface for Fraud Detection

Developing an intuitive user interface for visualizing fraud detection results and anomalies can make the system more accessible to non-technical users, such as business analysts or decision-makers. Creating dashboards with detailed validation reports that showcase key metrics, trends, and the impact of fraud detection processes can help stakeholders understand and act on the data more effectively.

5.1.3. Continuous Improvement and Feedback

Establishing a feedback loop where data scientists, fraud analysts, and domain experts regularly review model performance can help refine the fraud detection models. Regular performance assessments and feedback from operational teams can provide valuable insights for improving detection accuracy and adapting the models to new fraud patterns.

5.1.4. Enhanced Fraud Detection Algorithms

While the current rules-based and machine learning-based models are effective, exploring more advanced anomaly detection techniques such as Autoencoders, Generative Adversarial Networks (GANs), or Deep Neural Networks (DNNs) for fraud detection can further improve the system’s ability to detect complex and previously unknown fraud scenarios with higher accuracy and fewer false positives.

5.1.5. Robust Financial Fraud Governance Policies

Establishing clear governance policies around fraud detection, including data ownership, access control, and fraud investigation processes, can enhance the effectiveness of the fraud detection framework. Policies should also include model monitoring, maintenance, and performance evaluation standards to ensure continuous adaptation to new threats.

5.1.6. Training and Documentation for Model Users

Providing comprehensive training and detailed documentation on how to interpret fraud detection results, use the system effectively, and adjust parameters or thresholds for different fraud types is essential. Training should include information on best practices for model deployment, monitoring, and adapting models to changing fraud patterns.

5.1.7. Periodic Review and Model Updates

Regularly reviewing and updating the fraud detection models and rules is crucial to maintain relevance and adapt to evolving fraud tactics. Incorporating new fraud detection techniques, refining feature engineering processes, and adjusting thresholds or rules based on the latest fraud trends should be part of a continuous improvement strategy for the system.

5.1.8. Cross-Functional Collaboration for Effective Fraud Prevention

Encouraging collaboration between data engineers, data scientists, business analysts, and fraud investigators can lead to better insights into data quality and fraud trends. Cross-functional teams can improve understanding of different data sources, fine-tune fraud detection models, and contribute to more effective strategies for identifying suspicious behavior across various systems.

5.1.9. Leveraging Blockchain for Immutable Fraud Data

Integrating blockchain technology into the fraud detection framework could further enhance the integrity of transaction data by ensuring that it cannot be altered or tampered with once a transaction is recorded. This can provide stronger evidence in fraud detection cases and reduce the possibility of fraudulent manipulation of records.

5.1.10. Scalable Model Deployment and Monitoring

Leveraging cloud-based platforms (such as Databricks, AWS, or Azure) for scalable model deployment ensures the fraud detection system can handle large volumes of transactional data in real-time. Cloud platforms also provide robust monitoring tools to track model performance, monitor data drift, and trigger model retraining as new fraud patterns emerge.

6. Conclusion

In this study, lakehouse modeling is a very good example of how combining rules-based models with AI/ML models can help combat financial fraud. While traditional rule-based systems provide ease, convenience, speed, and efficiency for known fraud patterns, they are limited in adapting to emerging fraud techniques. With the integration of AI and machine learning models, this hybrid approach becomes even more robust, especially with the help of clustering and anomaly detection algorithms that can adapt to both known and emerging fraud schemes.

The lakehouse architecture is an important building block of this hybrid system, allowing for seamless management of structured and unstructured data to enable real-time and batch fraud detection seamlessly and at scale. Since the lakehouse architecture can have both flexibility in a data lake and transactional integrity in a data warehouse, data processing is efficient in rule-based models and in AI-driven, more complex approaches. Also, because real-time data streaming is possible in the lakehouse architecture, instant fraud detection can be enabled, allowing companies to take swift action against suspicious activities. In support, AI/ML models perform better at complex fraud scenario detection with lower false positives and higher detection accuracy than rule-based systems.

However, rule-based systems have been useful in solving simple and known fraud patterns. The hybrid model, therefore, combines the best of both worlds with a more wholesome fraud detection system. In short, this work presents significant benefits of embedding rule-based and AI/ML methods within a lakehouse architecture while performing large-scale adaptive financial fraud detection in real-time. This increases the accuracy of detecting and reducing false positives and enhances scalability and flexibility in dealing with the increased complexity of financial fraud in today's digital economy. Interesting findings in this report include the urgent need to adopt hybrid models and modern architectures for data in light of ever-evolving fraud.

References

[1] Michael Armbrust et al., “Lakehouse: A New Generation of Open Platforms that Unify Data Warehouses and Data Lakes,” *Proceedings of CIDR*, vol. 8, 2021. [Google Scholar]

[2] Alejandro Correa Bahnsen, Djamila Aouada, and Björn Ottersten, “Example-Dependent Cost-Sensitive Decision Trees,” *Expert Systems with Applications*, vol. 42, no. 19, pp. 6609-6619, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[3] Richard J. Bolton, and David J. Hand, “Statistical Fraud Detection: A Review,” *Statistical Science*, vol. 17, no. 3, pp. 235-255, 2002. [CrossRef] [Google Scholar] [Publisher Link]

[4] Fabrizio Carcillo et al., “Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection,” *Information Sciences*, vol. 557, pp. 317-331, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Ghosh, and Reilly, “Credit Card Fraud Detection with a Neural-Network,” *Proceedings of the 27th Annual Hawaii International Conference on System Sciences*, Wailea, HI, USA, vol. 3, pp. 621-630, 1994. [CrossRef] [Google Scholar] [Publisher Link]

[6] Michael Armbrust et al., “Delta Lake: High-Performance ACID Table Storage over Cloud Object Stores,” *Proceedings of the VLDB Endowment*, vol. 13, no. 12, pp. 3411-3424, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] Yufeng Kou et al., “Survey of Fraud Detection Techniques,” *IEEE International Conference on Networking, Sensing and Control, 2004*, Taipei, Taiwan, vol. 2, pp. 749-754, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[8] E.W.T. Ngai et al., “The Application of Data Mining Techniques in Financial Fraud Detection: A Classification Framework and an Academic Review of Literature,” *Decision Support Systems*, vol. 50, no. 3, pp. 559-569, 2011. [CrossRef] [Google Scholar] [Publisher Link]