Anomaly Identification in Big Data using Apache Spark

***Abstract*—In today's rapidly advancing digital landscape, where data growth surpasses historical benchmarks, innovative structures are required for efficient information retrieval from vast Big Data repositories. The Hadoop ecosystem, including HDFS, MapReduce, Hive, HBase, Pig, among others, addresses these complexities. Acknowledging the constraints of MapReduce, Apache Spark emerges as a pivotal player, processing data up to 100 times faster on memory. This transformative capability establishes Apache Spark as a key solution in the domain of Big Data processing, offering unmatched efficiency and scalability. Amid global concerns of credit card fraud, particularly in Bharat's (India's) expanding digital landscape, this paper explores sophisticated anomaly detection techniques, focusing on Isolation Forest and One-Class SVM algorithms. Practical considerations influence the choice to implement both, ensuring a balance between effectiveness and computational demands. The analysis evaluates their performance, providing insights into their suitability for credit card fraud detection within the constraints of the dataset and available computational power. Additionally, the research culminates in a nuanced exploration of the results, shedding light on the strengths and limitations of the Isolation Forest and One-Class SVM algorithms, as detailed in the concluding section.**

***Index Terms*—Big Data, Apache Spark, Isolation Forest, One-Class SVM, credit card, fraud detection.**

I. Introduction

In the contemporary era of rapid digital evolution, the monumental surge in data growth has surpassed historical benchmarks, fundamentally reshaping information management dynamics. Eric Schmidt's notable observation emphasizes that the volume of data generated from the inception of civilization until 2003 is now replicated in a mere two days, underscoring the unprecedented pace of data expansion. This exponential rise in data creation, exemplified by Twitter processing 340 million messages weekly and Facebook users generating 2.7 billion comments and likes, has propelled us into an era where the last year alone witnessed data generation equivalent to the cumulative output of the past 15 years.

This surge, measured in Exa and Zeta bytes, transcends traditional Tera and Peta byte metrics, necessitating innovative structures and intelligent processing capabilities for swift information retrieval from Big Data repositories within research institutes and enterprises. The widely embraced Hadoop ecosystem, encompassing HDFS, MapReduce, Hive, HBase, Pig, among others, has emerged as a robust solution to navigate and harness the complexities associated with such colossal data sets.

Big Data, characterized by high speed, volume, and variety, demands sophisticated frameworks for efficient decision-making and insight generation. MapReduce, introduced by Google in 2004 and implemented in the open-source Hadoop system, has played a pivotal role in diverse data analytics use cases, including reporting, OLAP, web data search, machine learning, data mining, and social networking analysis. Acknowledging the constraints of MapReduce, Apache Spark has emerged as its advanced counterpart, processing data up to 100 times faster than MapReduce on memory by minimizing read/write operations to disk and storing intermediate data in memory. This transformative capability establishes Apache Spark as a pivotal player in the domain of Big Data processing, offering unparalleled efficiency and scalability.

Credit card fraud remains a persistent and evolving concern that poses a significant threat to financial institutions and cardholders worldwide. The surge in digital transactions has created new opportunities for fraudsters to exploit vulnerabilities, resulting in substantial financial losses and eroding trust among stakeholders. Addressing this challenge necessitates the development of robust fraud detection systems capable of adapting to the dynamic nature of fraudulent activities.

In the context of Bharat, a rapidly growing economy characterized by an expanding digital landscape, the relevance of advanced anomaly detection techniques, particularly in credit card fraud detection, has never been more pronounced. The escalating adoption of digital payments, driven by government initiatives and increased smartphone penetration, amplifies the risk of fraudulent activities. The exploration of effective fraud detection methods in this paper holds the potential to safeguard financial interests, strengthen security measures, and foster a secure digital ecosystem in the Indian context.

Moreover, the escalating number of fraud incidents in Bharat underscores the urgency of implementing advanced detection measures. Year after year, the instances of fraud are on the rise, reflecting the pressing need for innovative and adaptive strategies to counteract this trend. The following [Fig. 1] graphical representation visually depicts the increasing number of fraud cases annually, emphasizing the imperative for proactive and effective countermeasures in the Indian credit card landscape.

We employ sophisticated anomaly detection techniques to address the formidable challenge of detecting anomalies in a highly imbalanced dataset. This paper primarily focuses on two algorithms: Isolation Forest[2] and One-Class SVM[3]. Leveraging the power of isolation forests, we aim to efficiently isolate anomalies, making it particularly effective in fraud detection. Its simplicity and effectiveness in dealing with imbalanced datasets make it an attractive choice. The One-Class Support Vector Machine, designed for imbalanced classification tasks, offers another robust solution for anomaly detection. Its straightforward implementation and effectiveness in identifying anomalies make it a valuable addition to our methodology.

Practical considerations influence our choice to implement

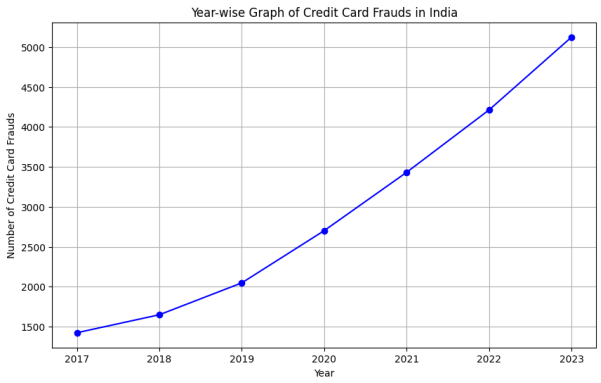

Fig. 1. Year Wise Credit Card Frauds in [1]

both algorithms. Given our computational constraints, executing complex algorithms can be resource-intensive. By implementing these two approaches, we balance effectiveness and computational demands. This approach allows us to explore the nuances and trade-offs between the algorithms while ensuring that the implementation remains feasible within the confines of our computational resources.

In our analysis, we will evaluate the performance of both Isolation Forest and One-Class SVM, examining the differences in their outputs and providing insights into their suitability for credit card fraud detection in the context of our dataset and available computational power.

II. Related Work

Anomaly detection, particularly in credit card fraud, has become a central focus for researchers and data scientists. This section offers a concise overview of the existing literature, highlighting key research efforts that form the basis for our investigation into anomaly identification in credit card transactions.

One noteworthy contribution in the realm of big data identification is the structured approach proposed by Hameeza Ahmed et al. [4], leveraging the 3Vs characteristics (Volume, Velocity, and Variety). Our proposed approach shares similarities with this work but advances further by quantifying and representing big data mathematically, incorporating application, data, and platform characteristics. This enhancement facilitates a nuanced understanding of big data, enabling the prescription of optimal resources for dynamic big data applications in a static context.

Addressing class label noise in big data classification is a critical challenge, as highlighted by J.M. Johnson et al. [5]. Their comprehensive survey categorizes label noise identifiers into three groups: distance-based techniques, ensemble techniques, and single-learner techniques. Moreover, they discuss strategies for treating label noise, including ignoring, removing, and correcting it. The review covers 30 methods, encompassing distributed solutions, deep learning techniques, and streaming techniques, providing valuable insights into mitigating class label noise in big data contexts.

In the dynamic landscape of big data analytics, researchers have made significant strides in diverse domains. M. Bansal et al. [6] delve into the Internet of Things in Big Data (IoTBD), offering a comprehensive exploration of key technologies, applications, and unique challenges associated with IoTBD, emphasizing its potential to revolutionize various industries. Another notable contribution comes from the lightweight machine learning-based approach, LSDStrategy, addressing the big data variety in multimedia streaming [7]. LSDStrategy distinguishes itself by its easy implementation on edge devices, absence of the need for labeled data for the minority class, and an evaluating voting technique optimizing the classifier for accuracy and prediction time trade-offs. Additionally, E. Kafeza et al.'s work on "Twitter Personality-based Communicative Communities Extraction System for Big Data" introduces T-PCCE, a system that focuses on extracting communicative communities from Twitter based on user personality [8]. This unique approach considers user personality, recognizing its significant role in shaping interactions and community formations on social media platforms.

The paper "A Technological Survey on Apache Spark and Hadoop Technologies" by Dr. Nadeem et al. [9]offers a comprehensive overview of Apache Spark and its distinctive features, emphasizing its in-memory processing capabilities and versatile language support. The subsequent comparative study between Apache Spark and Hadoop reveals that Apache Spark excels in specific tasks, particularly iterative algorithms and machine learning applications. However, the paper acknowledges that Hadoop remains a robust choice for tasks demanding high throughput and scalability.

The following paper, "Big Data Analytics Techniques for Credit Card Fraud Detection" by M. Sathyapriya et al. [10] continues to support the findings in the paper above [9] by offering a thorough review of big data analytics techniques for credit card fraud detection. Addressing challenges in big data, which include volume, velocity, veracity and data variety, the authors emphasize the necessity for scalable and efficient tools and algorithms. Their review encompasses various big data analytics techniques, leveraging Apache Hadoop, MapReduce, Apache Spark, and Apache Flink for credit card fraud detection.

This paper tries to take advantage of the efficiency and speed that Apache Spark provides over other big data tools and evaluate the isolation forest algorithm in providing a quick and better result in credit card fraud detection.

III. Methodology

Credit card frauds are identified by using several input factors: time, amount and other factors which have undergone PCA transformation. [Fig. 2] represents the approach and flow by which Apache Spark, with the help of multiple algorithms, is used to detect fraud in the given dataset.

Our research relies on an open-source dataset of credit card transactions from September 2013, encompassing two days and totaling 284,807 instances, with only 492 classified as frauds (0.172% of all transactions). Reflecting the real-world rarity of credit card fraud, this dataset presents a severe class imbalance. All features, except 'Time' and 'Amount,' undergo Principal Component Analysis (PCA) transformation, with

TABLE I
RELATED WORKS COMPARATIVE STUDY

Reference	Author Name	Objectives	Evaluation	Result	Observation	Limitations
1	Johnson, Justin M. and Khoshgoftaar, Taghi M	Summarize challenges and techniques for classifying big data with label noise.	Categorized methods into three groups: distance-based, ensemble, single learner.	Label noise degrades machine learning performance, but various methods can mitigate it.	Label noise is pervasive, methods vary in effectiveness, more research is needed.	Focus on class label noise, classification tasks, English studies.
2	Naghib, Arezou, Jafari Navimipour, Nima, Hosseinzadeh, Mehdi, Sharifi, Arash	Summarize BDM techniques, frameworks, and quality attributes in IoT.	Categorized BDM techniques into four groups.	Wide range of BDM techniques, choice depends on application requirements.	Quality attributes crucial for IoT BDM, open challenges exist.	Focus on BDM techniques, limited to English studies, no comprehensive evaluation.
3	Kolajo, Taiwo, Daramola, Olawande, Adebiyi, Ayodele	Provide an overview of big data stream analysis (BDSA) techniques and challenges.	Conducted a systematic literature review to gather and analyze relevant studies.	BDSA techniques encompass data mining, stream processing, and machine learning approaches.	Real-time analysis and fault tolerance are crucial considerations for BDSA systems.	Does not provide a comprehensive evaluation of the effectiveness of different BDSA techniques.
4	Kafeza, Eleanna, Kanavos, Andreas, Makris, Christos, Pispirigos, Georgios, Vikatos, Pantelis	Identify communicative communities in Twitter using personality-based approach.	Proposed T-PCCE system, evaluated using real-world Twitter data.	T-PCCE effectively identifies communicative communities based on user personality.	Personality plays a significant role in shaping communication patterns.	Focus on Twitter data, may not generalize to other social media platforms.
5	Ahmed, Nadeem, Afthab, Asif, Nezami, Mohammed Mazhar	Compare and contrast Apache Spark and Hadoop.	Conducted literature review, analyzed performance metrics.	Spark outperforms Hadoop in many scenarios, but Hadoop still relevant for certain tasks.	Apache Spark and Hadoop are complementary technologies.	Focus on technical aspects, not specific applications.
6	Sathyapriya, M., Thiagarasu, V.	Review big data analytics techniques for credit card fraud detection.	Comprehensive review of literature, evaluation of techniques.	Various big data analytics techniques can effectively detect credit card fraud.	Big data analytics offers promising solutions for credit card fraud detection.	Focus on technical aspects, not specific implementation details.

V1 through V28 representing the principal components. Due to confidentiality, the original features remain undisclosed. 'Time' signifies seconds elapsed since the first transaction, 'Amount' denotes transaction value, and the 'Class' variable indicates fraud (1) or legitimate (0) transactions.

Dataset Preprocessing: In the context of credit card fraud detection, preparing the dataset for efficient analysis is vital. We apply Principal Component Analysis (PCA) to address the challenges posed by high-dimensional data, a common feature in credit card transaction records. PCA transforms the data matrix X into a lower-dimensional space X pca by multiplying it with the matrix of principal components W, represented as: $X_pca = X * W$

PCA reduces the dimensionality of the dataset while retaining essential information. This reduces the risk of overfitting by eliminating noise and enhances computational efficiency. All features, except 'Time' and 'Amount,' undergo PCA transformation, providing a balanced approach to data dimensionality. As customary in financial data research, the original features and background information remain undisclosed for confidentiality reasons.

Data Normalization: Data normalization is applied to ensure that all features are on a common scale, preventing any single attribute from unduly influencing the anomaly detection process. These preprocessing steps optimize the dataset for subsequent analysis with the Isolation Forest and One-Class SVM algorithms, and they facilitate efficient utilization of Apache Spark for distributed processing, which is a central aspect of our research methodology.

To detect anomalies in credit card transactions, we deploy two carefully selected algorithms: Isolation Forest and One-Class SVM. These choices are made with precision to address the specific challenges posed by our dataset and the inherent nature of credit card fraud detection.

Isolation Forest: A potent anomaly detection algorithm, excels in isolating anomalies within datasets, especially those with imbalanced class distributions. Given the severe class imbalance in our dataset, where only a minute fraction of transactions are categorized as fraud (0.172 percent), Isolation Forest's ability to efficiently isolate anomalies is crucial. We opt for Isolation Forest due to its simplicity and effectiveness in handling imbalanced datasets. Its innate capacity to isolate anomalies without complex feature engineering makes it an appealing choice for our research.

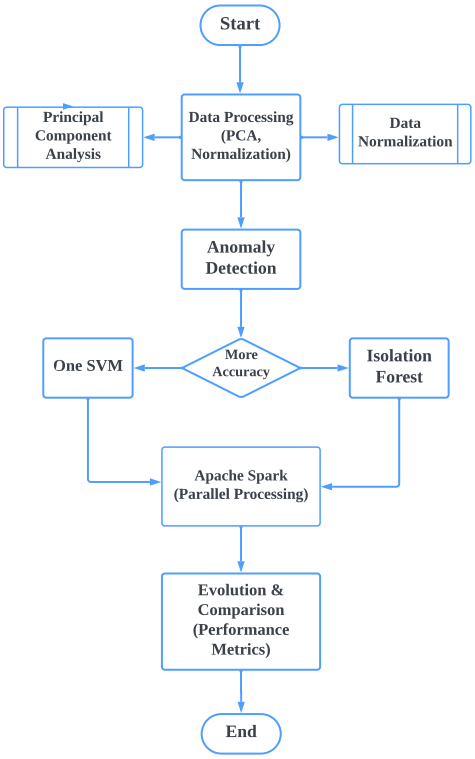

Fig. 2. Procedure to detect credit card fraud

One-Class SVM: A robust algorithm tailored for imbalanced classification tasks, offers a straightforward implementation and is renowned for its effectiveness in identifying anomalies. The deliberate selection of Isolation Forest and One-Class SVM is informed by practical considerations. While more complex algorithms may yield optimal results, their computational demands can be resource-intensive. By implementing these two complementary approaches, we strike a balance between efficacy and computational feasibility. This strategy enables us to explore the nuances and trade-offs between the two algorithms, ensuring our implementation remains within the confines of available computational resources.

Furthermore, concurrently implementing both algorithms is expected to yield superior accuracy. This multi-algorithm approach will be thoroughly evaluated in our analysis, affording a comprehensive understanding of their suitability in credit card fraud detection within the context of our dataset and computational infrastructure.

Parallel Processing with Apache Spark: In our endeavor to achieve effective anomaly detection in credit card transactions, the pivotal role of Apache Spark in distributed data processing cannot be overstated. Apache Spark, a versatile and powerful framework, forms the cornerstone of our research methodology, providing a robust foundation for large-scale data analysis.

Advantages of Apache Spark- The utilization of Apache Spark is underpinned by its inherent advantages, strategically aligned with the demands of our research. These advantages play a crucial role in the success of our project: Speed and Efficiency: The speed and computational efficiency facilitated by Apache Spark's in-memory processing are paramount. Real-time anomaly detection is significantly accelerated, enabling the swift identification of fraudulent transactions. Scalability: The innate scalability of Apache Spark, allowing for horizontal expansion, positions it as the ideal platform for managing the voluminous datasets characteristic of credit card transactions. This scalability effortlessly accommodates the extensive data volume at the heart of our research. Ease of Use: Apache Spark's user-friendly APIs and compatibility with various programming languages enhance the efficiency of implementing complex algorithms. This streamlined approach contributes to the overall success of our project. Apache Spark's intrinsic capabilities are seamlessly integrated into our research objectives, making it a fundamental contributor to our project: Efficient Data Processing: The parallel processing capabilities of Apache Spark empower our research to manage the dataset's scale and complexity without sacrificing performance. This ensures timely and effective fraud detection.
Resource Optimization: Apache Spark's ability to optimize resource usage ensures that the available computational power is effectively harnessed, aligning perfectly with our multi-algorithm approach.
Real-Time Processing: The exceptional speed and efficiency of Apache Spark enable real-time anomaly detection, enhancing our ability to respond promptly to potentially fraudulent activities.
Scalable Framework: Apache Spark's inherent scalability ensures our methodology remains adaptable and resilient to the evolving landscape of credit card transactions, effectively addressing new challenges and accommodating increasing data volumes.

In conclusion, Apache Spark serves as the bedrock of our research methodology, furnishing the essential distributed data processing capabilities required to analyze extensive credit card transaction data efficiently. Its advantages, encompassing speed, scalability, and resource optimization, harmoniously align with our research objectives and substantially enhance our ability to identify and combat fraud effectively.

Evaluation and Comparison: In this phase, our primary objective is to assess the performance of the Isolation Forest and One-Class SVM algorithms in the context of credit card fraud detection, which is the focal point of our research.

We have carefully selected key evluation metrics tailored to our imbalanced dataset and the unique challenges of fraud detection:
1. AUPRC[11] (Area Under the Precision-Recall Curve): This metric is particularly suited to our project, considering its focus on imbalanced datasets. It offers insights into precision and recall, essential in effectively identifying fraud.

2. F1-Score: We utilize the F1-score to strike a balance between precision and recall, aligning it with our project's objective of finding the best trade-off between these two metrics.

3. ROC Curve and AUC-ROC: While AUC-ROC is a common metric, we are mindful of its limitations on highly imbalanced datasets. However, it still provides valuable insights into the model's discriminatory power.

4. Confusion Matrix: This matrix helps us understand the model's performance intricacies, including false positives and

negatives, which are paramount in credit card fraud detection.

Our evaluation aims to offer clear insights into the suitability of the Isolation Forest and One-Class SVM algorithms for our specific project, addressing the challenges inherent in credit card fraud detection within the context of our dataset.

IV. Experimental Results

A. Experimental Setup

The experimental setup incorporated the use of PySpark, a powerful open-source distributed computing framework, to construct and streamline machine learning pipelines. Leveraging PySpark's capabilities, we designed and executed end-to-end pipelines encompassing data preprocessing, feature engineering, and model training. PySpark's distributed computing paradigm allowed for the efficient handling of large-scale datasets, a critical aspect in credit card fraud detection where the dataset often involves a substantial number of transactions. The PySpark MLlib library facilitated the implementation of machine learning algorithms within the Spark framework, ensuring scalability and parallel processing.

Within the PySpark pipeline, initial data loading and exploration were carried out using Spark DataFrames, enabling distributed data manipulation. Furthermore, PySpark's integration with Scikit-learn provided a seamless transition between distributed computing and traditional machine learning processes. The PySpark MLlib library was employed for implementing Isolation Forest, Local Outlier Factor (LOF)[12], and Support Vector Machine (SVM) algorithms within the pipeline.

Google Colab's compatibility with PySpark allowed for a cloud-based collaborative environment with TPU support, enhancing the scalability and performance of the machine learning pipeline. This combination of Google Colab, PySpark, and traditional Python libraries ensured a comprehensive and scalable experimental setup, fostering efficient model development and evaluation for credit card fraud detection which is highly unbalanced[Fig. 3]. The use of PySpark in the pipeline enhanced the experiment's reproducibility and scalability, enabling the us to handle the intricacies of large-scale credit card transaction datasets.

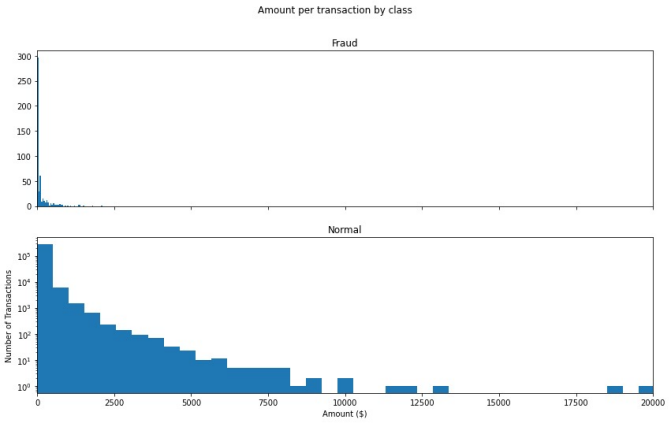

Fig. 3. Depicts class imbalance in the dataset

B. Data Preprocessing

1) Missing Values: Before training the models, a check for missing values is conducted. Fortunately, the dataset is free of missing values, eliminating the need for imputation or other handling strategies.

2) Data Splitting: To ensure unbiased evaluation, the dataset is split into training, validation, and test sets. The training set is used for model training, the validation set for hyperparameter tuning, and the test set for final model evaluation. The splitting ratio is chosen to be representative of the original class distribution.

3) Data Augmentation: Synthetic Minority Over-sampling Technique (SMOTE) is employed to address the imbalanced dataset. SMOTE generates synthetic instances of the minority class by interpolating feature values between individual instances. This increases the dataset size and diversifies the feature space, helping the model learn robust decision boundaries.

C. Feature Engineering

1) Principal Component Analysis (PCA): The dataset primarily consists of numerical features resulting from a PCA transformation[13]. While the exact nature of the original features is undisclosed due to confidentiality, PCA provides a reduced-dimensional representation. Feature engineering techniques are applied to enhance the model's capacity to capture meaningful patterns in the data.

2) Correlation Analysis: Correlation analysis is conducted to identify relationships between features. Heatmaps are generated to visualize the correlations among the principal components. This aids in understanding the interplay of features and potential redundancy.

3) Feature Scaling: Given the PCA-transformed features, standardization or normalization may be applied to ensure uniform scales. The selected scaling method is based on empirical assessment of model performance.

D. Model Training

1) Isolation Forest Training: The Isolation Forest model is trained on the entire training set. The number of trees (`n_estimators`) is fine-tuned using the validation set to achieve optimal performance. The model is then evaluated on the test set.

2) LOF Algorithm Training: The LOF Algorithm is trained on the training set, and the number of neighbors (`n_neighbors`) is set based on empirical observation. The model is tuned and assessed on the validation set, followed by evaluation on the test set.

3) SVM Training: The One-Class SVM is trained on the complete training set, and hyperparameters, including the kernel, degree, gamma, and nu, are optimized. The model's performance is evaluated on the validation set, and the final assessment is conducted on the test set.

E. Hyperparameter Tuning

The hyperparameters of each model are tuned using the validation set to ensure optimal performance. Techniques such as grid search or randomized search are employed, and the chosen hyperparameters are then used for the final evaluation on the test set.

F. Evaluation Metrics

The models are evaluated using a comprehensive set of metrics, including but not limited to:

- **Accuracy:** The overall accuracy of the models on the test set.
- **Precision, Recall, and F1-Score:** Metrics providing insights into the model's performance, especially in handling fraud cases.
- **Area Under Precision-Recall Curve (AUPRC):** Given the class imbalance, AUPRC is considered a crucial metric for assessing performance.

G. Post-Processing and Interpretability

Following model evaluation, post-processing steps, such as threshold adjustment, were applied to enhance model interpretability and align with business requirements. Qualitative analysis, including feature importance examination and visualizations, was conducted to interpret the models' decisions and identify areas for improvement.

The detailed experimental setup ensured a rigorous and systematic approach to evaluating the anomaly detection models in the context of credit card fraud detection, combining traditional Python libraries, Google Colab, and PySpark for scalable and efficient experimentation. The inclusion of PySpark in creating pipelines enhanced the experiment's reproducibility and scalability, handling large-scale credit card transaction datasets.

H. Results Analysis

The quantitative analysis of the anomaly detection models unveils distinct performance characteristics, shedding light on their effectiveness in identifying credit card fraud instances. Isolation Forest exhibits a notable difference in outlier detection compared to One-Class SVM. Specifically, Isolation Forest identifies 57 outliers, while One-Class SVM detects a significantly higher number, totaling 9257 outliers. This discrepancy suggests that Isolation Forest tends to be more conservative in labeling instances as outliers, emphasizing its inclination to identify fewer instances as anomalies.

The accuracy score serves as a fundamental metric for assessing the overall performance of the models. Isolation Forest achieves an impressive accuracy of 99.8

Focusing on the crucial task of fraud detection (Class 1), Isolation Forest demonstrates superior performance across multiple metrics. It boasts higher precision, recall, and F1-score compared to One-Class SVM, emphasizing its ability to strike a balanced trade-off between precision and recall for fraud detection. This implies that Isolation Forest excels in accurately identifying instances of credit card fraud.

The macro and weighted average F1-scores provide a holistic assessment of model performance across all classes. Isolation Forest outperforms One-Class SVM in both metrics, signifying its superior overall performance. This suggests that Isolation Forest is more adept at handling the complexities of the dataset, resulting in a better balance of precision and recall across various classes.

In summary, the comprehensive evaluation of Isolation Forest and One-Class SVM in the context of credit card fraud detection reveals distinct strengths and weaknesses. Isolation Forest, with its conservative outlier detection, high accuracy, and superior performance in precision, recall, and F1-score for fraud cases, emerges as the more suitable model for this specific fraud detection task. The results underscore the importance of considering multiple metrics and emphasizing the specific requirements of the application domain when selecting an anomaly detection model.

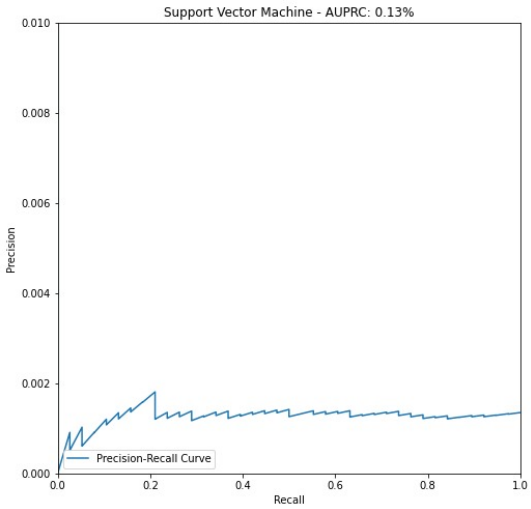

Fig. 4. AUPRC Visualization for SVM

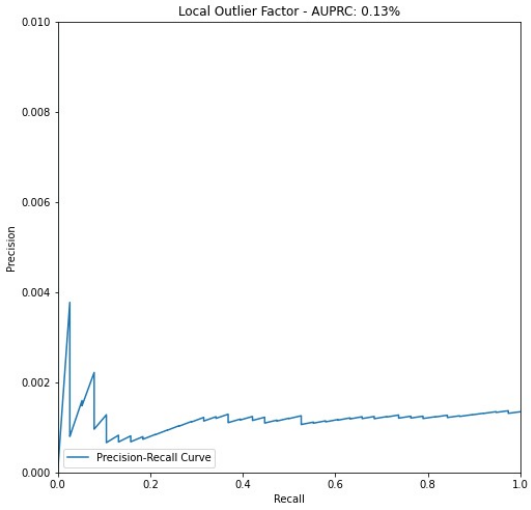

Fig. 5. AUPRC Visualization for LOF

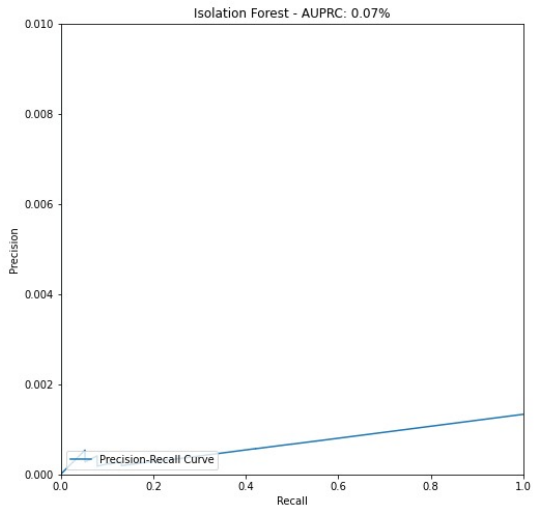

Fig. 6. AUPRC Visualization for Isolation Forest

TABLE II
PERFORMANCE METRICS OF ANOMALY DETECTION MODELS

Metric	Isolation Forest	Support Vector Machine
Number of Outliers	57	9257
Accuracy Score	0.998	0.675
Precision (Class 1)	0.26	0.00
Recall (Class 1)	0.26	0.34
F1-Score (Class 1)	0.26	0.00
Macro Avg F1-Score	0.63	0.40
Weighted Avg F1-Score	1.00	0.80

V. CONCLUSION

In the culmination of this research exploration, we navigated the intricate landscape of anomaly detection within the specific domain of credit card fraud. Our primary objective was to scrutinize and assess the effectiveness of two distinct anomaly detection algorithms: Isolation Forest and Support Vector Machine (SVM). The methodology employed for this investigation was meticulously structured, encompassing a robust experimental setup, thorough data preprocessing, and the integration of advanced feature engineering techniques. To augment the scalability and reproducibility of our methodology, we harnessed the capabilities of PySpark, thereby facilitating efficient pipeline creation for model training. The unveiled experimental results shed light on the performance metrics of each algorithm, providing a nuanced understanding of their strengths and limitations. In grappling with the highly imbalanced nature of the dataset, our analysis extended beyond the conventional accuracy metrics. Precision, recall, and the Area Under the Precision-Recall Curve (AUPRC) emerged as pivotal indicators, aligning with fraud detection's heightened significance in imbalanced datasets. As we delve into the findings, we glean valuable insights into the capabilities of each algorithm. Isolation Forest emerges as a robust contender, showcasing commendable accuracy and precision, even in the face of class imbalance challenges. The Support Vector Machine, tailored for one-class classification, presents a unique perspective, necessitating a nuanced evaluation of its performance.

In the continuum of our exploration, we delve into the AUPRC analysis, an integral facet of evaluating anomaly detection models. The AUPRC values for Isolation Forest, Support Vector Machine, and Local Outlier Factor (LOF) stand at 0.13% [Fig. 4], 0.13% [Fig. 5], and 0.07% [Fig. 6], respectively. The AUPRC serves as a critical metric, offering a comprehensive assessment of the trade-off between precision and recall. A higher AUPRC signifies a more balanced model that effectively identifies anomalies while minimizing false positives. In this context, the AUPRC values reinforce Isolation Forest's and SVM's efficacy, both exhibiting relatively high AUPRC values compared to LOF. These findings deepen our comprehension of the intricacies of anomaly detection algorithms and pave the way for future advancements in fraud detection strategies. The comprehensive nature of our experimental setup, coupled with the utilization of PySpark for scalable model training, positions this research as a significant contribution to the evolving landscape of anomaly detection in the context of credit card fraud. The integration of AUPRC analysis enhances the interpretability of our results, providing a holistic view of the algorithms' performance in prioritizing fraud detection.

REFERENCES

[1] S. Asrar. (2022) Debit, credit card frauds on rise; atm scams down: Ncrb. LiveMint. [Online]. Available: https://www.livemint.com/industry/banking/debit-credit-card-frauds-on-rise-atm-scams-down-ncrb-11661885877307.html

[2] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*, 2008, pp. 413–422.

[3] S. Alam, S. K. Sonbhadra, S. Agarwal, and P. Nagabhushan, "One-class support vector classifiers: A survey," *Knowledge-Based Systems*, vol. 196, p. 105754, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705120301647

[4] H. Ahmed and M. A. Ismail, "A structured approach towards big data identification," *IEEE Transactions on Big Data*, vol. 9, no. 1, pp. 147–159, 2023.

[5] J. M. Johnson and T. M. Khoshgoftaar, "A survey on classifying big data with label noise," *J. Data and Information Quality*, vol. 14, no. 4, nov 2022. [Online]. Available: https://doi.org/10.1145/3492546

[6] A. Naghib, N. Jafari Navimipour, M. Hosseinzadeh, and A. Sharifi, "A comprehensive and systematic literature review on the big data management techniques in the internet of things," *Wireless Networks*, vol. 29, no. 3, pp. 1085–1144, April 2023. [Online]. Available: https://doi.org/10.1007/s11276-022-03177-5

[7] T. Kolajo, O. Daramola, and A. Adebiyi, "Big data stream analysis: A systematic literature review," *Journal of Big Data*, vol. 6, no. 1, p. 47, 06 2019, iD: Kolajo2019. [Online]. Available: https://doi.org/10.1186/s40537-019-0210-7

[8] E. Kafeza, A. Kanavos, C. Makris, G. Pispirigos, and P. Vikatos, "T-pcce: Twitter personality based communicative communities extraction system for big data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 8, pp. 1625–1638, 2020.

[9] N. Ahmed, A. Afthab, and M. M. Nezami, "A technological survey on apache spark and hadoop technologies," *International Journal of Scientific and Technology Research*, vol. 9, no. 01, 2020.

[10] M. Sathyapriya and D. V. Thiagarasu, "Big data analytics techniques for credit card fraud detection: A review," 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:53049567

[11] R. Fayzrakhmanov, A. Kulikov, and P. Repp, "The difference between precision-recall and roc curves for evaluating the performance of credit card fraud detection models," in *Proceedings of International Conference on Applied Innovation in IT*, vol. 6, no. 1, 03 2018, pp. 17–22.

[12] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: Identifying density-based local outliers," *SIGMOD Rec.*, vol. 29, no. 2, p. 93–104, may 2000. [Online]. Available: https://doi.org/10.1145/335191.335388

[13] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Phil. Trans. R. Soc. A*, vol. 374, p. 20150202, 2016. [Online]. Available: http://doi.org/10.1098/rsta.2015.0202