SECURE AND SCALABLE HORIZONTAL FEDERATED LEARNING FOR BANK FRAUD DETECTION

Nnaemeka Obiefuna,* **Iremide Oyelaja,*** **Similoluwa Odunaiya,*** **Samuel Oyeneye***
ML Collective

ABSTRACT

Financial fraud detection presents a critical challenge: balancing high model accuracy with stringent data privacy regulations such as GDPR and CCPA. Centralized machine learning approaches, which require pooled transaction data, pose significant privacy risks, while institution-specific models suffer from data scarcity. We propose a privacy-preserving framework for fraud detection using Horizontal Federated Learning (HFL). Our study compares three paradigms: (i) global centralized models, (ii) partially isolated models, and (iii) HFL, trained using FedAvg with the Flower framework using deep learning. Experiments on the BAF-base datasets simulate real-world fraud detection scenarios, evaluating key performance metrics, including ROC-AUC, and time efficiency. We benchmark comparisons with SOTA models on the BAF-base dataset to validate our approach further. The results highlight trade-offs between data privacy, model performance, and generalization ability, demonstrating that Federated Learning is a viable alternative that effectively balances security, efficiency, and predictive performance in financial fraud detection.

1 INTRODUCTION

Financial fraud is an escalating threat, costing global bank institutions billions of dollars annually. Traditional fraud detection models rely on centralized machine learning, requiring access to large amounts of transaction data. However, privacy regulations such as the General Data Protection Regulation (GDPR) (European Parliament, 2016), the California Consumer Privacy Act (CCPA) (California Legislative Information, 2018), and the Revised Payment Services Directive (PSD2) (European Parliament, 2015) restrict data sharing across institutions, limiting the effectiveness of these models. This creates a tension between improving fraud detection and ensuring data privacy. Federated Learning offers a promising solution by allowing multiple banks to train a fraud detection model collaboratively without sharing raw customer data. Specifically, Horizontal Federated Learning (HFL) (Malgorzata et al., 2024) enables institutions with similar data structures (e.g., different banks with transaction records) to jointly improve fraud detection while using the HFL privacy-preserving framework that allows banks to detect fraud patterns across institutions while keeping data local.

In our approach, we use Federated Averaging (FedAvg) with the Flower framework to train models across multiple institutions while preserving data locally. While FedAvg ensures that raw data remains on client devices, it does not inherently provide secure aggregation. We demonstrate that our method maintains data privacy by preventing direct data sharing and achieves a competitive fraud detection accuracy compared to single-institution models while maintaining data security. We also leverage attention mechanism as one of our federated approaches, detailed in Section 3, with visual representations of the federated MLP and federated transformer workflows, presented in Figures 2 and 3 respectively, including comparison results with existing SOTA benchmarks on BAF fraud detection data. Our contributions:

1. Demonstrates the feasibility and effectiveness of using FL for collaborative machine learning in the fintech & banking industry.

*These authors contributed equally to this work.

2. Provide a practical example of how banks and fintech companies can build a robust fraud detection model using FL.

3. Shows that FL is capable of achieving similar model accuracy to a centralized approach while preserving data privacy.

4. Securely aggregate the model parameters without the central body having access to the raw data, demonstrating improved fraud detection accuracy and efficiency over traditional models.

5. Introduces a federated transformer-based architecture, outperforming the previous MLP based HFL model, validated on the BAF dataset.

2 Related Works

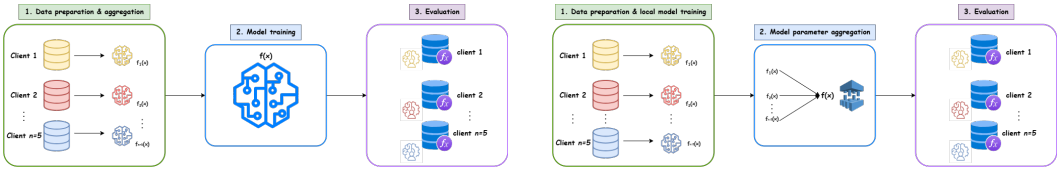
Traditional Fraud Detection in Banking Traditional fraud detection in banking relies on centralized machine learning models trained on a single institution's dataset. Approaches like supervised learning (e.g., Random Forest, XGBoost, Neural Networks) and unsupervised anomaly detection methods (e.g., Auto-encoders, Isolation Forests) are common. However, these models often suffer from data limitations and lack cross-institutional fraud intelligence, making them less effective against new fraud patterns.

Several surveys detail the evolution of fraud detection techniques. (Nilofar et al., 2022), provide an overview of rule-based systems, machine learning models, and deep learning techniques for detecting fraudulent transactions. (Clifton et al., 2010), categorize fraud detection methods into statistical approaches, AI-based methods, and hybrid systems used in banking, telecommunications, and cybersecurity. (Andrea Dal et al., 2017), discuss state-of-the-art machine learning approaches, emphasizing the challenges of class imbalance, real-time detection, and feature engineering.

Federated Learning for Financial Applications Federated Learning (FL) has emerged as a promising technique for collaborative machine learning while preserving data privacy. (Keith et al., 2016a), introduced FL to enable decentralized training without exposing raw data. (Stephen et al., 2017), applied FL in financial risk assessment, demonstrating its ability to train models across banks without violating privacy regulations. Recent research has explored FL for fraud detection, such as (Wensi et al., 2019), where different institutions hold different feature sets for the same users. In contrast, Horizontal Federated Learning, which allows multiple banks with similar data structures to collaborate, remains under-explored in fraud detection.

Privacy-Preserving Techniques in FL FL include Secure Aggregation (Keith et al., 2016b), which encrypts model updates to prevent information leaks; Differential Privacy (Martín et al., 2016), which introduces noise to model updates to prevent sensitive data reconstruction; and Homomorphic Encryption & Secure Multi-Party Computation (Gentry, 2009), which allows encrypted computations to enhance privacy protection. Despite its promise, FL introduces potential risks, including data leakage and susceptibility to adversarial attacks, which numerous studies actively address.

Challenges in FL for Fraud Detection Existing FL applications face challenges in data heterogeneity, communication efficiency, and adversarial robustness. (Tian et al., 2020) proposed optimization techniques for handling non-IID (non-independent and identically distributed) data. (Pranav et al., 2020), explored adversarial attacks on FL, which is crucial for fraud detection where fraud patterns vary across banks. Showing that model poisoning could degrade performance, making security measures essential. In response to these challenges, we specifically investigate whether banks and fintech companies can leverage FL to develop accurate predictive models without sharing sensitive customer data and how FL-based solutions compare to conventional data pooling methods. We also explore whether FL can effectively mitigate data scarcity and privacy constraints that frequently hamper collaboration in the banking sector.



(a) Traditional Data Pooling and Model Training (b) Federated (Horizontal) Model Training

Figure 1: Comparison of traditional centralized (a) and federated (b) approaches to fraud detection model training

3 Dataset & Methodology

3.1 Overview

In the Horizontal Federated Learning (HFL) training paradigm, (Malgorzata et al., 2024) introduced an approach that mitigates the challenges of traditional machine learning, particularly reducing exposure to sensitive sector-specific data while ensuring optimized performance. For our study, we utilized the Bank Account Fraud (BAF) dataset (Jesus et al., 2022), a bank account creation dataset from Kaggle, to replicate the financial situation in a real-world banking scenario. The dataset was originally designed to replicate real-world financial fraud scenarios, providing a comprehensive and realistic testbed for evaluating machine learning and fairness-aware fraud detection techniques. To create a realistic federated learning environment, we simulated five distinct clients, representing different financial institutions, we realized a significant amount of class imbalance of the negative class samples to their corresponding positive class samples, we then downsampled the negative class by retaining 2% of its instances to achieve a more balanced marginal distribution.

Finally, we orchestrated and simulated these clients to prepare them for federated model training, ensuring that each client retained its local data while contributing to the collaborative learning process.

3.2 Training Paradigms

To distinguish the differences in paradigms of centralized and decentralized collaborative fraud detection strategies, we evaluate three distinct training paradigms, including the HFL technique;

Global Centralized Models As illustrated in Figure 1a, the global traditional model follows the conventional machine learning workflow, where all our combined simulated client data is pooled into a central repository and trains a single model on this pooled dataset. The entire data pipeline—loading, preprocessing, and feature engineering—occurs on a single platform where the unified dataset trains a shared model. We used four algorithms in this aspect (Logistic Regression, Random Forest, XGBoost, LightGBM), which were split into train-validate-test (80-10-10) splits. This approach assumes full data sharing between client's data, with raw transaction records centralized on one server. Although effective for model accuracy, global pooling raises practical concerns, as this violates GDPR principles by exposing sensitive customer data.

Partial Isolated Models Partial models simulate a scenario where each client operates in complete isolation. We trained each client with an independent XGBoost classifier (selected for its balance of recall and precision in fraud detection), using only each client's partitioned dataset (client 0, client 1, client 2, client 3, and client 4). Our training mirrors the global approach but operates on siloed data: client-specific splits are processed locally without any cross-institutional coordination. Although preserving privacy data by avoiding sharing, this approach suffers from the limited sample size, i.e about 20% of the global data per client, and fails to leverage a collective fraud pattern across the board

Horizontal Federated Learning (HFL) Model Our federated learning framework, inspired by the work of (Malgorzata et al., 2024), has been adapted for partitioned bank clients as in Figure 1b,

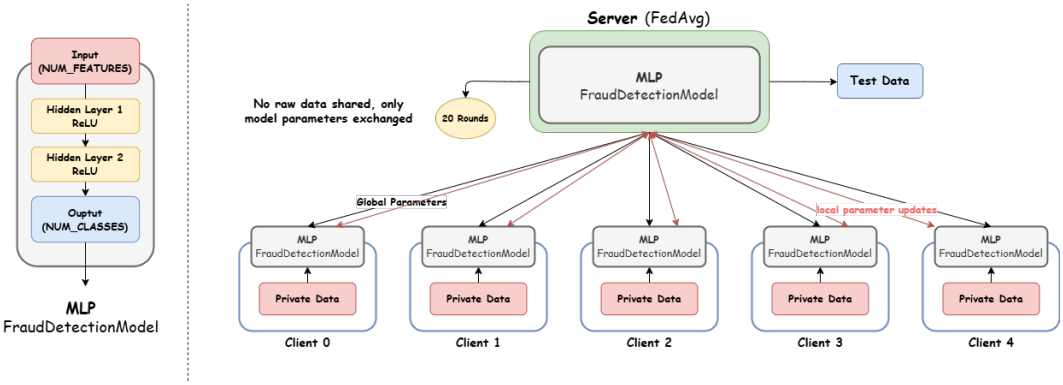


Figure 2: MLP (HFL) architecture for fraud detection

using flower (Daniel J. et al., 2020) with neural networks. In this setup, clients retain their raw data locally, as we explored two methodological approaches for collaborative training.

We first used a three-layer Multilayer Perceptron (MLP) with a structured architecture of 51 $\rightarrow$ 25 $\rightarrow$ 15 $\rightarrow$ 2 nodes (see Figure 2), through a secure parameter exchange:

1. **Local Training**: Each client initializes a neural network and trains for 12 epochs per communication round using the Adam optimizer (LR = 0.01). Our architecture integrates ReLU activations between the hidden layers to enhance learning performance.
2. **Secure Aggregation**: Model weights are encrypted and sent to our custom central server. This server applies Federated Averaging, (Pranav et al., 2020) FedAvg strategy in Flower, combining updates from all 5 clients while filtering outliers using weighted averaging.
3. **Global Synchronization**: Updated parameters are evaluated and redistributed to the FL clients for the next round, iterating for 20 rounds total.

In addition, we further explored an advanced model, where we used a federated transformer architecture. The transformer incorporates two encoder layers with multi-head self-attention and feedforward networks, which enhances the performance of non-IID features among clients' data that might not have correlated statistical distributions. This enhancement occurs through the transformer's ability to capture complex dependencies regardless of feature position or distribution characteristics. Specifically, just like in the attention paper (Vaswani et al., 2017),

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\mathbf{W}^O \tag{1}$$

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) \tag{2}$$

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{3}$$

$$\text{FFN}(\mathbf{x}) = \max(0, \mathbf{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2 \tag{4}$$

the multi-head self-attention mechanism allows the model to attend to different representation subspaces simultaneously. When dealing with bank fraud detection, transaction patterns may vary significantly across institutions due to different customer bases, regional behaviors, or bank-specific services. The self-attention mechanism enables the model to identify fraud patterns by focusing on relevant feature interactions rather than relying on consistent statistical distributions. Furthermore, the federated transformer architecture has input embeddings and positional encoding, culminating in an output linear layer, which also follows the same secure parameter exchange during training, only with a slightly different initialization step. Unlike the local training step used by the federated

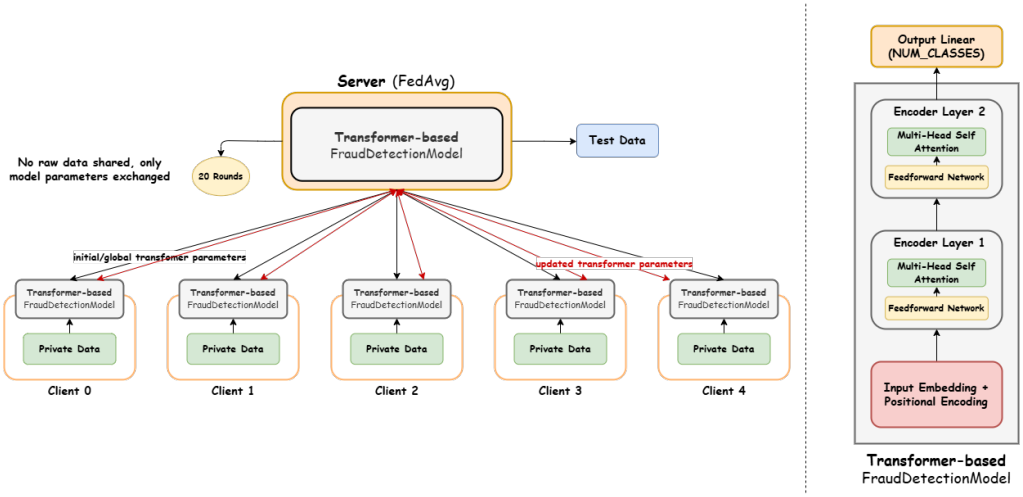


Figure 3: Federated Transformer architecture for fraud detection

MLP approach, in the federated transformer architecture, the server creates an initial transformer-based fraud detection model with a defined architecture in Figure 3. This initialization establishes the starting parameters that will be distributed to all participating clients.

These two approaches enforce data minimization by design: clients process data locally and the server only handles encrypted weight tensors (`ndarrays_to_parameters` conversion). While differential privacy and SMPC are planned for production deployments discussed in Section 5, the current simulation uses basic parameter encryption through Flower's built-in privacy safeguards. This tripartite paradigm systematically explores the accuracy-privacy trade-off inherent in collaborative fraud detection, providing a foundation for evaluating FL's viability in regulated financial environments.

4 Experiments

4.1 Results

We detailed our evaluated results in Table 1, which shows the collaborative training paradigms focused on ROC-AUC scores to quantify fraud detection performance across methodologies and their respective training time. For the global centralized model, four algorithms were trained on the pooled data, with LightGBM achieving the highest ROC-AUC (0.89), followed by XGBoost and Random Forest (0.88), and Logistic Regression (0.87), showing an upper bound baseline under unrestricted data sharing conditions.

In the partial isolated models, five clients trained independently using XGBoost. ROC-AUC scores varied across clients, ranging from 0.87 (Client 2) to 0.90 (Client 3), with a mean of 0.88 (±0.01 standard deviation). This reflects the impact of data heterogeneity and localized training limitations.

For the federated models, both trained for 20 rounds, the federated MLP achieved an ROC-AUC of 0.86, and the federated transformer achieved a 1% gain above the federated MLP. While marginally lower than the global LightGBM (0.89), it achieved the same result with three of the five partial models and surpassed one, demonstrating competitive performance without raw data exchange.

We also benchmark our HFL fraud detection models on the BAF-base data against other SOTA methods in Table 2, which were mostly from RIFF Martins et al. (2024).

Although the global and partial models slightly outperform our HFL approaches in terms of ROC-AUC, HFL adheres to the GDPR principles by preserving data privacy—offering a near-equivalent yet more secure option for collaborative fraud detection.

Table 1: Performance Metrics and Training Time for Selected Models Across Paradigms. For Global Centralized, LightGBM achieved the highest ROC-AUC (0.89). For Partial Isolated, Client 3 had the highest ROC-AUC (0.90). Federated models include MLP (ROC-AUC = 0.86) and Transformer (ROC-AUC = 0.87).

Paradigm	**Model/Client**	**ROC-AUC**	**Training Time (s)**
Global Centralized	LightGBM	0.89	5.13
Partial Isolated	Client 3	0.90	14.98
Federated (HFL)	MLP	0.86	164.17
Federated (HFL)	Transformer	0.87	675.18

Table 2: SOTA Benchmark on Fraud Detection BAF-Base Data. Asterisked models with Bold values show our HFL model performances compared to other models from the benchmark.

Model	**Recall @1% FPR**
LightGBM	25.2%
Federated Transformer*	**25%**
FIGS	21%
MLP+HFL*	**19%**
CART+RIFF	18.4%
CART	16%

4.2 Limitations

While our study demonstrates the feasibility and effectiveness of Horizontal Federated Learning (HFL) for fraud detection in banking, it is important to acknowledge the limitations associated with computational and communication overhead, including latency.

Federated learning, while effective for privacy-preserving fraud detection, incurs significant computational and communication costs compared to centralized methods. The federated Transformer model, with its advanced architecture, requires more training time (675.18 seconds) than the federated MLP (164.17 seconds) or centralized LightGBM (78.43 seconds) models (see Table 3). Communication demands are also higher, with the Transformer model having its total training communication rise to 138.6MB, exchanged for 20 rounds within 5 clients, considering both client-to-server and server-to-client iterations. This represents a 9,800% increase over the federated MLP's training communication (1.4MB), which, despite higher training efficiency, introduces higher latency during communication due to frequent data exchanges across nodes. As such, the federated approach incurs additional delays and resource consumption when compared to centralized methods, which do not face the same complexities due to the absence of aggregated servers. These overheads, detailed in Table 3, grow with model complexity, as seen in the Transformer's substantial demands versus the MLP's more manageable costs.

Table 3: Computational and Communication Overhead for Different Approaches

Approach	**Model**	**Training Time (s)**	**Training Communication (MB)**
Centralized	All	78.43	0
Partial Isolated	XGBoost	15.37	0
Federated	MLP	164.17	1.4
Federated	Transformer	675.18	138.6

5 Discussion

This research demonstrates that, through simulated bank account opening data, Federated Learning (FL) enabled secure collaboration among clients' banking institutions, alleviated data scarcity issues, and maintained a strong ROC-AUC score. By comparing centralized, partially isolated, and

federated approaches, we show that Horizontal Federated Learning (HFL) achieves an effective balance between predictive performance and data privacy. Our experiments using the BAF-base dataset reveal that federated models, especially those based on transformer architectures, can match or even outperform traditional centralized models as more clients join the collaborative training, all while preserving data security. These findings validate the potential of federated learning as a scalable and regulation-compliant solution for collaborative fraud detection across financial institutions.

While our approach has shown promising results, we acknowledge certain limitations in Subsection 4.2. Future work could focus on advanced hyperparameter tuning to further enhance accuracy with a lower computational cost. Additionally, exploring other privacy-enhancing techniques, such as Secure Multi-Party Computation (SMPC) for encrypting data and computation, differential privacy for adding controlled noise to model updates, and Secure Aggregation (SecAgg) or Homomorphic Encryption (HE) for enhanced privacy protection incorporated with deep learning, could help ensure scalable and secure solutions for financial applications.

REFERENCES

Pozzolo Andrea Dal, Boracchi Giacomo, Caelen Olivier, Alippi Cesare, and Bontempi Gianluca. Credit card fraud detection: A realistic modeling and a novel learning strategy. *IEEE, DOI: 10.1109/TNNLS.2017.2736643*, 2017.

Phua Clifton, Lee Vincent, Smith Kate, and Gayler Ross. A comprehensive survey of data mining-based fraud detection research. *arxiv preprint arXiv:1009.6119*, 2010.

Beutel Daniel J., Topal Taner, Mathur Akhil, Qiu Xinchi, Fernandez-Marques Javier, Gao Yan, Sani Lorenzo, Li Kwing Hei, Parcollet Titouan, Buarque de Gusmão Pedro Porto, and Lane Nicholas D. Flower: A friendly federated learning research framework. *arxiv preprint arXiv:2007.14390*, 2020.

Craig Gentry. Fully homomorphic encryption using ideal lattices. *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC'09), 169–178*, 2009. URL https://dl.acm.org/doi/10.1145/1536414.1536440.

Sérgio Jesus, José Pombal, Duarte Alves, André Cruz, Pedro Saleiro, Rita P. Ribeiro, João Gama, and Pedro Bizarro. Turning the Tables: Biased, Imbalanced, Dynamic Tabular Datasets for ML Evaluation. *Advances in Neural Information Processing Systems*, 2022.

Bonawitz Keith, Ivanov Vladimir, Kreuter Ben, Marcedone Antonio, McMahan H. Brendan, Patel Sarvar, Ramage Daniel, Segal Aaron, and Seth Karn. Practical secure aggregation for federated learning on user-held data. *arxiv preprint arXiv:1611.04482*, 2016a.

Bonawitz Keith, Ivanov Vladimir, Kreuter Ben, Marcedone Antonio, McMahan H. Brendan, Patel Sarvar, Ramage Daniel, Segal Aaron, and Seth Karn. Practical secure aggregation for federated learning on user-held data. *arxiv preprint arXiv:1611.04482*, 2016b.

Smietanka Malgorzata, Liew Dylan, Hand Scott, and Haoyuan Loh Harry. Privacy preserving neural network predictive modelling in insurance using horizontal federated learning. 2024. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4861490. Accessed: 2025-02-04.

Lucas Martins, João Bravo, Ana Sofia Gomes, Carlos Soares, and Pedro Bizarro. Riff: Inducing rules for fraud detection from decision trees. *arXiv preprint arXiv:2408.12989*, 2024. URL https://arxiv.org/abs/2408.12989.

Abadi Martín, Chu Andy, Goodfellow Ian, McMahan H. Brendan, Mironov Ilya, Talwar Kunal, and Zhang Li. Deep learning with differential privacy. *arxiv preprint arXiv:1607.00133*, 2016.

Mulla Nilofar, Shinde Sonali, Sudrik Preeti, Paliwal Sanskriti, and Bhat Tejasvi. Survey on fraud detection methods in banking transactions. *International Journal of Engineering Research Technology (IJERT)*, 2022.

Sankhe Pranav, Azim Saqib, Goyal Sachin, Choudhary Tanya, Appaiah Kumar, and Srikant Sukumar. Indoor distance estimation using lstms over wlan network. *arxiv preprint arXiv:2003.13991*, 2020.

Hardy Stephen, Henecka Wilko, Ivey-Law Hamish, Nock Richard, Patrini Giorgio, Smith Guillaume, and Thorne Brian. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arxiv preprint arXiv:1711.10677*, 2017.

Li Tian, Sahu Anit Kumar, Zaheer Manzil, Sanjabi Maziar, Talwalkar Ameet, and Smith Virginia. Federated optimization in heterogeneous networks. *arxiv preprint arXiv:1812.06127*, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017. URL https://arxiv.org/abs/1706.03762.

Yang Wensi, Zhang Yuhang, Ye Kejiang, Li Li, and Xu Cheng-Zhong. Ffd: A federated learning based method for credit card fraud detection. *Big Data – BigData 2019. BIGDATA 2019. Lecture Notes in Computer Science(), vol 11514. Springer, Cham. https://doi.org/10.1007/978-3-030-23551-2*$_2$, 2019.