A FEDERATED MODEL FOR INSURANCE CLAIMS FRAUD DETECTION

STEPHEN KATIECHI OKENO[1], DR. ENG. LAWRENCE MUCHEMI[2]

[1]University of Nairobi, School of Computing and Informatics
Email: sokeno@students.uonbi.ac.ke
[2]Univeristy of Nairobi, School of Computing and Informatics
Email: lmuchemi@uonbi.ac.ke

Abstract

Practical insurance fraud detection solutions require sufficient quality data from insurers to build effective models. However, insurance data is generally proprietary information for specific insurance companies and thus not publicly available. Also, the Insurance datasets are often imbalanced, making it challenging to develop fraud detection models that are not biased. Data privacy and class imbalance are two significant challenges when developing artificial intelligence applications in the insurance setup. In this research study, we tackle these challenges and propose a decentralized and privacy-preserving federated approach using an adjusted random forest model. The method is asynchronous federated learning of the traditional adjusted random forest classifier, i.e., achieving a higher performance and accuracy level than the traditional centralized learning approach. Based on it, we achieved secure collaborative machine learning that allows the training of quality federated fraud detection models from imbalanced data without sharing data. Experiments on Kaggle and Oracle insurance datasets demonstrate that the federated adjusted random forest classifier is more accurate and efficient than the non-federated counterpart. Our model is verified to be practical, efficient and scalable for real-life insurance fraud detection tasks.

Keywords: Insurance Fraud Detection, Federated Learning, Adjusted Random Forests

INTRODUCTION

Insurance claims fraud (illegitimate claims), other than tax fraud, is recorded to be the most practiced fraud globally. The significant accumulation of liquid financial assets makes insurance companies susceptible to loot schemes and takeovers (Association of Certified Fraud Examiners, 2019). Insurance claims fraud occurs when the insured attempts to gain profits through premiums paid without complying with the insurance agreement terms (Association of Certified Fraud Examiners, 2019).

Detecting fraud manually has always been costly for insurance companies. Low incidents that go undetected contribute immensely to the claim ratio. For example, the Industry average Incurred claims ratio (loss ratio) is 64.34%, with motor insurance accounting for 24.6% of the total industry paid claims under the general insurance business (Insurance Regulatory Authority, 2020).

The research community has focused on insurance fraud detection methods that require centralized datasets from specific insurers. There is vast body of literature published on fraud detection methods in the Insurance Industry. These methods, however, use insurance data from specific insurers that might not be representative of the industry fraud problem. Feeble attempts have been made to look at fraud detection methods from an industry perspective. The quality of the data needed to train predictive models is as important as the quantity required. Datasets must be representative and balanced to provide a better picture and avoid bias (Rama Devi Burri, et al., 2019).

DATA SCARCITY

Significant studies have been conducted to explore the detection and prevention of Insurance fraud. However, because of the evolving nature of insurance fraud, there still exist challenges due to the lack of sufficient insurance data and a class imbalance problem in claim datasets that have attracted the attention of researchers. Vast amounts of data required for machine learning have created additional risk for insurance companies. The increase in data collection and connectivity among applications can lead to data leaks and security breaches. This makes Insurers struggle to provide relevant data for training machine learning models (Rama Devi Burri, et al., 2019).

CLASS IMBALANCE

Insurance fraud detection problems are often biased because they reduce the overall error rate instead of taking care of minority classes (Johannes & Rajasvaran, 2020). Studies have shown that the lack of primary insurance data and imbalanced datasets is a challenge when developing machine learning models in insurance. Imbalanced datasets often produce biased models that cannot make correct predictions (Johannes & Rajasvaran, 2020). Insurance companies that adopt a centralized approach for insurance claim fraud detection face a class imbalance problem, a case where fraud incidents are less than the total number of claims (R Guha et al., 2017).

FEDERATED MACHINE LEARNING

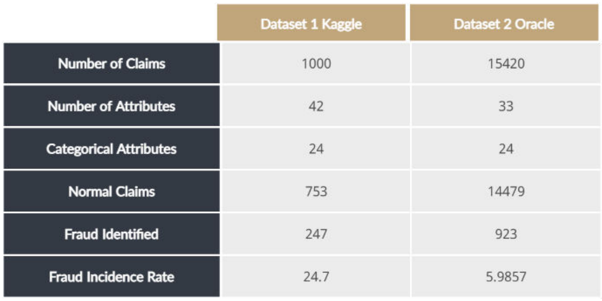
Although research has been done to show that using Federated Learning improves prediction accuracy while preserving the accuracy of the data, the methods suffer from abilities to train federated models asynchronously (McMahan et al., 2017) while eliminating a single source of failure (Ongati & Lawrence, 2019). The methods also present limitations due to tightly coupled nodes and dependencies (Liu et al., 2019). Centralized methods that preserve data privacy, like seen in (Dhieb et al., 2019), require centrally aggregated data and, therefore, not effective methods for asynchronous federated learning. There is limited research to show the effectiveness of these collaborative model training techniques on highly imbalanced datasets and the applications in the insurance industry. Studies present a need to conduct a study that determines whether using decentralized learning will improve prediction accuracy and prove an effective method of insurance claims prediction.

METHODOLOGY

We first collected past insurance claims data for simulations. The collected data was then cleaned and features were selected for training. Next, a federated architecture and algorithm were designed; after that, we carried out Simulations and prototyping, the federated model was trained and evaluated.

Data Collection

The variables used in developing the model for this Due to the unavailability of local primary insurance data, we used the Kaggle dataset (Roshan, 2019) and the Oracle dataset (Charlie, 2010) to train and test our models. The datasets contain many insurance features and columns used in this study; this includes columns containing the insured personal details. The input variables under study include the customer profile with personal details like name, age, location and sex. We also study the risk profile of the property insured, the claim details and policy details of the insured. They include the type of risk, car make and model, year of manufacture, the number of days to the expiry and the sum insured. For all the datasets, a small portion of claims is identified as frauds while others as normal. Some claims marked as normal might be fraudulent, but the suspicions were not followed through because of time delays, late detection, among other reasons. Table 1 below summarizes our datasets:



	Dataset 1 Kaggle	Dataset 2 Oracle
Number of Claims	1000	15420
Number of Attributes	42	33
Categorical Attributes	24	24
Normal Claims	753	14479
Fraud Identified	247	923
Fraud Incidence Rate	24.7	5.9857

Figure 1 Features of various datasets

Feature Selection

Insurance claims fraud detection involves many features that can cause increased dimensionality, multicollinearity and overfitting. Studies have shown that as the number of features increases, the classifier's performance increases to a specific limit of features. Adding more features past the optimal limit degrades the performance of the classifier. The feature selection process selects a subset of features with more information from the original features by removing redundant and irrelevant features without losing information. Because of the many features we have selected to determine whether a claim is fraudulent, as shown in figure 8, we adopted embedded methods (Random Forests) suitable for feature selection in high dimension datasets. Random Forests uses a tree-based strategy that naturally ranks the features by how best they improve the purity of the node. The methods use modelling in their approaches, and hence they account for each feature interaction during training (Johannes & Rajasvaran, 2020).

Federated Adjusted Random Forests

Fraud detection can be considered a classification problem where machine learning models need to classify input features and predict which class the features fall. Suppose we assume that our predicted variable falls in either two classes (fraud or no fraud). In that case, categorical classification techniques cannot be applied, and binary classification techniques solve this problem. In the study by (Dhieb et al., 2019), an extreme gradient boosting algorithm was used to detect the status of a claim for insurance applications as fraudulent, not fraudulent, and other categories. This study presents a federated adjusted random forests algorithm based on the CART tree to deal with the class imbalance problem and make it a practical approach. The adjusted random forest algorithm works as shown below.

i. For each round, we randomly select a sample of fraudulent claims with a replacement that is from the lower class. We also select a similar number of non-fraudulent claims with replacement.

ii. We create a tree with the sample above to a full depth, i.e. without pruning. A split is made on each node should be based on a set of randomly selected features. This makes the correlation between the trees to be lower.

iii. We repeat steps (i) and (ii) for n times and use bootstrapped aggregation to find the final result we use as the final predictor.

The Adjusted Random Forest classifier doesn't over fit imbalanced data because it performs tenfold cross-validations at every iteration level. Figure 8 below illustrated the functionality of adjusted random forest.

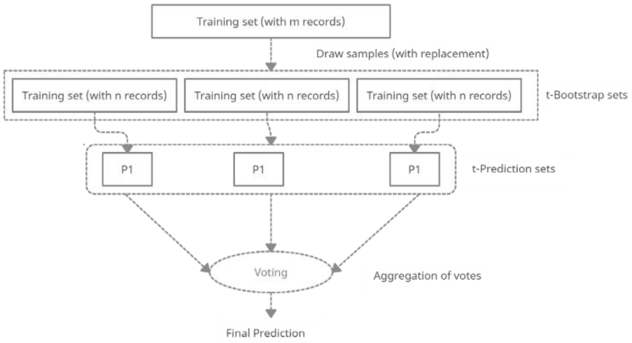

Training set (with m records)
Draw samples (with replacement)
Training set (with n records)
Training set (with n records)
Training set (with n records)
t-Bootstrap sets
P1
P1
P1
t-Prediction sets
Voting
Aggregation of votes
Final Prediction

Figure 2 Bagging Using Adjusted Random Forest

Each client trains an ensemble adjusted random forest classifier on their data. Training on the Individual node happens such that individual models are trained in parallel; a random subset of data trains each model. Thus, the adjusted random federated forest becomes, by default, an ensemble model of bagged federated trees. Figure 9 below shows the framework that is based on the CART tree. The framework can be used for both classification and regression problems.

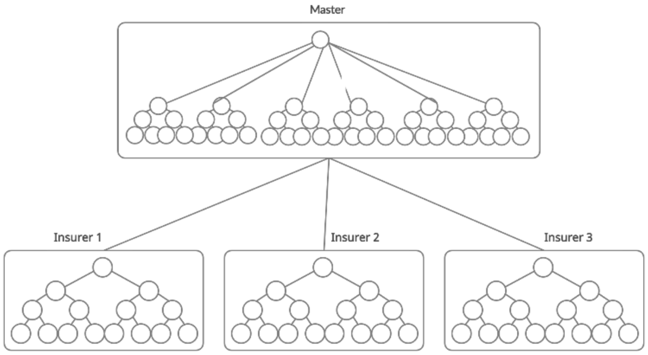


Figure 3 Random Federated Forests

Decentralized Architecture Design

The Insurance Regulatory Authority is a government regulatory agency mandated by law to regulate, supervise and develop the insurance industry (Insurance Regulatory Authority, 2020). In regulating and developing the Industry, IRA was our natural target as a central node as it will ensure fairness in rewarding participating members. Each primary insurer downloads a list of participating features from the central node. They then train their model with their data on their servers and upload them to the Insurance Regulatory Authority centralized node. IRA ensembles the edge models through aggregation to form a standard centralized model and distributes them back to all the primary insurers. This approach will be a simple version of Federated Learning because the number of insurers is expected to be smaller. In addition, the communication bandwidth between enterprise servers is expected to be more than that of mobile phones. The study uses serialization and encryption of the models during transfer to and from the central node to protect privacy for policyholders, protect the insurer’s data, and benefit from the global federated model. Figure 6 below illustrates the federated design that effectively trains a shared model in the Insurance setup.

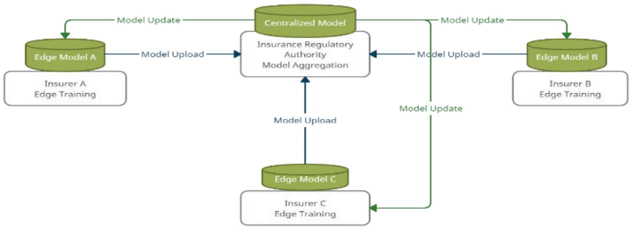


Figure 4 Decentralized Design

Decentralized Algorithm Design

An optimal decentralized algorithm is implemented by improving the asynchronous limitations of the existing federated forest algorithm presented by (Liu et al., 2019). In our work, the forest classifier is built and evaluated for accuracy and efficiency on the node. As opposed to (Liu et al., 2019), all parties don't have to work together to build the federated model making the process asynchronous and therefore eliminating points of failures and problems of multiple dependencies as reported by (Ongati & Lawrence, 2019). After downloading a list of similar features from the central node, the dataset is ready to train a model. The training process is asynchronous, which makes client nodes ensure the confidentiality of their training process. Using Random federated forests as an example, training the model is divided into five steps.

1. Clients A and B downloads the latest list of standard features.
2. Clients A and B perform data preprocessing and analysis to align data to the downloaded features.
3. Clients A and B independently trains their classifier using Adjusted Random forests until they reach a splitting state.
4. Clients A and B upload their serialized models to the central Node C for aggregation.
5. Central node C de-serializes the uploaded models. The central node also aggregates the models by combining the estimators from individual trees.
6. The central node C then distributes the combined model back to the client's nodes and completes a single training cycle.

We repeat the process above when a single client node uploads an updated model; The respective data of A and B is kept locally and not shared. There is no interaction during training; therefore, no sharing of client data.

Implementation and Prototyping

This study used the python programming language to develop model architectures. We also used secure shell protocol (SSH) for communication and transfer of model representations and weights. The environment was set up on personal computers, running on Windows 10 Pro operating system with Python 3.6 installed with NumPy, pandas and sklearn python libraries. First, we loaded data from (Roshan, 2019) with 1000 claim records and got it through the data analysis steps as shown in sub-section 3.5. We assigned the target variable "fraud reported" to be our y and dropped it from the x-axis. Next, we shuffled the data with a random state to reproduce it when the code is rerun. Shuffling helps to balance the data to reduce noise.

We then split our training data into 80% training sets and 20% test/validation sets. Studies have shown that assigning more samples (n) to the training set improves accuracy (Dobbin & Simon, 2011). We repeated the same process with the dataset from (Charlie, 2010). The dataset contains 15420 claim instances between January 1994 to

December 1996 and an average of 430 claims per month. We used embedded methods (Random Forests) suitable for feature selection in high dimension datasets. The methods use modelling in their approaches, and hence they account for each feature interaction during training (Johannes & Rajasvaran, 2020). For each dataset, we recorded metrics before feature selection and after feature selection.

The model development involved classical algorithms as well as our federated algorithms. On each algorithm, we trained and recorded the metrics required to evaluate its performance. The selected algorithms included the classical Random Forest Classifier, adjusted random forest classifier and Extreme Gradient boosting classifier. We created 20 Random forest classifiers and recorded the evaluation metrics for each forest. We also created 20 adjusted random forest classifiers and recorded the evaluation metrics. The last step involved combining individual bagged random forest models into one giant federated model by aggregating its estimators. We tested both the classical and the federated models on the two datasets. We then calculated the results that are considered as the performance criterion and did a comparison.

RESULTS AND DISCUSSION

Classification Report

We examine other metrics like precision, recall, and F1 score to get more insight into our model's performance. Precision is a fraction of members of a class that were correctly identified amongst all members that were predicted to belong in a particular class. A model's recall is a fraction of members who were predicted to belong to a class amongst all of the members that belong to the class. F1-Score combines both precision and recall metrics into one metric. If precision and recall are high, F1-score will be high, and if they are low, the F1-Score will be lower. The figures below give classification reports for the algorithms in this study.

Model Name	**AVG Precision**	**AVG Recall**	**Avg F1-Score**	**Accuracy**
Federated Balanced Random Forest Classifier	0.53	0.53	0.49	0.52
Federated Random Forest Classifier	0.38	0.50	0.43	0.76
Balanced Random Forest Classifier	0.55	0.56	0.53	0.57
Random Forest Classifier	0.50	0.50	0.45	0.74

Table 1 Classification Report Before Feature Selection-Kaggle Dataset

Model Name	**AVG Precision**	**AVG Recall**	**Avg F1-Score**	**Accuracy**
Federated Balanced Random Forest Classifier	0.57	0.79	0.50	0.63
Federated Random Forest Classifier	0.72	0.50	0.49	0.94
Balanced Random Forest Classifier	0.56	0.76	0.51	0.66
Random Forest Classifier	0.80	0.51	0.51	0.94

Table 2 Classification Report Before Feature Selection-Oracle Dataset

Model Name	**AVG Precision**	**AVG Recall**	**Avg F1-Score**	**Accuracy**
Federated Balanced Random Forest Classifier	0.74	0.78	0.75	0.80
Federated Random Forest Classifier	0.66	0.60	0.61	0.76
Balanced Random Forest Classifier	0.72	0.78	0.73	0.77
Random Forest Classifier	0.73	0.63	0.65	0.79

Table 3 Classification Report After Feature Selection-Kaggle Dataset

Model Name	**AVG Precision**	**AVG Recall**	**Avg F1-Score**	**Accuracy**
Federated Balanced Random Forest Classifier	0.57	0.79	0.50	0.63
Federated Random Forest Classifier	0.72	0.50	0.49	0.94
Balanced Random Forest Classifier	0.56	0.76	0.51	0.66
Random Forest Classifier	0.83	0.51	0.51	0.94

Table 4 Classification Report After Feature Selection-Oracle Dataset

Discussion

Many research shows that the quality and quantity of available data significantly impact insurance fraud detection models (Johannes & Rajasvaran, 2020) (R Guha et al., 2017). The federated models performed poorly on the Kaggle dataset (Roshan, 2019). The performance is majorly attributed to the small number of claims (1000) and more significant variance in the classes. The federated models, however, performed well on the oracle dataset (Charlie, 2010). The performance is attributed to a large number of claims (15420). Therefore, the quality and quantity of data at the nodes have significant effects on the overall performance of the federated model. The overall performance of the federated model depends on the performance of the local model.

There is a significant improvement in all algorithms when using the embedded Random Forest feature selection method. The federated adjusted random forest classifier records the highest balance of metrics after feature selection, increasing accuracy to 28% from 52% on the Kaggle dataset. As discussed by (Johannes & Rajasvaran, 2020), Random Forests uses a tree-based strategy that naturally ranks the features by how best they improve the purity of the node, and they, therefore, account for each feature interaction during training. We required additional information from the classification report and confusion matrix to help us understand how well the model performed, how it performed on each class and where it found difficulties to distinguish classes. We note that accuracy is not a great measure of classifiers performance, especially when classes are imbalanced, as is our case.

The confusion matrix shows that the federated random forest classifier performs poorly on imbalanced datasets because it introduces bias. The model is therefore not suitable for insurance claims fraud prediction. Therefore, we cannot use it in a federated setting and collaborative learning. On the other hand, the balanced federated random forest classifier gives an exemplary performance on the imbalanced insurance dataset and outshines its classical counterpart on both datasets. Moreover, the model has a balanced representation of the classification metrics making it an optimal model for collaborative machine learning in detecting insurance claims fraud.

CONCLUSION

This study proposed a federated ensemble and an adjusted tree-based machine learning approach, which is asynchronous, meaning all nodes don’t have to be online for training to occur. The method, therefore, solves the problem of synchronous learning as shown by (Liu et al., 2019) and the challenges of having a central coordinator to manage all the nodes during training as shown by (Ongati & Lawrence, 2019). Furthermore, the federated adjusted random forest model solves the class imbalance problem on the dataset by performing tenfold cross-validation at every iteration level during training as it builds the forest. As a result, the adjusted random forest classifier was able to identify most of the fraudulent cases with a higher precision, F1-score and a lower false-positive rate.

The federated model for insurance claims fraud prediction presented in this research was tested and found efficient for the detection of fraudulent claims in the insurance setup. The federated model allows all players in insurance to collaboratively build efficient fraud detection models without sharing data. The model allows smaller insurance entities to detect claim fraud without sufficient insurance data. Furthermore, the asynchronous federated machine learning approach achieves data privacy by allowing each node to train an ensemble adjusted random forest classifiers. Generally, this solution is an important asset for insurance entities as it helps them reduce the loss ratio.

RECOMMENDATION

We selected IRA as our central node that aggregates models weights. We however recommend a neutral node applicable in the region to be a central node that aggregates the model weights. We also recommend a verification method that will be used to verify the updates before they are aggregated. The method will ensure that poor updates don’t spoil the quality model. The method can verify the model updates by running the update on the test dataset before they are accepted and integrated into the global model. Given the difference in insurance fraud predictors at different geographical locations, we recommend that each implementation of this method adopts a feature engineering and selection method that best suits the fraud scenario in the specific regions.

It is also essential that participants in a collaborative machine learning model do sufficient tests on their local models before uploading, as poor models will affect the overall performance of the federated model. Quality local models can be built by ensuring that sufficient data is aggregated before training.

FUTURE RESEARCH

Given the inherent characteristics of various datasets on different nodes, it would be impractical to recommend an optical technique or a feature engineering methodology that would best perform the model at the node. Therefore, a future study can present a standard feature engineering methodology used with federated learning in the insurance setup. Future research can also focus on a verification method that will be used to verify the updates before they are aggregated. The method will ensure that poor updates don't spoil the quality model. The method can verify the model updates by running the update on the test dataset before they are accepted and integrated into the global model.

REFERENCES

1. Association of Certified Fraud Examiners. (2019). *INSURANCE FRAUD HANDBOOK*. Association of Certified Fraud Examiners, Inc.
2. Burri, R. D., Burri, R., Reddy Bojja, R., & Rao Buruga, S. (2019). Insurance Claim Analysis using Machine Learning Algorithms. *International Journal of Innovative Technology and Exploring Engineering*, *8*(6S4), 577–582. https://doi.org/10.35940/ijitee.F1118.0486S419
3. Charlie, B. (2010, January 18). *Fraud and Anomaly Detection Made Simple*. https://blogs.oracle.com/machinelearning/fraud-and-anomaly-detection-made-simple
4. Dhieb, N., Ghazzai, H., Besbes, H., & Massoud, Y. (2019). Extreme Gradient Boosting Machine Learning Algorithm For Safe Auto Insurance Operations. *2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, 1–5. https://doi.org/10.1109/ICVES.2019.8906396
5. Dobbin, K. K., & Simon, R. M. (2011). Optimally splitting cases for training and testing high dimensional classifiers. *BMC Medical Genomics*, *4*(1), 31. https://doi.org/10.1186/1755-8794-4-31
6. Insurance Regulatory Authority. (2020). *INSURANCE INDUSTRY ANNUAL REPORT 2019* [ANNUAL REPORT]. Insurance Regulatory Authority.
7. Johannes, S. K., & Rajasvaran, L. (2020). AUTO-INSURANCE FRAUD DETECTION: A BEHAVIORAL FEATURE ENGINEERING APPROACH. *Journal of Critical Reviews*, *7*(03). https://doi.org/10.31838/jcr.07.03.23
8. Konečný, J., McMahan, H. B., Ramage, D., & Richtárik, P. (2016). Federated Optimization: Distributed Machine Learning for On-Device Intelligence. *ArXiv:1610.02527 [Cs]*. http://arxiv.org/abs/1610.02527
9. Liu, Y., Liu, Y., Liu, Z., Zhang, J., Meng, C., & Zheng, Y. (2019). Federated Forest. *ArXiv:1905.10053 [Cs, Stat]*. https://doi.org/10.1109/TBDATA.2020.2992755
10. McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & Arcas, B. A. y. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data. *ArXiv:1602.05629 [Cs]*. http://arxiv.org/abs/1602.05629
11. Ongati, F., & Lawrence, M. (2019). *Big Data Intelligence Using Distributed Deep Neural Networks*. 7.
12. Palacio, S. M. (2019). Abnormal Pattern Prediction: Detecting Fraudulent Insurance Property Claims with Semi-Supervised Machine-Learning. *Data Science Journal*, *18*(1), 35. https://doi.org/10.5334/dsj-2019-035
13. Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). *A Comprehensive Survey of Data Mining-based Fraud Detection Research*. 14.

14. R Guha, Shreya Manjunath, & Kartheek Palepu. (2017). Comparative Analysis of Machine Learning Techniques for Detecting Insurance Claims Fraud. *Wipro*, 19.
15. Roshan, S. (2019, March 7). *Insurance Claim*. https://www.kaggle.com/roshansharma/insurance-claim
16. Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated Machine Learning: Concept and Applications. *ArXiv:1902.04885 [Cs]*. http://arxiv.org/abs/1902.04885
17. Zhang, X., Han, Y., Xu, W., & Wang, Q. (2019). HOBA: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture. *Information Sciences*, *557*, 302–316. https://doi.org/10.1016/j.ins.2019.05.023