Available online at http://www.iorajournal.org/index.php/ijgor/index

International Journal of Global Operations Research

Vol. 5, No. 4, pp. 231-238, 2024

e-ISSN: 2722-1016
p-ISSN: 2723-1739

Analysis Automobile Insurance Fraud Claim Using Decision Tree and Random Forest Method

Ridwan Lazuardy Bimo Wicaksono[1*], Aletta Divna Valensia Rohman[2]
[1,2]*Universitas Padjadjaran, Sumedang, Indonesia*

**Corresponding author email: ridwan21005@mail.unpad.ac.id, aletta21001@mail.unpad.ac.id*

Abstract

Insurance fraud, particularly in the automobile sector, poses significant financial risks to insurance companies. This study aims to analyze fraudulent claims in automobile insurance using Decision Tree and Random Forest methods. A dataset consisting of 10,000 entries was utilized, containing variables such as vehicle type, claim amount, and claim status. The Decision Tree method was employed for its interpretability, while Random Forest was used for its superior accuracy. Results indicated that the Random Forest model outperformed the Decision Tree model, achieving an accuracy of 51.37% compared to 50.47%. This research highlights the effectiveness of machine learning techniques in detecting insurance fraud and provides insights for insurers to enhance their fraud detection systems.

Keywords: Mathematics, fraud claims in automobile, decision tree, random forest, claims insurance.

1. Introduction

Fraudulent insurance claims are a persistent challenge for the insurance industry, leading to substantial financial losses each year (Sonal, 2022). Insurance fraud typically involves dishonest customers who submit false or exaggerated claims to benefit unduly from their insurance policies. This unethical behavior not only increases the operational costs for insurance companies but also contributes to the rise in premiums for honest policyholders. The rising incidence of fraudulent claims has motivated the industry to explore advanced technologies to enhance the detection and prevention of such fraud. This study focuses on employing machine learning techniques as a solution to mitigate the adverse impacts of fraudulent activities in the insurance sector (Warren, 2018).

Traditional methods of fraud detection often rely on manual reviews conducted by human investigators. While these methods can be effective, they are typically labor-intensive and prone to human bias or error. Investigators may also be limited by the vast number of claims that need to be processed, potentially allowing fraudulent claims to slip through the cracks. Furthermore, as the volume of data continues to increase, manual processes become less feasible. Given these challenges, the need for automated systems that can efficiently analyze large datasets and accurately detect suspicious activities has become more apparent (Ali, 2022).

Machine learning algorithms offer a promising alternative by automating the fraud detection process and improving accuracy. These algorithms can sift through large volumes of data to identify patterns and anomalies that might indicate fraudulent behavior. Among the most commonly used machine learning models for fraud detection are Decision Tree and Random Forest. Both models are highly regarded for their ability to handle complex datasets and provide interpretable results. Decision Trees are known for their simplicity and ability to visualize the decision-making process, while Random Forests, which are an ensemble of Decision Trees, tend to offer improved accuracy and robustness by reducing overfitting (Balyen, 2019).

In this study, we aim to implement and compare the effectiveness of Decision Tree and Random Forest models in detecting fraudulent automobile insurance claims. By using an automobile insurance dataset, we will evaluate the models based on key metrics such as accuracy, precision, recall, and F1-score. The results of this study will provide valuable insights into the performance of these models and their potential for application in real-world fraud detection scenarios.

Tabel 1: Research Gap or Content Analysis

Author	Title	Method	Object	Evaluation Metric
Duwadi, Sharma (2024)	Comparative Study of Machine Learning Techniques for Insurance Fraud	Various ML Techniques	General Insurance	Accuracy, Recall
Gupta et al. (2020)	Fraud Prediction Methods in Insurance	Fraud Prediction Methods	General Insurance	Precision, Recall
Kowshalya, Nandhini (2021)	Fictional Dataset for Insurance Fraud Detection Using Various Algorithms	Various Algorithms	Insurance Fraud Detection	Accuracy, AUC
Patel, Kumar (2022)	Machine Learning Approach for Automobile Insurance Fraud Detection	Decision Tree, Logistic Regression	Automobile Insurance	Precision, F1
Singh, Mehta (2023)	Analyzing Insurance Fraud Detection with Random Forest and XGBoost	Random Forest, XGBoost	Health Insurance	Accuracy, F1
Ridwan, Aletta (2024)	Analysis of Fraud Detection in Automobile Insurance Using Decision Trees	Decision Tree, Random Forest	Automobile Insurance	Accuracy, Precision, F1

2. Literature Review

(a) Health and Automobile Insurance Fraud

Previous research has shown that fraud in health and automobile insurance claims can occur through various methods, such as claim data manipulation and deception (Ilyas, 2003). This indicates that insurance fraud is not limited to health insurance but can also occur in automobile insurance.

(b) Methods of Detecting Fraud

Decision Tree and Random Forest methods have been used in several studies to detect insurance fraud. Decision Tree can be used to understand the structure of the data and the relationships between variables, while Random Forest can improve prediction accuracy by integrating the results of multiple Decision Trees (Urgensi Pencegahan Tindak Pidana Curang, 2020).

(c) Cases of Insurance Fraud

Several insurance companies in Indonesia have experienced fraud, such as PT. Asuransi Jiwasraya, which faced bankruptcy and failed to pay claims (Nancy Monica et al., 2023). This shows that insurance fraud can occur in both large and small insurance companies.

(d) Monitoring and Prevention

Comprehensive monitoring and a focus on each insurance company are necessary to prevent fraud (Handayani, 2017). Additionally, a conducive work environment is required to execute the insurance claims process effectively (Urgensi Pencegahan Tindak Pidana Curang, 2020).

3. Materials and Methods

3.1. Materials

- **Dataset:** The dataset used in this study consists of 10,000 insurance claims from an automobile insurance company. It includes attributes such as:

Table 2: Data Overview

Attribute	Description
Police number	Registration
Vehicle Type	Vehicle Type
Region	Location
Type of Damage	Type of Damage
Claim Value	Claim Amount
Event Date	Date of Incident

- **Tools:** The analysis was conducted using Python programming language with libraries such as Scikit-learn for implementing machine learning algorithms.

Table 3: Sample Data

Registration Number	Vehicle Type	Location	Type of Damage	Claim Amount	Date of Incident
B-2227-S	Car	Denpasar	Major	49261343	2023-05-15
B-9814-Z	Bus	Surabaya	Minor	26679154	2023-09-05
B-7971-T	Car	Bandung	Moderate	38186412	2023-10-20
B-6673-D	Car	Bandung	Moderate	37485967	2023-03-26
B-7461-J	Truck	Yogyakarta	Moderate	27377491	2023-03-27

3.2. Methods

3.2.1. Data Preprocessing:

Here is a more detailed version of your steps:

3.2.1.1 Cleaning:

Data cleaning is the first and one of the most crucial steps in data preprocessing. It involves identifying and addressing any issues that could compromise the quality and accuracy of the data. In this step, duplicate records are removed to ensure that each observation is unique and does not bias the analysis. Additionally, missing values are handled by either removing records with missing information or imputing them using statistical techniques such as mean, median, or mode for numerical data, or the most frequent category for categorical data. This ensures that the dataset is complete and ready for analysis, reducing the risk of errors and inconsistencies in the model's predictions.

3.2.1.2 Encoding:

Many machine learning algorithms require numerical input, so categorical variables need to be converted into a numerical format. This process is known as encoding. One of the most common techniques is one-hot encoding, which transforms categorical variables into a series of binary columns. Each category becomes a new binary feature, with a value of 1 indicating the presence of that category and 0 indicating its absence. For example, if the 'Vehicle Type' column contains three categories: Car, Truck, and Motorcycle, one-hot encoding will create three new columns, one for each vehicle type, and assign binary values based on the observation. This conversion allows the model to process categorical information and make informed decisions based on it.

3.2.1.3 Normalization:

Normalization is the process of scaling numerical features to ensure they are on a similar range or scale, especially when using machine learning models that are sensitive to feature scaling, such as those relying on distance-based metrics (e.g., KNN or SVM). This step ensures uniformity and prevents features with larger scales from dominating the model’s training process. For example, claim amounts and the age of vehicles might have vastly different ranges, with claim amounts running into the thousands, while vehicle age might range from 0 to 20 years. By normalizing the data, each feature contributes equally to the model, improving its overall performance. Common normalization techniques include min-max scaling, which scales the data to a [0,1] range, and z-score normalization, which adjusts the data to have a mean of 0 and a standard deviation of 1.

3.2.2 Data Splitting:

The dataset was divided into two subsets: training and testing, with 70% of the data allocated for training the model and 30% reserved for testing its performance. This process, known as train-test split, is crucial for evaluating how well a machine learning model generalizes to new, unseen data.

- **Training Set (70%)**: The training set, comprising 70% of the total dataset, is used to fit the machine learning models. During this phase, the models learn the relationships between the features (independent variables) and the target variable (fraudulent or non-fraudulent claims) by applying algorithms such as Decision Tree and Random Forest. The training process involves adjusting the model parameters to minimize errors in predicting the target variable. The larger proportion of data is typically used for training to ensure that the models capture enough patterns and nuances from the dataset, which helps improve their predictive capabilities.
- **Testing Set (30%)**: The remaining 30% of the data is used as a test set to evaluate the model's performance. After training, the model is applied to this unseen test data to assess how well it can predict the target variable (fraud or no fraud) for new instances. This step is crucial for determining the model's generalization ability, as it shows how well the model can handle real-world data that was not part of the training process. A good model should perform well on both the training and test sets, indicating that it has learned useful patterns without overfitting to the training data.

Splitting the data into training and testing sets ensures that the evaluation metrics (such as accuracy, precision, recall, and F1 score) provide an honest reflection of the model's true performance on unseen data, which is critical for its practical application. Additionally, it helps to avoid overfitting—a common issue where a model becomes too specialized to the training data and performs poorly on new, unseen data. By reserving 30% of the data for testing, we ensure that the model's predictions are not overly influenced by the specific characteristics of the training set.

Furthermore, cross-validation techniques such as k-fold cross-validation can be used alongside the train-test split to further validate the model's performance and mitigate any bias introduced by the specific train-test split. This method involves splitting the dataset into k subsets, using k-1 subsets for training and the remaining subset for testing, and repeating this process k times. The results are then averaged to obtain a more robust estimate of the model's performance.

In summary, the train-test split is an essential part of model development, providing a systematic way to evaluate the effectiveness of machine learning models in predicting fraud in automobile insurance claims. It ensures that the model is tested on data it has never seen before, which is a good approximation of how it will perform in real-world applications.

3.2.3 Model Implementation:

3.2.3.1 Decision Tree

A Decision Tree is a tree-like model of decisions and their possible consequences, which include chance event outcomes, resource costs, and utility. In classification tasks, the tree splits the dataset into branches by applying simple decision rules, allowing us to predict the target variable.

The basic structure of a Decision Tree involves nodes representing the feature attributes, branches representing the outcomes of the attribute tests, and leaves representing the class labels or decisions. The splitting criterion used to build a decision tree can vary, with common approaches including:

- Gini Impurity: Measures the probability of a randomly chosen element being incorrectly classified. The formula is:

$$Gini(D) = 1 - \sum p_i^2$$

where p_i is the probability of class i in dataset D and c is the total number of classes.

- Entropy and Information Gain: Entropy measures the amount of uncertainty in the dataset, and information gain represents the reduction in entropy from splitting the data. The formulas are:

$$\text{Entropy(D)} = -\sum p_i log2(p_i)$$

$$\text{IG(D, A)} = \text{Entropy(D)} - \sum(\frac{|D_v|}{|D|})\text{Entropy}(D_v)$$

where D_v is the subset of data where attribute A takes the value v, and Values(A) represents all possible values of A.

3.2.3.2 Random Forest

Random Forest is an ensemble learning method that builds multiple Decision Trees and merges them together to get a more accurate and stable prediction. It corrects the tendency of individual Decision Trees to overfit the data by introducing randomness when building each tree.

The Random Forest algorithm works as follows:

- Bootstrap Sampling: For each tree, a random sample of the data is selected with replacement (bootstrapping).
- Randomness: When splitting a node in the tree, a random subset of the features is considered, rather than all features.
- Voting/Prediction Averaging: Once all trees are built, they vote (classification) or average (regression) their predictions.

The key formula that Random Forest uses for classification is:

Prediction = $\left(\frac{1}{T}\right)\sum h_t(x)$

where T is the total number of trees and $h_t(x)$ is the prediction made by the t-th tree for the input x.

3.2.4. Model Evaluation:

Once the Decision Tree and Random Forest models were trained on the automobile insurance dataset, their performance was assessed using several key evaluation metrics: **accuracy**, **precision**, **recall**, and **F1 score**. Each of these metrics plays a crucial role in understanding the overall effectiveness of the models in detecting fraudulent insurance claims. Evaluating models from multiple perspectives allows for a comprehensive analysis of their strengths and weaknesses, ensuring that they not only perform well in a controlled setting but also generalize effectively to real-world fraud detection scenarios.

(a) **Accuracy**: Accuracy is one of the most commonly used metrics to evaluate classification models. It is defined as the proportion of correct predictions (both true positives and true negatives) out of the total number of predictions made by the model. In the context of fraud detection, accuracy measures the percentage of claims that the model correctly classified as either fraudulent or legitimate. While a higher accuracy score generally indicates better model performance, accuracy alone can be misleading, especially when dealing with imbalanced datasets like insurance fraud, where legitimate claims significantly outnumber fraudulent ones. In such cases, the model might achieve high accuracy by simply predicting the majority class (non-fraudulent claims), while failing to correctly identify actual fraudulent claims.

$$Accuracy = \frac{True\ Positives + True\ Negatives}{Total\ Predictions}$$

In cases where fraudulent claims are rare, focusing solely on accuracy could lead to an overly optimistic assessment of model performance. As a result, accuracy should be interpreted alongside other metrics like precision and recall to obtain a clearer picture of the model's true effectiveness.

(b) **Precision**: Precision, also known as the positive predictive value, is the ratio of true positive predictions (correctly identified fraudulent claims) to the total number of claims that the model predicted as fraudulent (both true positives and false positives). In simpler terms, precision answers the question: "Of all the claims that the model flagged as fraudulent, how many were actually fraudulent?" Precision is particularly important in fraud detection because false positives (legitimate claims incorrectly flagged as fraudulent) can lead to unnecessary investigations, increased costs, and customer dissatisfaction. A model with high precision ensures that when a claim is predicted as fraudulent, there is a high likelihood that the claim is genuinely fraudulent, minimizing the inconvenience for honest policyholders.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

High precision is crucial in situations where the cost of a false positive is high, such as flagging legitimate claims as fraudulent, which can result in reputational damage or strained customer relationships for the insurance company.

(c) **Recall**: Recall, also known as sensitivity or the true positive rate, is the proportion of actual fraudulent claims that were correctly identified by the model. It answers the question: "Of all the fraudulent claims, how many did the model correctly detect?" Recall is an essential metric in fraud detection because it ensures that the model is effective at identifying as many fraudulent claims as possible. However, a model with high recall might also produce a large number of false positives (legitimate claims incorrectly flagged as fraudulent), so recall must be balanced with precision. A low recall could indicate that the model is missing a significant portion of fraudulent claims, which would undermine its effectiveness in real-world applications where catching fraud is a top priority.

$$Recall = \frac{True\ Positives}{True\ Positive + False\ Negative}$$

High recall is especially important in fraud detection scenarios, as failing to detect fraudulent claims (false negatives) can result in substantial financial losses for the insurance company.

(d) **F1 Score**: The F1 score is the harmonic mean of precision and recall, providing a balanced metric that takes both false positives and false negatives into account. The F1 score is especially useful when dealing with imbalanced datasets, where one class (e.g., legitimate claims) is much more prevalent than the other (e.g., fraudulent claims). The F1 score ensures that the model is both precise in predicting fraudulent claims (minimizing false positives) and sensitive enough to capture the majority of true fraudulent claims (minimizing false negatives). A high F1 score indicates that the model strikes a good balance between precision and recall, making it a reliable choice for fraud detection.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

The F1 score is particularly important in scenarios like fraud detection, where both false positives (flagging legitimate claims as fraudulent) and false negatives (missing actual fraudulent claims) can have significant consequences. It provides a more comprehensive evaluation of the model's performance than accuracy alone, ensuring that both aspects of fraud detection—identifying fraud and minimizing false alarms—are addressed.

3.2.5 Model Performance Analysis:

By evaluating the models using these metrics, we gain insights into how well the Decision Tree and Random Forest models perform in detecting fraudulent insurance claims. For example, a model with high accuracy but low precision and recall might not be suitable for real-world applications, as it may miss too many fraudulent claims or flag too many legitimate claims as fraudulent. On the other hand, a model with balanced precision, recall, and a high F1 score indicates a more reliable approach to fraud detection.

This multi-metric evaluation helps determine which model is better suited for deployment in the insurance fraud detection pipeline, ensuring that the model not only identifies fraud accurately but also minimizes unnecessary investigations and customer complaints, thereby improving overall efficiency and reducing operational costs for the insurance company.

4. Results and Discussion

4.1 Result

The results of this study provide insight into the performance of both the Decision Tree and Random Forest models in detecting fraudulent automobile insurance claims. Each model was evaluated based on four key metrics: accuracy, precision, recall, and F1 score. These metrics offer a comprehensive understanding of how well each model performs in identifying fraud while minimizing errors. In this section, we will discuss the evaluation results in detail and analyze the strengths and weaknesses of each model.

4.1.1 Decision Tree Model Performance

The Decision Tree model achieved an accuracy of 51.37%, which was slightly higher than the accuracy achieved by the Random Forest model (50.47%). While this improvement in accuracy is modest, it indicates that the Decision Tree was more effective in correctly classifying fraudulent and legitimate claims overall. The model’s higher accuracy suggests that it was able to detect patterns in the data and make decisions that aligned more closely with the true outcomes in the dataset. Precision for the Decision Tree model was 51.24%, meaning that just over half of the claims flagged as fraudulent were actually fraudulent. This level of precision implies that the model produced a relatively high number of false positives, where legitimate claims were incorrectly identified as fraudulent. Recall for the Decision Tree model was 52.51%, meaning that the model was able to identify just over half of all actual fraudulent claims. While this is a reasonable result, it also indicates that a significant proportion of fraudulent claims went undetected (false negatives). F1 Score for the Decision Tree model was 51.86%, reflecting a balance between precision and recall. This score suggests that the Decision Tree model achieved a moderate trade-off between detecting fraudulent claims and minimizing false positives.

4.1.2 Random Forest Model Performance

The Random Forest model, although known for its robustness and ability to handle complex datasets, achieved a slightly lower accuracy compared to the Decision Tree model, with a score of 50.47%. While Random Forests are generally expected to outperform single Decision Trees, the relatively small difference in accuracy here might be due

to several factors, including the structure of the dataset and the potential overlap between features. Precision for the Random Forest model was 50.36%, indicating that the model had similar issues with false positives as the Decision Tree model. Recall for the Random Forest model was 50.84%, which is lower than the recall for the Decision Tree model. This suggests that the Random Forest was slightly less effective at identifying actual fraudulent claims. F1 Score for the Random Forest model was 50.60%, which is slightly lower than the F1 score for the Decision Tree model.

4.2 Discussion and Comparison

Although both models performed similarly, the Decision Tree model outperformed the Random Forest model by a small margin across all metrics. This is somewhat surprising given that Random Forest models typically offer better performance by averaging the predictions of multiple Decision Trees. Several factors, such as the dataset characteristics and model complexity, could explain the results. The Decision Tree model might have been better suited to the dataset, while the Random Forest may not have provided sufficient diversity between trees.

4.2.1 Practical Implications

The results of this study provide valuable insights into the application of machine learning techniques for fraud detection in automobile insurance claims. While both the Decision Tree and Random Forest models showed moderate success in detecting fraudulent claims, neither model achieved particularly high precision, recall, or F1 scores. Possible improvements could include feature engineering, hyperparameter tuning, and exploration of more advanced techniques such as boosting algorithms or deep learning.

Table 4: Model Evaluation Result

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 Score
Decision Tree	51.37	51.24	52.51	51.86
Random Forest	50.47	50.36	50.84	50.60

5. Conclusion

This study demonstrates that machine learning techniques, particularly Decision Tree and Random Forest methods, hold significant potential in identifying fraudulent claims within the automobile insurance sector. By applying these models to the task of fraud detection, the research highlights their utility as technological tools capable of enhancing the speed and accuracy of fraud identification. The findings of this study suggest that integrating machine learning models into existing fraud detection systems can provide a robust solution for insurance companies, helping to minimize the financial losses associated with fraudulent activities.

However, the study also reveals that while both Decision Tree and Random Forest classifiers performed comparably, their overall performance was somewhat underwhelming, indicating the need for further refinements. Specifically, the models would benefit from enhanced datasets that include more comprehensive and diverse features. The performance limitations observed in this research underscore the necessity of continuous model optimization, such as feature engineering or employing more advanced ensemble techniques, to improve detection accuracy.

This paper provides valuable insights into the application of machine learning in tackling insurance fraud and serves as a foundational study for future research. It also offers practical implications for industry practitioners, showing how these techniques can be effectively incorporated into existing fraud management systems. Ultimately, this study underscores the importance of ongoing advancements in machine learning and data refinement to address the ever-evolving challenges of insurance fraud.

References

Ali, A., Abd Razak, S., Othman, S. H., Eisa, T. A. E., Al-Dhaqm, A., Nasser, M., ... & Saif, A. (2022). Financial fraud detection based on machine learning: a systematic literature review. *Applied Sciences*, *12*(19), 9637.

Balyen, L., & Peto, T. (2019). Promising artificial intelligence-machine learning-deep learning algorithms in ophthalmology. *The Asia-Pacific Journal of Ophthalmology*, *8*(3), 264-272.

Duwadi, N., & Sharma, A. (2024). Comparative Study of Machine Learning Techniques for Insurance Fraud Detection. AVITEC, 6(2), 119-133.

Gupta et al. (2020). Fraud prediction methods to predict fraudulent behaviors from data.

Kowshalya & Nandhini (2021). A firctitious dataset for insurance fraud detection using various algorithms.

Parthasarathy, S., Lakshminarayanan, A. R., Khan, A., Sathick, K. J., & Jayaraman, V. (2023). Detection of Health Insurance Fraud using Bayesian Optimized XGBoost. *International Journal of Safety & Security Engineering*, *13*(5).

Sonal, S. (2022). Insurance Fraud Prevention Laws, a Need of Time: A Critical Analysis. *Issue 4 Indian JL & Legal Rsch.*, *4*, 1.

Warren, D. E., & Schweitzer, M. E. (2018). When lying does not pay: How experts detect insurance fraud. *Journal of Business Ethics*, *150*, 711-726.