WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Hybrid Machine Learning Model for Vehicle Insurance Fraud Detection

Ateeq Ahmed [1], Mirza Abdul Rahman Baig [2], Mohd Abdul Zubair [3], Mohammed Ashraf Ali [4]

Department of IT, Nawab Shah Alam Khan College of Engineering and Technology, Hyderabad, India
Email md7786955@gmail.com@gmail.com

A B S T R A C T:

This project presents a machine learning-based system to detect fraudulent vehicle insurance claims. Leveraging supervised and unsupervised models such as Random Forest, SVM, AdaBoost, and ensemble techniques, the system analyzes claim data—including demographics, vehicle info, and accident details—to identify potential fraud. Addressing challenges such as class imbalance and data preprocessing, the solution is built using Python and integrates real-time fraud detection with a web interface. The proposed model improves detection accuracy, reduces manual effort, and supports secure, efficient insurance claim processing.

Keywords: Insurance Fraud Detection, Machine Learning, Ensemble Learning, AdaBoost, Real-time Prediction, SMOTE, Claim Classification, Data Preprocessing.

1. Introduction:

Insurance fraud poses a significant threat to the financial stability of the insurance industry. With increasing sophistication in fraud techniques, traditional detection mechanisms such as rule-based systems and manual review are proving inadequate. This project proposes a machine learning-based system that uses supervised and unsupervised learning techniques to automatically detect fraudulent vehicle insurance claims in real-time. By analyzing demographic details, accident characteristics, and policy information, the model flags suspicious claims and helps insurance providers prevent financial losses.

2. Literature Review:

Research has demonstrated the effectiveness of machine learning models in detecting fraudulent patterns in insurance data. For example,

• ***Soham Shah and Shrutee Phadke (2021)***
applied decision trees and support vector machines (SVM) to historical claim data, achieving promising accuracy. In another study,

• ***Aslam et al. (2022)***
highlighted the benefits of ensemble models in reducing false positives.

• ***Benedek et al. (2022)***
emphasized the use of hybrid and cost-sensitive models for robust fraud detection. These findings support the adoption of advanced algorithms and hybrid architectures in developing intelligent, scalable fraud detection systems.

3. Methodology:

3.1 Setting Up the Environment

The system is developed using Python with libraries like NumPy, Pandas, Scikit-learn, TensorFlow, and Flask.

3.2 Dataset Collection and Preparation

A labeled dataset of vehicle insurance claims, including both fraudulent and legitimate instances, is collected.

3.3 Feature Extraction and Preprocessing

The dataset undergoes preprocessing steps such as label encoding, normalization, and outlier removal.

3.4 Model Architecture

The proposed system integrates models like AdaBoost, Random Forest, and SVM in a hybrid ensemble framework.

3.5 Model Training and Evaluation

Models are trained using training datasets and evaluated on test datasets using metrics such as accuracy, precision, recall, and F1-score.

3.6 Real-time Prediction

The trained model is integrated into a web-based interface for real-time fraud detection and user interaction.

4. Illustrations:

Below is the core implementation code used in our hybrid fraud detection system using Random Forest:

```
# Import necessary libraries
import pandas as pd
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.impute import SimpleImputer
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset
data = pd.read_csv('claims_data.csv')

# Handle missing values
imputer = SimpleImputer(strategy='mean')
data['ClaimAmount'] = imputer.fit_transform(data[['ClaimAmount']])

# Encode categorical features
label_encoder = LabelEncoder()
data['ClaimType'] = label_encoder.fit_transform(data['ClaimType'])

# Normalize numerical features
scaler = StandardScaler()
data[['ClaimAmount', 'Age']] = scaler.fit_transform(data[['ClaimAmount', 'Age']])

# Feature engineering
data['ClaimToIncomeRatio'] = data['ClaimAmount'] / data['Income']
data['PreviousClaims'] = data['PreviousClaims'].apply(lambda x: 1 if x > 0 else 0)

# Exploratory Data Analysis (EDA)
sns.countplot(x='Fraudulent', data=data)
plt.title('Distribution of Fraudulent vs. Legitimate Claims')
plt.show()

correlation_matrix = data.corr()
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```

```
sns.pairplot(data, hue='Fraudulent')
plt.show()

# Features and target
X = data.drop('Fraudulent', axis=1)
y = data['Fraudulent']

# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train model
rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)

# Predictions and evaluation
y_pred = rf_model.predict(X_test)
print(f'Accuracy: {accuracy_score(y_test, y_pred)}')
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))

# Hyperparameter tuning
param_grid = {
   'n_estimators': [100, 200, 300],
   'max_depth': [10, 20, None],
   'min_samples_split': [2, 5, 10],
   'min_samples_leaf': [1, 2, 4]
}
grid_search = GridSearchCV(estimator=rf_model, param_grid=param_grid, cv=3, n_jobs=-1, verbose=2)
grid_search.fit(X_train, y_train)

print(f"Best Parameters: {grid_search.best_params_}")
best_rf_model = grid_search.best_estimator_
y_pred_best = best_rf_model.predict(X_test)
print(f'Improved Accuracy: {accuracy_score(y_test, y_pred_best)}')

# Function for real-time detection
def detect_fraud(new_claim_data):
   new_claim_data = imputer.transform(new_claim_data[['ClaimAmount']])
   new_claim_data[['ClaimAmount', 'Age']] = scaler.transform(new_claim_data[['ClaimAmount', 'Age']])
   new_claim_data['ClaimToIncomeRatio'] = new_claim_data['ClaimAmount'] / new_claim_data['Income']
   new_claim_data['PreviousClaims'] = new_claim_data['PreviousClaims'].apply(lambda x: 1 if x > 0 else 0)
   return best_rf_model.predict(new_claim_data)
```

5. Result:

The ensemble model showed superior accuracy in classifying fraudulent claims, with AdaBoost achieving the highest F1-score. Using techniques like SMOTE improved performance on imbalanced datasets. Real-time tests confirmed the system's ability to flag suspicious claims efficiently.

6. Requirements:

6.1. Hardware Requirements

- Processor : Intel i5 (minimum), Intel i7 (recommended)
- Ram : Min 8 GB
- Storage : 500 GB HDD (min), 1 TB SSD (recommended)
- GPU : NVIDIA GTX 1060 or higher (recommended for deep learning)

- Network : 1 Gbps Ethernet (min), Wi-Fi 6 or 10 Gbps Ethernet (recommended)

6.2. Software Requirements

- Operating System : Windows 10 or Ubuntu 18.04+ (min), Ubuntu 20.04 or Windows Server 2019
- Languages : Python (main), R/Java
- Libraries : Scikit-learn, TensorFlow,
- Framework : Flask
- Database : SQL or NoSQL (MongoDB)
- Visualization Tools : Matplotlib, Seaborn, Plotly, Tableau or Power BI

7. Conclusion:

This paper presents a comprehensive approach to detecting botnet attacks in IoT environments using a hybrid deep learning architecture. By integrating ANN, CNN, RNN, and LSTM models, the system effectively analyses network traffic patterns to identify malicious activities in real time. The project demonstrates the practical viability of applying advanced machine learning techniques to enhance IoT security. Through extensive evaluation using the UNSW-NB15 dataset, the proposed model achieves high accuracy and reliability while maintaining scalability for deployment in real-world networks. This work underscores the critical role of intelligent, automated detection systems in protecting IoT infrastructure from evolving cyber threats.

REFERENCES

[1] A. A. Khalil, Z. Liu, and A. A. Ali, "Using an adaptive network-based fuzzy inference system 25, model to predict the loss ratio of petroleum insurance in Egypt," *Risk Management and Insurance Review*, vol. no. 1, pp. 5–18, 2022, doi: 10.1111/rmir.12200.

[2] C. Bockel-Rickermann, T. Verdonck, and W. Verbeke, "Fraud analytics: A decade of research: Organizing challenges and solutions in the field,"*Expert Syst Appl*, vol. 232, p. 120605, 2023, doi: https://doi.org/10.1016/j.eswa.2023.120605.

[3] Y. Wang and W. Xu, "Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud," *Decis Support Syst,* vol. 105, pp. 87–95, 2018, https://doi.org/10.1016/j.dss.2017.11.001.

[4] B. Itri, Y. Mohamed, Q. Mohammed, and B. Omar, "Performance comparative study of machine learning algorithms for automobile insurance fraud detection," in 2019 *Third International Conference on Intelligent Computing in Data Sciences (ICDS)*, 2019, pp. 1–4. 10.1109/ICDS47004.2019.8942277.

[5] R. P. B. Piovezan, P. P. de Andrade Junior, and S. L. Ávila, "Machine Learning Method for Return Direction Forecast of Exchange Traded Funds (ETFs) Using Classification and Regression Models," *Comput Econ, 2023*, doi: 10.1007/s10614023-10385-4.

[6] A. A. Khalil, Z. Liu, A. Salah, A. Fathalla, and A. Ali, "Predicting Insolvency of Insurance Companies in Egyptian Market Using Bagging and Boosting Ensemble Techniques," *IEEE Access***,** vol. 10, pp. 117304–117314, 2022, 10.1109/ACCESS.2022.3210032.

[7] N. Boodhun and M. Jayabalan, "Risk prediction in life insurance industry using supervised learning algorithms," *Complex & Intelligent Systems*, vol. 4, no. 2, pp. 145–154, 2018, doi: 10.1007/s40747-0180072-1.

[8] D. Tiwari, B. Nagpal, B. S. Bhati, A. Mishra, and M. Kumar, "A systematic review of social network sentiment analysis with comparative study of ensemble-based techniques," *Artif Intell Rev*, vol. 56, no. 11, pp. 13407–13461, 2023, doi: 10.1007/s10462-023-10472-w.

[9] M. Liao, S. Tian, Y. Zhang, G. Hua, W. Zou, and X. Li, "PDA: Progressive Domain Adaptation for Semantic Segmentation," *Knowl Based Syst,* vol. 284, p. 111179, 2024, https://doi.org/10.1016/j.knosys.2023.111179.

[10] A. Khalil, Z. Liu, and A. Ali, "Precision in Insurance Forecasting: Enhancing Potential with Ensemble and Combination Models based on the Adaptive Neuro Fuzzy Inference System in the Egyptian Insurance Industry," *Applied Artificial Intelligence,* vol. 38, no. 1, p. 2348413, Dec. 10.1080/08839514.2024.2348413. 2024