NIYAM IT™

Fraudulent Insurance Claim Detection System

Using AI and Machine Learning to Detect and Prevent Fraud

Overview

In this paper, empirical data are analyzed to design statistical models and neural networks to predict fraudulent applications by both applicants and healthcare providers. Behavioral paterns are explored to improve model accuracy.

Introduction

Healthcare and medical insurance is a rich area for fraud schemes due to a complex and bureaucratic process, which requires many approvals, verifications, and other paperwork. The most common scams are fake claims that use false or invalid social security numbers, claims duplication, billing for medically unnecessary tests, fake diagnosis, etc. Both hospitals and insurance companies are suffering from these issues. Insurance carriers lose money and hospitals take risks being involved in serious crimes, like drug turnover. Multiple data analytics approaches can mitigate such fraud risks.

The machine learning approach to fraud detection has received a lot of publicity in recent years and shifted industry interest from rule-based fraud detection systems to ML-based solutions.

The Rule-Based Approach

Detecting fraudulent activities is nuanced, but can be achieved by algorithmically observable signals. Unusually large claims or claims occurring in atypical locations warrant additional verification. Purely rule-based systems entail using algorithms that perform several fraud detection scenarios, manually written by fraud analysts. Today, legacy systems apply about 300 different rules on average to approve a claim. That is why rule-based systems remain too straightforward. They require adding/adjusting scenarios manually and can hardly detect implicit correlations. On top of that, rule-based systems often use legacy software that can hardly process the real-time data streams that are critical for the digital space.

ML-Based Fraud Detection

However, there are also subtle and hidden events in user behavior that may not be evident, but still signal possible fraud. Machine learning allows for creating algorithms that process large datasets with many variables and help end these hidden correlations between user behavior and the likelihood of fraudulent actions. Another strength of machine learning systems, as compared to rule-based systems, is faster data processing and less manual work. ML based methods also do not require a high level of domain knowledge to define rules.

Data and Methodology

Anomaly detection is a common data science approach for fraud detection. It is based on classifying all objects in the available data into two groups: normal distribution and outliers. Outliers, in this case, are the objects (e.g. claims) that deviate from normal ones and are considered potentially fraudulent.

The variables in data that can be used for fraud detection are numerous. By analyzing these parameters, anomaly detection algorithms can answer the following questions:

1. Do clients **access services** in an expected way?
2. Are **user actions** normal?
3. Are **claims** typical?
4. Are there any **inconsistencies** in the information provided by users?

Advanced systems are not limited to finding anomalies but, in many cases, can recognize existing patterns that signal specific fraud scenarios. We will be designing two types of machine learning approaches commonly used in anti-fraud systems: unsupervised and supervised machine learning.

Supervised learning entails training an algorithm using labeled historical data. In this case, existing datasets already have target variables marked, and the goal of training is to make the system predict these variables in future data. Unsupervised learning models process unlabeled data and classify it into different clusters detecting hidden relations between variables in data items.

We pulled in data from four different datasets:

1. **Beneficiary data.** This dataset contains all the

information about the beneficiary and their personal and insurance details.

2. **Inpatient data.** This dataset consists of beneficiaries that were admitted to a hospital or clinic for treatment.
3. **Outpatient data.** This dataset consists of beneficiaries that were not admitted to a hospital/clinic and received treatment/advice during visitation.
4. **Flag data.** This dataset consists of two fields only. One, lists the Provider ID and the second lists whether or not the claims filed by these providers were detected or fraudulent or not.

Models & Results: Supervised Fraud Detection Methods

Logistic Regression

A Logistic Regression (LR) model is used to model the probability of a certain class or event existing - in the context of this paper, claims being fraudulent or non-fraudulent. LR is the utilization of this statistical function to model a binary dependent variable.

According to Figure 1, our LR model seems to perform virtually similarly on the train and validation set across the dataset. By our exploratory data analysis, we determined that the dataset is an imbalanced one. In such a case, we shall use evaluation metrics such as AUROC (Area Under Receiver Operating Characteristic) curve and F-1 scores to get a good overview of how the model is doing.

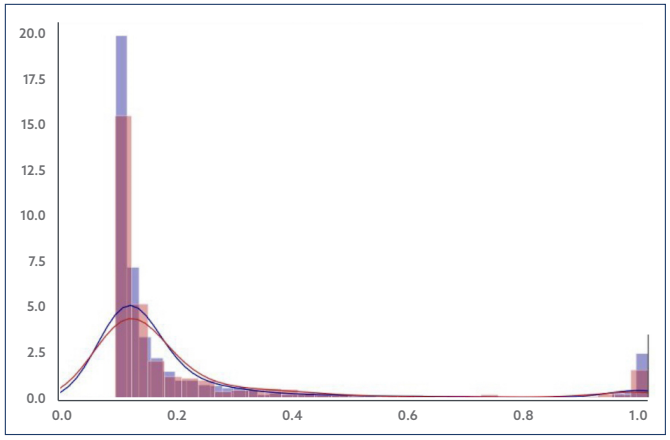

Figure 1: Predictions of Train and Validation

The area under the curve for our LR model gives us a classification accuracy of 94%. Getting into the specifics of how the model is classifying and misclassifying, we scrutinize the True Positive rate (TPR) vs. False Positive rate (FPR).

Based on Figure 2, we conclude that our LR model is performing well to classify True Positives (TR) and True Negatives (TN). However, there are a few instances where TRs and TNs are falsely classified. The F-1 score achieved from the LR model is 0.59 on the validation set.

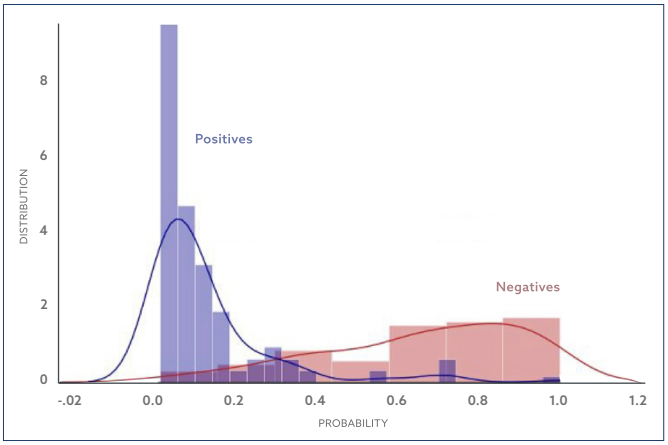

Figure 2: TPR vs. FPR, Positives and Negatives

Random Forest

Random Forests (RF) are an ensemble learning method for classification. They operate by constructing a multitude of decision trees at training time and outputting the class that has been "voted" by the decision trees most frequently.

The area under the curve for our RF model gives us a classification accuracy of 93%. Getting into the specifics of how the model is distinguishing between the classes, we scrutinize the TPR vs. FPR. Looking at Figure 3, we conclude that our LR model is performing well to classify TR and TN. However, there are more instances where TRs and TNs are falsely classified than in the case of our LR model.

The Confusion Matrix confirms our readings from the TP vs. FP. The F-1 score achieved by the Random Forest model was 0.58 on the validation set.

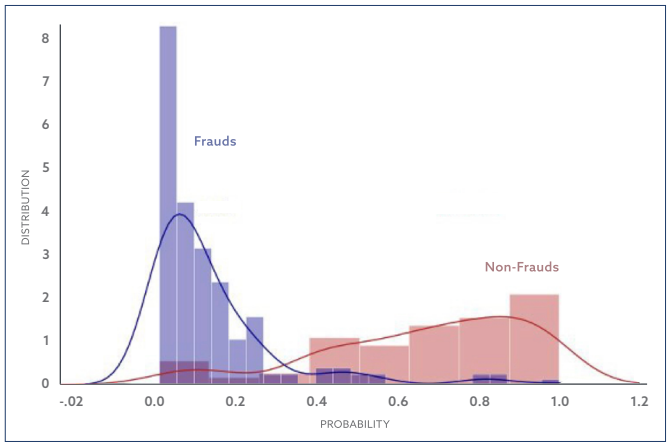

Figure 3: TPR vs. FPR, Frauds and Non-Frauds

Models & Results: Unsupervised Fraud Detection Method

Autoencoder

An autoencoder is a type of artificial neural network used to learn efficient data coding in an unsupervised manner. The aim of an autoencoder is to learn a representation (encoding) for a set of data, typically for dimensionality reduction. Along with the reduction side, a reconstructing side is learned, where the autoencoder tries to generate from the reduced encoding a representation as close as possible to its original input (hence its name). In order to increase the efficiency of the network, we minimize reconstruction error between the input and the output. This helps the autoencoders learn the important features present in the data.

The justification of selecting autoencoders over other approaches in our paper would be credited to its ability to handle unexpected events/features. Autoencoders are back propagation-based mechanisms that reproduce the feature vector of each claim. A reality check is then performed on such a reproduction. If the distance between the original claim and reproduced claim is below a certain threshold, the claim is determined to be legitimate, otherwise it will be flagged as "fraudulent."

Thresholding

We train the network on a large chunk of "Non-fraud" data and reserve another chunk for the test set. The validation set for the Autoencoder network consists of all the "fraud" data and a small portion of "non-fraud."

Once the autoencoder has finished training, the autoencoder knows how to reproduce feature vectors representing legitimate claims onto the output layer. In order to spot suspicious claims:

- A new claim 'χk' is run through the autoencoder. The original reproduction is generated onto the output layer.
- Reconstruction error 'εk' is the distance between original and reconstruction. Thus, for a legitimate claim: $\varepsilon k \leq \chi k$, where k is threshold
- For a fraudulent claim: $\varepsilon k > \chi k$, where k is threshold.

The value of k is defined on a validation set. We optimize k against the accuracy of fraud detection for a labelled dataset and a high percentile of the reconstruction error on the validation set.

The Autoencoder network achieved an accuracy of only 74%, which is lesser than both of our supervised methods. Although the accuracy is poor, the Autoencoder model does well to detect non-fraud claims and achieves an F-1 score of 0.55 with only 2 added layers and 100 epochs. This indicates that there is a high potential to improve this model to possibly outperform our supervised methods.

Conclusion

Implementing the proposed solutions described in the paper we reap the following benefits:

1. The ML models will assist with **predicting claim application fraud,** which will be useful for scrutinizing claims thoroughly.
2. Further improvement in the project will support the government in **taking action against** fraudulent applicants, and will help with **amending rules and regulations** in the domain.
3. Improvement in the model will help indentify networks of **fraudulent physicians, providers, and beneficiaries.**
4. With virtually no domain knowledge, we can develop a **highly efficient model** to flag fraudulent claims.
5. The models will only **continue to improve** as they are fed more data over time.

Learn More

For more information on this topic, or to learn about Niyam IT's capabilities and services, contact Kurt Whiting at 703-429-2450 or email kwhiting@niyamit.com.

NIYAM IT™

PRIMARY HEADQUARTERS 202 Church St. SE, Suite 301 Leesburg, VA 20175

SOLUTION DEVELOPMENT CENTER 10201 Fairfax Blvd., Suite 224 Fairfax, VA 22030

+1 703.429.2450 info@niyamit.com www.niyamit.com