Journal of Computer Science

Research Article

Identification of Fraud in Accidental and Healthcare Insurance Using Local Outlier Factor: A Machine Learning Approach

Jyoti Lele, Vaidehi Deshmukh, Abhinav Chandra and Radhika Desai

Department of Electrical and Electronics Engineering, Dr. Vishwanath Karad MIT World Peace University, Pune, India

Article history
Received: 06-01-2025
Revised: 18-06-2025
Accepted: 23-06-2025

Corresponding Author:
Dr. Vaidehi Vaibhav Deshmukh
Department of Electrical and Electronics Engineering, Dr. Vishwanath Karad MIT World Peace University, Pune, India
Email: vaidehi.deshmukh@mitwpu.edu.in

Abstract: An unsupervised machine learning model that uses the mechanism of the Local Outlier Factor to flag and detect ambiguous as well as potentially fraudulent claims in Accidental and Healthcare insurance is proposed in this paper. It entirely automates the manual investigation of claims by claim appraisers in any organization. The ethos of this model is to comprehensively automate and expedite the claim investigation process using certain parameters to aid the claim appraiser's workload of going through straightforward claims and saving their time to investigate more critical and complex claims. The model flags anomalous claims by comparing them to the model's threshold, and input parameters and alerts are generated. These alerts generated are then investigated for fraud based on the parameters stated. The model can classify these claims and the cost of billable associated with these claims by reporting an accuracy of 99.5% for the Local Outlier Factor model in comparison with other implemented techniques of Isolation Forest which had an accuracy of only 78.37%. Our model has been tested and validated on real-world data and is showing promising results. Being able to identify and flag potentially fraudulent claims before they are paid out can save insurance companies a lot of money and resources.

Keywords: A&H, Lof, Dbscan, Tsne, Bow, Kpis, Healthcare Insurance

Introduction

Health insurance fraud is a complex and pressing issue, causing substantial financial losses that reverberate throughout the industry. In the quest to address this challenge, a robust model that can effectively identify potential instances of fraudulent activities is required. The proposed work attempts to develop one such model that takes a two-fold approach it amalgamates well-established statistical methods with contemporary machine learning techniques, thereby enhancing the model's accuracy and practicality (Kumaraswamy *et al.*, 2022). The crux of the problem revolves around the inherent ambiguity and resemblance between fraudulent claims and legitimate ones. This likeness necessitates a localized approach for outlier detection, one that can discern anomalies that a global perspective might miss. A global perspective which uses distance from neighbors as a metric to find outliers may face a scenario where a data point located nearest is anomalous. Hence it is intelligent decision to observe density locally. It is within this context that the Local Outlier Factor (LOF) algorithm finds its relevance. The research work presented here also ushers in automation for detecting fraudulent claims. The driving force behind this automation is the potential to eliminate human intervention and associated errors. By doing so, the proposed model strives not only to enhance the precision of fraud detection but also to yield substantial savings in terms of time, resources, and capital.

The central objective of this paper is to present a comprehensive and effective solution to combat health insurance fraud. By ingeniously blending traditional statistical methods with cutting-edge machine learning techniques, the model is poised to become a stalwart defense against fraudulent activities. The motivation to curtail significant financial losses attributed to health insurance fraud, coupled with the desire to preserve the integrity of the insurance ecosystem, propels this research forward.

Literature Review

Healthcare fraud has emerged as a daunting challenge, causing substantial financial setbacks and impacting patient well-being (Kumaraswamy *et al*., 2022). Addressing this complex issue calls for innovative strategies within the intricate framework of the US healthcare system. The ultimate objective is to introduce automation into fraud detection, a move that holds the potential to curb human errors and save valuable resources (Joudaki *et al*., 2014). However, the endeavor is not without hurdles, as detecting healthcare fraud and abuse through traditional methods remains an uphill battle. This underscores the pressing need for automated solutions capable of navigating the complexity of the healthcare landscape (Markovskaia *et al*., 2020) Recognizing that static approaches are insufficient, the industry has embraced dynamic technologies to proactively identify fraud patterns (Burri *et al*., 2019). Amidst this evolution, data analytics and machine learning stand out as the pillars of modernizing the insurance market. Yet, the journey is not without challenges, as insurers face a dearth of analytical models and algorithms that can truly support their objectives. It’s clear that machine learning holds the key to unlocking deeper insights and efficiencies within the sector (Rawat *et al*., 2021). The expansion of insurance clientele has propelled the importance of thorough claim analysis. This analysis, enabled by exploratory data examination and feature selection, empowers insurance companies to distinguish between valid and fraudulent claims (Rawte and Anuradha, 2015). Data mining techniques also fuel efforts to expose fraudulent claims within the healthcare insurance domain. A novel hybrid approach, melding supervised and unsupervised learning, is poised to elevate fraud detection capabilities (Waghade and Karandikar, 2018). Fraud claims lead to the misuse of medical insurance which adds a layer of complexity to an already intricate field. Machine learning and data mining are tools which can identify and combat healthcare fraud. The call for advanced techniques and data sources is apparent, suggesting a path to affordability and fraud mitigation. However, the road ahead involves strategic maneuvering through these advanced methodologies (Gill and Aghili, 2020). A systematic review of healthcare insurance fraud detection techniques underscores the industry's pursuit of effective solutions. The quest to uncover ideal application solutions is a testament to the ongoing efforts against fraud (Lalithagayatri *et al*., 2017). Against this backdrop, a hybrid model combining classification and clustering steps forward to differentiate legitimate claims from fraudulent ones. The impact of this approach echoes on a larger scale, potentially uplifting economies by curbing healthcare fraud (Li *et al*., 2022). The proposed theoretical model for medical insurance fraud identification in Kunickaitė *et al*. (2020) takes a holistic approach, exploring dimensions of time, quantity, and expenses. This approach, validated through real-world medical records, sheds light on distinctive behavioral characteristics that can drive AI and machine learning technologies for fraud detection. Machine learning's potential to revolutionize fraud detection is tangible. Decision Trees, Bagging, Random Forests, and Boosting algorithms all play a part in this transformation. The efficacy of these algorithms comes to light through rigorous evaluation metrics. While challenges persist, the promise of machine learning in tackling the costly menace of insurance fraud remains undiminished. In Baader and Krcmar (2018) the use of machine learning algorithms takes center stage. A fusion of these algorithms aims to categorize statements as true or false which is a fundamental aspect of fraud detection. This approach is grounded in real-world datasets, highlighting the relevance of accurate data in detecting fraud. In the existing landscape of research, the complexity of health insurance fraud detection has been acknowledged, yet comprehensive solutions remain limited. The gap lies in the integration of diverse methodologies into a unified framework that not only identifies anomalies but also does so with a reduced reliance on human intervention.

The novelty of this conducted research lies in the fusion of localized outlier detection using LOF with a comprehensive machine learning approach. The seamless integration of these two elements empowers the model to not only detect fraud effectively but also minimize the chances of false positives and negatives. This dual-edged approach encapsulates the essence of this paper's innovation.

System Description

In the proposed work, the Accident and Health Insurance dataset under the name India A&H Fraud for Liberty Mutual Insurance Group open source Bitbucket repository is used. The dataset contains a total of 21,263 records and 85 features. Some features have text data type whereas some are numeric. The dataset is first pre-processed by the model, which includes data cleaning, imputation, and normalization. The next step is to find outliers and anomalies in the data using conventional statistical techniques like hypothesis testing, regression analysis, and clustering. Following the identification of the outliers, the data is fed into machine learning algorithms like decision trees, balanced random forests, and ANN. The cleaned data is used to train these algorithms to find trends and connections that point to fraud. After that, predictions about the likelihood of a claim being fraudulent are made using algorithms. The model is validated using a holdout sample of data that has been set aside for this purpose. The model is evaluated based on its F1 score, recall, accuracy, and precision. This assessment is visualized via a dashboard that helps in tracking the model's performance on a timely basis. We can conclude that the model offers a solid and trustworthy solution to the issue of health insurance fraud by combining conventional statistical techniques and machine learning algorithms.

Data Collection

As shown in Figure 1, ETL stands for Extract, Transform, Load, and it refers to the process of extracting data from various sources, transforming that data into a usable format, and then loading it into a target system, such as a database or data warehouse (El-Sappagh *et al*., 2011; Ying-lan and Bing, 2009). Here's a more detailed breakdown of how ETL works:

i. Extract: The first step is data extraction from multiple sources, such as databases, APIs, online services, or flat files, which is the initial stage. From these sources, the information is gathered and copied into a staging area where it may be processed. In our case, the data is extracted from a backend Excel sheet, but the model proposes to extract data from the Jump Box
ii. Transform: After the data has been extracted, it must be converted into a format that can be used. Prepare the data for analysis, this includes cleaning, eliminating duplicates, and reformatting the data. In this step, the data may also be enhanced by the addition of new fields, such as computed fields or geo-location information
iii. Load: The modified data must then be loaded into a target system, like a database or data warehouse, as the last step. Data must be mapped to the target schema for it to comply with the standards of the target system

Overall, the ETL process is critical for organizations that need to integrate data from multiple sources and make it available for analysis and decision-making in real-time scenario. It ensures that data is accurate, consistent, and reliable, and can help organizations gain insights that can drive business success.

Data Cleaning and Feature Selection

There were a lot of challenges involved in modifying the dataset to make it suitable for model training as shown in Figure 2. They included data acquisition challenges, the feature selection process being difficult due to an imbalance in data, and unidentified fraudulent claims in the data. This challenge was faced using specific data sources such as AWS Redshift. The challenge for unidentified fraud was aced using LOF anomaly detection which is explained in section 3.3. BRF (Balanced Random Forest) is used for data imbalance. The process of EDA, and implementation of all algorithms has been done using Python 3.4.

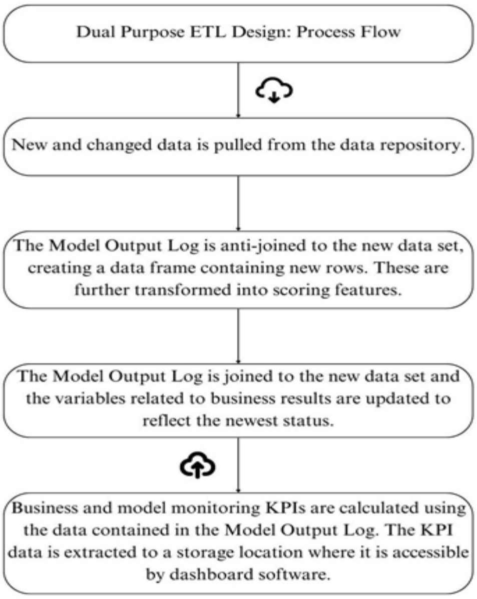
Dual Purpose ETL Design: Process Flow
New and changed data is pulled from the data repository.
The Model Output Log is anti-joined to the new data set, creating a data frame containing new rows. These are further transformed into scoring features.
The Model Output Log is joined to the new data set and the variables related to business results are updated to reflect the newest status.
Business and model monitoring KPIs are calculated using the data contained in the Model Output Log. The KPI data is extracted to a storage location where it is accessible by dashboard software.

Fig. 1: Explaining the Purpose of ETL Used in Our Model

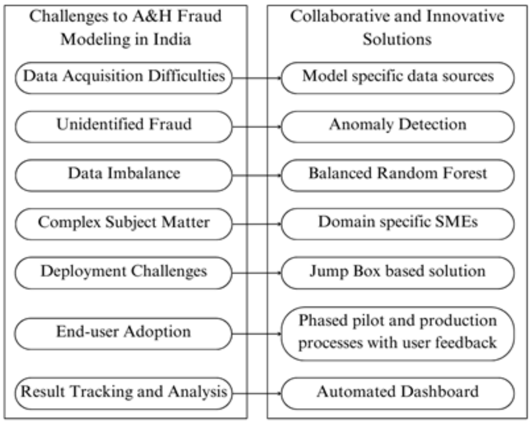
Challenges to A&H Fraud Modeling in India
Collaborative and Innovative Solutions
Data Acquisition Difficulties
Model specific data sources
Unidentified Fraud
Anomaly Detection
Data Imbalance
Balanced Random Forest
Complex Subject Matter
Domain specific SMEs
Deployment Challenges
Jump Box based solution
End-user Adoption
Phased pilot and production processes with user feedback
Result Tracking and Analysis
Automated Dashboard

Fig. 2: Challenges and Complexities Involved and their Solutions

Exploratory data analysis is necessary as size of the dataset is huge. It is performed by resolving missing values, feature engineering, and target variable enhancement. It is followed by feature selection and feature validation using Boruta's algorithm (Aslam *et al*., 2022), which enables us to select features based on which features would impact our model most to least. Boruta's algorithm's primary objective is to meticulously navigate through an array of features and pinpoint those of paramount importance. It trains on real and simulated data, assigning scores based on impact. Features with higher scores in real data are kept, while others are discarded. This iterative process continues until confidence in chosen features is high. These culled features build an optimized model, enhancing predictions. Essentially, Boruta accelerates discovery in data, boosting predictive abilities. Figure 3 shows the features that have been selected using Boruta's algorithm, i.e., the subset is 'Pin_of_Hospital.WOE','ICD_WOE', 'FreqPin_of_Hospital', 'Claimed_Amount', 'Doctor_Charges', 'MedianClaimed', 'Sum_Insured', 'LossToExp', 'StartToLoss', 'ICDZeroFraud' i.e total of 9 features out of 85 features present in the dataset.

The heat map reflects the dependencies of the features of the model, i.e., the input parameters are their correlation. We have made sure that the features that have been used as input parameters do not have a correlational value above 0.6.

Outlier Detection

To detect outliers, the process of anomaly detection is implemented which is a subcategory of unsupervised machine learning that identifies cases that are probable statistical outliers and overall categorizes the data into clusters in which these outliers are present. The reason we are using anomaly detection to detect outliers is that it helps us classify cases that are ambiguous in nature, maybe one of their kind in terms of their uniqueness or also spot claim cases prone to be illegitimate or false i.e. fraudulent. These ambiguous cases can be potentially fraudulent and cannot be differentiated easily as they resemble a lot of similarities from legitimate cases.

Anomaly detection is implemented using Local Outlier Factor (LOF) algorithm as shown in Figure 4. The algorithm is based on the k-nearest neighbor algorithm. Using typically 20% of data points as neighbor, it produces an anomaly score that identifies the outlier data points in the data set. The local density deviation of a given data point in relation to neighboring data points is calculated to achieve this. The local density is estimated based on the distance between a given data point and neighbors. Thus points having similar local density are clustered together in red (legitimate claims) and only one point is in blue (fraudulent claims).

The Isolation Forest algorithm is also implemented to check performance of LOF with it. This unsupervised machine learning technique is based on the principle of isolating anomalies rather than the general practice of profiling good data points. The algorithm randomly selects features from the dataset and randomly selects a split value between the maximum and minimum values of the selected features (Jiang *et al*., 2021). Our results show an accuracy of 78.37% which is considerably less when compared to LOF. The advantage offered by LOF technique is calculation of local density of a data point; that aptly indicates outlier. As the data is imbalanced, we cannot rely on a distance metric used by isolation forest to detect outlier in this application. Hence LOF performs better than isolation forest for such applications involving imbalanced data.

Use of NLP to Process Text Features

Considering that each phrase's context might be used to train the model after yielding numerical representations of related terms; Natural Language Processing (NLP) techniques are incorporated into the proposed model to analyze the claims' text data and find discrepancies and uncommon language usage. A column 'Diagnosis Text' in the dataset consisted of unstructured data that could be converted, and additional features along with those obtained using Bouruta's algorithms could be taken into consideration from the column.

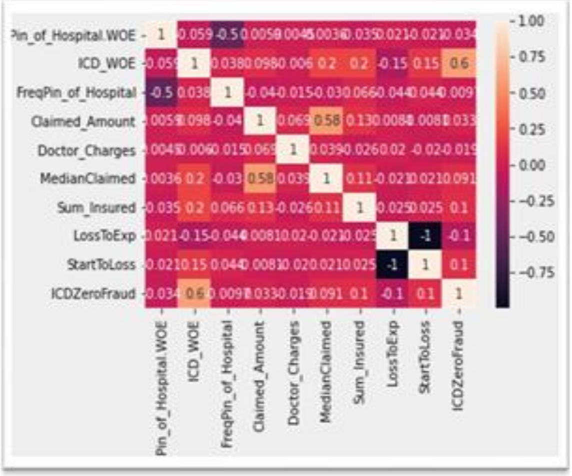

Fig. 3: Heat Map Showing Feature Dependencies

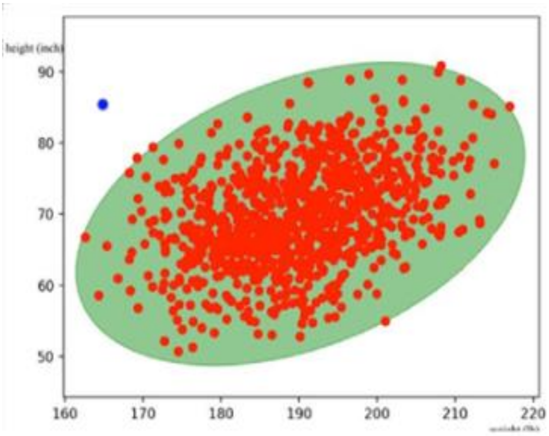

Fig. 4: Anomaly Detection Example using LOF

Specific techniques from NLP viz. TF-IDF, GloVe and Word2Vec are explored that are significant for text analysis. Term Frequency-Inverse Document Frequency (TF-IDF) yields vocabulary-based calculations (Wang *et al*., 2018), These computations accentuate the weight of words in the text, capturing their importance effectively. This process aids in constructing features that have the potential to enhance model performance (Li *et al*., 2023), GloVe involves creating a matrix that captures the frequency of words appearing together in texts. Through dimensionality reduction, these co-occurrence scores are transformed into meaningful insights, revealing the frequency of word collaborations with other terms (Pennington *et al*., 2014; Johnson and Khoshgoftaar, 2021). Word2Vec is an algorithm that generates a distributed semantic representation of words in the text. This sophisticated technique generates word embedding that encapsulate contextual meanings and relationships. This empowers a deeper comprehension of nuanced meanings embedded in the vocabulary of the text (Johnson and Khoshgoftaar, 2020; 2022). Finally, using the Bag of Words paradigm, TF-IDF technique is implemented to obtain sparse vector representations of the text data. The number of times a word appears in a document is counted; hence the text content is converted to numerical feature vectors. These word counts are used to compare documents and determine how similar they are for applications like topic modeling, document classification, and search.

The outcomes of Bag of Words (BoW) are displayed in Table 1. A vocabulary out of all the distinct terms in the corpus is generated in order to create a BoW representation of a document. The frequency of each word in the lexicon is then created as a vector for each page. Vectors for multiple ICD codes that represent similar diagnoses or related diseases, such as fever combined with a cold or pneumonia, fractures accompanied by fever, or merely fever are constructed. Since fever is the recurring element, vectors with the same diagnosis or ailment are generated. One of the limitations of the BoW model is that it does not consider the order of words or their semantic meaning. Therefore, it can result in a loss of important information, particularly in tasks where the context of words is crucial, such as language translation and sentiment analysis.

Table 1: Bag of Words

Variable Name	Mean Variable Importance	Associated Terms
V2	51.26916	FEVER
V23	48.43494	UTI
V19	43.98489	DENGUE WITH TCP
V18	41.28179	DENGUE
V1	33.09415	(Entry is blank)
V24	32.22399	PNEUMONIA
V8	29.66566	DENGUE FEVER
V7	29.23064	COVID 19

Implementation of ML Algorithm

The cleaned data containing selected features is applied to a simple clustering algorithm DBSCAN which uses the density of data points to group them together in clusters (Amiruzzaman *et al*., 2021; Diaz-Granados *et al*., 2015; El-Sappagh *et al*., 2011; Nabrawi and Alanazi, 2023). The colors of the clusters in the Figure 5 indicate levels of density of different data points.

The popular algorithm t-SNE (t-distributed Stochastic Neighbor Embedding), which is used to visualize high-dimensional datasets (Lacruz and Saniie, 2021) is also used to visualize clustered data and is demonstrated in Figure 6. Here the outliers that are the fraudulent claims are visualized in two-dimensional space as small orange dots whereas the blue dots are the legitimate claims. The figure represents the local significance of the outliers in all clusters and how fraudulent cases can be very similar to legitimate cases.

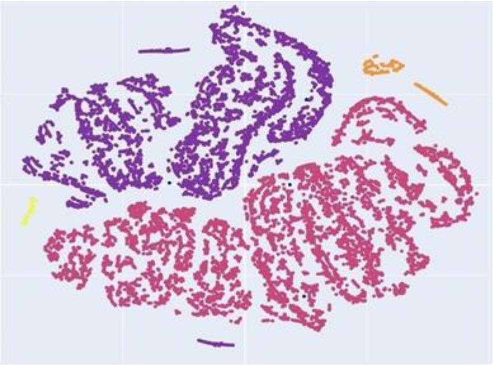

Fig. 5: Clusters obtained using DBSCAN

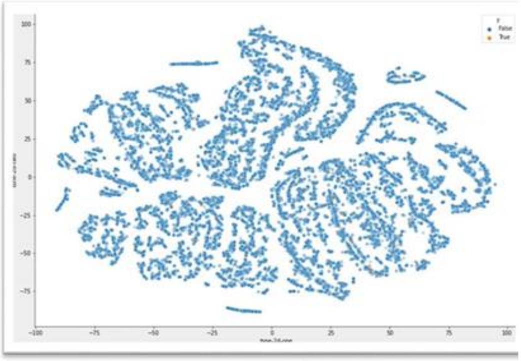

Fig. 6: Outliers Using t-SNE

Representation of Data Using Dashboard

Clustered data, thus obtained, is represented using Power BI visualization tool as explained in this section. Figure 7 shows the data flow through pre-processing to clustering steps. To provide analysis of claims to appraisers who assess and evaluate claims, it is easy to present the analysis using Power BI dashboard. It is the appraisers who assess medical treatments, procedures, accident cause, injuries and damages and policy documents. Based on this assessment, they evaluate claims and are responsible for settlement of claims. This model is developed to reduce efforts of appraisers in claims assessment and evaluation. Two dashboards are created; the first one is developed to track business impact and the second one is developed to track the model's performance during the assessment process and to visualize the results. Figure 8 shows the business impact dashboard. The dashboard displays KPIs (Key Performance Indicators) that comprehensively track business results and their disposition at any given amount of time. It directly reflects the cost savings of the model at any given point of time. The values of KPIs are calculated each week over a sample period of April to August in the year 2022. Figure 9 shows a model monitoring Dashboard using Power BI. It indicates metrics like accuracy, specificity, precision, recall indicating model's performance.

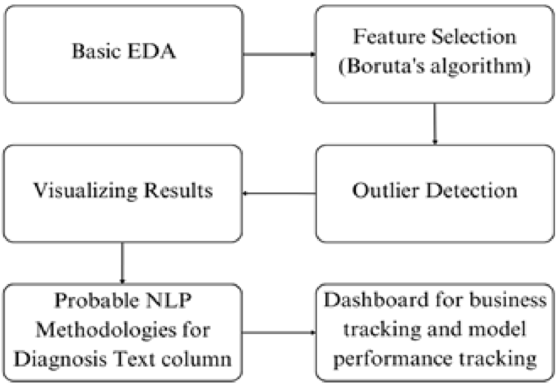
Basic EDA
Feature Selection (Boruta's algorithm)
Outlier Detection
Visualizing Results
Probable NLP Methodologies for Diagnosis Text column
Dashboard for business tracking and model performance tracking

Fig. 7: Flow of data

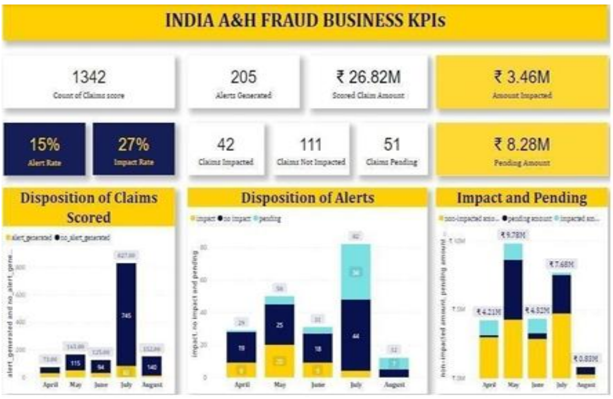

Fig. 8: Business KPIs Tracking

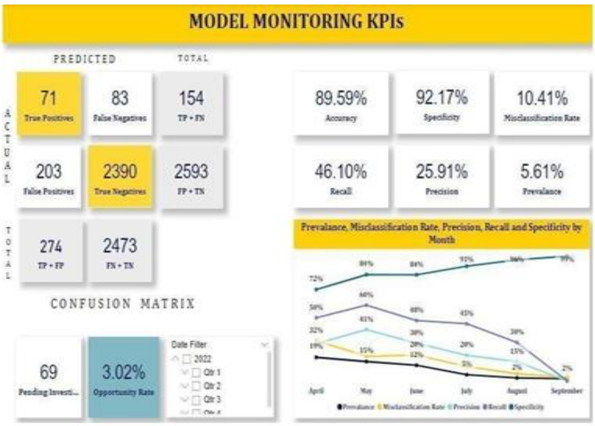

Fig. 9: Model monitoring dashboard

KPIs used for business tracking and indicated by the dashboard are as follows:

a. Count of Claims Scored - It measures how many claims have been processed via the model and compared to a predetermined threshold to determine whether they are fraudulent
b. Alerts generated – It evaluates the number of fraudulent claims that have been generated in comparison to the model's threshold. If we give the claim a score that is higher than the threshold, we declare it to be fraudulent
c. Total Non-Impacted Claims It measures the overall number of claims on which the company had no bearing
d. Total No of Claims Pending – It measures the total number of claims that are still being processed, such as those that are being assessed, reimbursed, or subject to more scrutiny
e. Scored Claimed Amount – It provides the total amount value of all fraudulent claims that the model has assessed and is given by:

$$\text{Scored Claimed Amount} = \sum(\textit{Claimed Amount}) \qquad (1)$$

f. Amount Impacted – It provides a total amount accounting of all fraudulent claims that have been impacted by the business and is given by:

$$\text{Amount Impacted} = \sum(\textit{Impacted Amount}) \qquad (2)$$

g. Pending Amount – It provides a total amount accounting of all claims that have been scored by the model under the disposition of pending and is given by:

$$\text{Pending Amount} = \sum(\textit{Pending Amount}) \qquad (3)$$

h. Alert Rate – The model's capacity to differentiate between claims it scores as fraudulent and the overall number of claims it has validated is described by the alert rate KPI. Alert rate is given by:

$$\text{Alert rate} = \sum(Alerts\ generated)/\sum(claims) \tag{4}$$

i. Impact Rate – The impact rate KPI describes the model's capacity to separate the claims that are impacted from the total claims containing both unaffected and impacted. The impact rate is given by:

$$\text{Impact rate} = \frac{\sum(claims\ impacted)}{\sum(impacted + non\_impacted)} \tag{5}$$

j. Model Output Rate – The model ratios to the total number of claims impacted to the total number of alerts created. This ratio is referred to as the model output rate KPI and is given by:

$$\text{Model output rate} = \frac{\sum(claims\ impacted)}{\sum(impacted + non-impacted + pending)} \tag{6}$$

k. Disposition of Claims Scored – It displays the distribution of the total number of claims that the model passed in comparison to the monthly alerts that were generated

l. Disposition of Alerts – It shows the model's classification of all warnings as fraudulent into three categories: impact, no-impact, and pending claims. In a dashboard shown in Figure 8, these claims are shown as monthly distributions using a stacked bar chart

m. Impact and Pending – It shows how the model divides the impacted amount and pending amount against the impacted claims and pending claims

Table 2 gives an ad-hoc analysis of the claim appraisers and stakeholders of the business every week over the period of April to August in the year 2022. E.g. in the month of April, model assesses 21 claims and the model generated alert for all 21 claims. Out of these 21 claims, 5 are declared as fraudulent, 14 as legitimate and 1 as pending and then corresponding amounts are calculated. The values of the KPIs calculated every week. These values are further summed up to display month wise data using the Power BI dashboard as shown in Figure 8.

Table 3 shows the number of alerts classified as impact, i.e., fraudulent claims, non-impacted, i.e., legitimate claims, and pending claims each month. The values of the Table 3 are calculated by adding values of different types of claims stated for each week of a particular month in the Table 2. As seen from these values, the model can predict fraudulent claims with reasonable accuracy. Table 4 shows the disposition of alerts generated and same is represented using a bar-graph in the dashboard shown in Figure 8. Table 5 shows the impacted amount, non-impacted amount, and pending amount and same values are represented using a bar-graph in the dashboard of Figure 8.

Table 2: Business Spreadsheet View in Tabular Format

Starting date of a Week	Count of Claims Scored	Alert Generated	Alert Rate	Impact (fraudulent claims)	Non-impacted (legitimate claims)	Pending claims	Impacted Amount	Pending Amount
18-04-2022	21	21	100%	5	14	1	7,85,023	1,49,738
25-04-2022	52	9	17%	4	5	0	2,91,343	0
02-05-2022	64	17	27%	2	8	2	4,66,840	10,30,731
09-05-2022	21	5	24%	7	3	0	96,394	0
16-05-2022	51	13	25%	6	7	0	3,88,027	0
23-05-2022	29	15	52%	5	7	3	2,17,975	33,27,892
06-06-2022	20	16	80%	5	8	3	3,25,740	1,51,941
13-06-2022	79	13	16%	4	8	1	7,13,510	2,72,685
20-06-2022	10	1	10%	0	1	0	0	0
27-06-2022	45	4	9%	0	2	2	0	1,34,366
04-07-2022	61	6	10%	0	2	4	0	3,32,777
11-07-2022	461	47	10%	1	35	11	63,798	12,77,990
18-07-2022	12	1	8%	0	1	0	0	0
25-07-2022	264	25	9%	3	5	17	1,15,500	10,51,339
01-08-2022	112	10	9%	0	5	5	0	3,07,698
08-08-2022	40	2	5%	0	0	2	0	2,41,130
Total	1342	205	15%	42	111	51	34,64,150	82,78,287

Table 3: Segregation of Alerts

KPI/Month	April	May	June	July	Aug
Impact	9	20	9	4	----
Non-impacted	19	25	18	44	5
Pending	1	5	4	34	7

Table 4: Disposition of Alerts

KPI/Month	April	May	June	July	Aug
Alert generated	30	50	31	82	12
No alert generated	43	115	94	745	140

Table 5: Impact and Pending Amount

Type of claims	April	May	June	July	Aug
Non-impacted	2.98M	4.25M	2.85M	4.7M	2.78M
Pending	0.15M	4.35M	0.42M	2.79M	0.54M
Impacted	1.07M	1.16M	1.03M	0.17M	0.27M

To calculate the model performance metrics, a log table is created at the backend of the dashboard. The log table contains a week-by-week record of cases with their claim amount, generated alerts, true cases with their claim amount, true negatives, false positives, pending investigations, etc. Average values of various performance metrics of the model for a month are calculated using the log table and are presented in Table 6 and their overall average values are indicated by the dashboard shown in Figure 10. The dashboard also displays a confusion matrix of the model; wherein values of true positives, true negatives, false positives and false negatives are provided. The cases of false positives and false negatives have to be handled by humans with critical assessment.

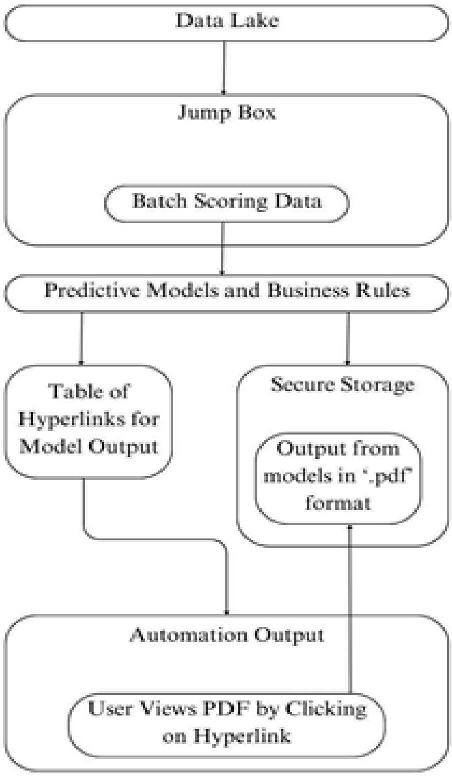
Data Lake
Jump Box
Batch Scoring Data
Predictive Models and Business Rules
Table of Hyperlinks for Model Output
Secure Storage
Output from models in '.pdf' format
Automation Output
User Views PDF by Clicking on Hyperlink

Fig. 10: Entire pipeline-based solution using Jump box, i.e. Data Lake

Generation of Reports

Large amounts of organized and unstructured data can be stored and analyzed at any scale using a data lake, which is a centralized repository. The storage layer, the processing layer, and the analytics layer are common layers in a data lake architecture. A pipeline-based solution in this sense refers to a collection of connected data processing procedures that convert raw data into actionable insights. Data ingestion, data cleaning, data transformation, and data analysis are common stages in a pipeline.

A Jump Box, or Bastion Host, is a server that is used to securely access and manage other servers or devices within a network. In a data lake architecture, a Jump Box can be used to securely access and manage the various components of the system, including the storage and processing layers. An entire pipeline-based solution using a Jump Box in a Data Lake architecture involves several steps. First, raw data is ingested into the data lake through various sources such as API calls, log files, and batch uploads. The data is then cleaned, transformed, and prepared for analysis using various tools and techniques. The data is then saved in the storage layer of the data lake, which might use a variety of tools including the Hadoop Distributed File System (HDFS), Amazon S3, or Azure Blob Storage. Following that, the data is processed and analyzed using programs like Apache Spark or Apache Hadoop, which would run on the data lake's processing layer. Finally, the insights gained from the analysis is presented to the end-users through various visualization tools, such as Tableau or Power BI, which run on the analytics layer of the data lake. Throughout this entire process, the Jump Box is used to securely manage and access the various components of the data lake architecture, ensuring that the data is stored, processed, and analyzed securely and efficiently.

This pipeline of data processing is to be used by real-world users i.e. appraisers to fetch results and generate reports. The report may include information on the claims being appraised, such as its location, size, condition, and any unique features. The report may include an assessment of the claims value based on approaches to simplify the understanding of the alert levels. Table 7 shows format of the report generated by data-lake and jump box architecture, depending upon the score given by the model, intensity of fraud alert is allotted. Similarly, Table 8 shows report format for the model performance. Table 9 shows performance of LOF algorithm.

Table 6: Model Evaluation Metrics

Metric/Month	April	May	June	July	Aug	Sept
Accuracy	68%	80%	80%	89%	94%	97%
Prevalence	19%	15%	12%	5%	2%	2%
MF Rate	32%	20%	20%	11%	6%	3%
Precision	29%	41%	30%	20%	15%	0%
Recall	50%	60%	48%	45%	30%	0%
Specificity	72%	84%	84%	91%	96%	99%

Table 7: Output Format of the Results Viewed by Business and Claim Appraisers based on the alert level

Alert Date	LGIL Claim Number	City	DOA	ICD	Claimed Amount	Model Score	Fraud Alert
9/6/2022	44721X-XXXXXX-XXXXX	Garhi Harsaru	24-03-2022	K29	72,498	0.8011954	Very High
9/6/2022	44721X-XXXXXX-XXXXX	BAHRAICH	24-04-2022	K29	61,789	0.5970871	Medium
9/6/2022	44721X-XXXXXX-XXXXX	Arjun Nagar	13-04-2022	R50	47,152	0.5930373	Medium
9/6/2022	44721X-XXXXXX-XXXXX	NEW DELHI	23-04-2022	A01	91,294	0.5323448	Medium
9/6/2022	44721X-XXXXXX-XXXXX	Chorasi	9/4/2022	A01	46,692	0.5306475	Medium
9/6/2022	44721X-XXXXXX-XXXXX	BHIWANDI	26-04-2022	A75	93,254	0.5125751	Medium
9/6/2022	44721X-XXXXXX-XXXXX	Daskroi	16-04-2022	A01	64,388	0.5125349	Medium
9/6/2022	44721X-XXXXXX-XXXXX	Huzur	1/5/2022	A01	86,474	0.5070088	Medium
9/6/2022	44721X-XXXXXX-XXXXX	Daskroi	8/4/2022	N20	1,11,145	0.5065927	Medium
9/6/2022	44721X-XXXXXX-XXXXX	Ghaziabad	6/6/2022	H65	55,455	0.5032221	Medium
10/6/2022	44722X-XXXXXX-XXXXX	Bangalore	25-05-2022	T84	2,27,254	0.5721175	Medium
10/6/2022	44722X-XXXXXX-XXXXX	Bangalore	25-05-2022	T84	74,930	0.5600798	Medium

Table 8: Output Format of the Results of model performance

	Precision	Recall	F1- score	Support	Accuracy	Macro Avg	Weighted Avg
False	0.99	0.99	0.99	21146	21262	0.5	0.99
True	0	0	0	116	0.99	21262	21262

Table 9: Model Performance Results

LOF Score: 223	LOF 0.9895118050982974

The output format of the results viewed by business and claim appraisers will depend on the specific tool or system being used. However, typically these professionals will be presented with a detailed report or summary of the appraisal results based on alert levels that have been used.

Results and Discussion

The suggested model for analysis of claims find potential fraud cases using a combination of conventional statistical methods and machine learning techniques like clustering and anomaly detection. It identifies groups of claims with comparable traits and claims that stand out from the rest of the data. As a result, it is simpler to identify claims that might be a component of a larger fraud scheme. A dashboard used to track the model's performance during the assessment process helps to visualize the results. The benefits that are inculcated from the model developed are reduced referral cycle time, i.e. less time consumed in reviewing claims, identifying claims based on fraud alert severity, automated result tracking via the dashboard, no direct costs involved in deployment, and use case of a reusable framework.

Challenges included difficulties with data acquisition, a difficult time choosing features because of data imbalance and presenting unidentified fraudulent claims in the data. While resolving these challenges, SMOTE was used to combat with data imbalance, model-specific data selection was carried out, and feature selection was improved by incorporating NLP methodologies into pertinent dataset. Undiscovered incidents of fraud in the data were taken care of by anomaly detection. Approaching the problems by trial and error worked best with a dataset that came with certain challenges. It was important to get an understanding of the dataset, what it represents, the terminologies and calculations, and the workings of healthcare firms to ensure we relied on the correct features to achieve good accuracy for our model. These were some of the key takeaways. Using the same technical foundation, additional models can be added. To detect more fraud, sophisticated methods like deep learning and network analysis can be used. At various points during the claim lifecycle, models may be scored, and learning and network analysis can be used.

Conclusion

The proposed model offers a thorough method for identifying health insurance fraud claims, using an approach of anomaly detection LOF with a very high accuracy of 99.5% which in turn results in cost savings for any organization. Thus, the model offers a solid and dependable answer to the issue of health insurance fraud by combining conventional statistical techniques and machine learning algorithms. The model has demonstrated promising results after being tested and validated on actual data, making it an important tool for insurance companies to lessen the effects of fraud. It can track down cases based on its fraud alert level which alerts the appraisers to give high priority to complex claims and ease their burden by appraising simple claims automatically. The comprehensive business dashboard elevates and tracks the business impact of the model actively and the model monitoring dashboard tracks that the model is performing well and that there are no data population changes in the model. There are some limitations to the model deployment in real time scenario. We need to work upon computational cost and scalability when more records pour into the dataset.

Acknowledgment

We would like to acknowledge Liberty Mutual Insurance Global Risk Management, East division for providing us with the insurance data for this research work.

Funding Information

The authors have not received any financial support or funding to report.

Author's Contributions

Jyoti Lele: Oversight and leadership responsibility for the research activity planning and execution, including mentorship external to the core team.

Vaidehi Deshmukh: Drafting the article or reviewing it critically for significant intellectual content; and give final approval of the version to be submitted and any revised version.

Abhinav Chandra: Considerable contributions to conception and design, and/or acquisition of data.

Radhika Desai: Analysis and interpretation of data.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Amiruzzaman, M., Rahman, R., Islam, Md. R., & Nor, R. M. (2021). Evaluation of DBSCAN algorithm on different programming languages: An exploratory study. *2021 5th International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, Dhaka, Bangladesh. https://doi.org/10.1109/iceeict53905.2021.9667925

Aslam, F., Hunjra, A. I., Ftiti, Z., Louhichi, W., & Shams, T. (2022). Insurance fraud detection: Evidence from artificial intelligence and machine learning. *Research in International Business and Finance*, *62*, 101744. https://doi.org/10.1016/j.ribaf.2022.101744

Baader, G., & Krcmar, H. (2018). Reducing false positives in fraud detection: Combining the red flag approach with process mining. *International Journal of Accounting Information Systems*, *31*, 1–16. https://doi.org/10.1016/j.accinf.2018.03.004

Burri, R. D., Burri, R., Bojja, R. R., & Buruga, S. R. (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

Diaz-Granados, M., Diaz-Montes, J., & Parashar, M. (2015). Investigating insurance fraud using social media. *2015 IEEE International Conference on Big Data (Big Data)*, Santa Clara, CA, USA. https://doi.org/10.1109/bigdata.2015.7363893

El-Sappagh, S. H. A., Hendawi, A. M. A., & El Bastawissy, A. H. (2011). A proposed model for data warehouse ETL processes. *Journal of King Saud University - Computer and Information Sciences*, *23*(2), 91–104. https://doi.org/10.1016/j.jksuci.2011.05.005

Gill, J. K., & Aghili, S. (2020). *Health insurance fraud detection*.

Jiang, X., Lin, K., Zeng, Y., & Yang, F. (2021). Medical Insurance Medication Anomaly Detection based on Isolated Forest Proximity Matrix. *2021 16th International Conference on Computer Science & Education (ICCSE)*, Lancaster, United Kingdom. https://doi.org/10.1109/iccse51940.2021.9569723

Johnson, J. M., & Khoshgoftaar, T. M. (2020). Semantic Embeddings for Medical Providers and Fraud Detection. *2020 IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI)*, Las Vegas, NV, USA. https://doi.org/10.1109/iri49571.2020.00039

Johnson, J. M., & Khoshgoftaar, T. M. (2021). Medical Provider Embeddings for Healthcare Fraud Detection. *SN Computer Science*, *2*(4). https://doi.org/10.1007/s42979-021-00656-y

Johnson, J. M., & Khoshgoftaar, T. M. (2022). Encoding High-Dimensional Procedure Codes for Healthcare Fraud Detection. *SN Computer Science*, *3*(5). https://doi.org/10.1007/s42979-022-01252-4

Joudaki, H., Rashidian, A., Minaei-Bidgoli, B., Mahmoodi, M., Geraili, B., Nasiri, M., & Arab, M. (2014). Using Data Mining to Detect Health Care Fraud and Abuse: A Review of Literature. *Global Journal of Health Science*, *7*(1). https://doi.org/10.5539/gjhs.v7n1p194

Kumaraswamy, N., Markey, M. K., Ekin, T., Barner, J. C., & Rascati, K. (2022). Healthcare Fraud Data Mining Methods: A Look Back and Look Ahead. *Perspect Health Inf Manag*, *19*(1), 1.

Kunickaitė, R., Zdanavičiute, M., & Krilavičius, T. (2020). Fraud Detection in Health Insurance Using Ensemble Learning Method. *International Conference on Information Technology*, 70–77.

Lacruz, F., & Saniie, J. (2021). Applications of Machine Learning in Fintech Credit Card Fraud Detection. *2021 IEEE International Conference on Electro Information Technology (EIT)*, Mt. Pleasant, MI, USA. https://doi.org/10.1109/eit51626.2021.9491903

Lalithagayatri, T., Priyanka, T., & Pavate, A. (2017). Fraud Detection in Health Insurance using Hybrid System. *International Journal of Engineering Research and Technology (IJERT*, *5*(1), 1–3.

Li, J., Lan, Q., Zhu, E., Xu, Y., & Zhu, D. (2022). A Study of Health Insurance Fraud in China and Recommendations for Fraud Detection and Prevention. *Journal of Organizational and End User Computing*, *34*(4), 1–19. https://doi.org/10.4018/joeuc.301271

Li, W., Ye, P., Yu, K., Min, X., & Xie, W. (2023). An abnormal surgical record recognition model with keywords combination patterns based on TextRank for medical insurance fraud detection. *Multimedia Tools and Applications*, *82*(20), 30949–30963. https://doi.org/10.1007/s11042-023-14529-4

.

Markovskaia, N. (2020). Detecting Insurance Fraud with Machine Learning. *Plug and Play Tech Center.*

Nabrawi, E., & Alanazi, A. (2023). Fraud Detection in Healthcare Insurance Claims Using Machine Learning. *Risks*, *11*(9), 160. https://doi.org/10.3390/risks11090160

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. https://doi.org/10.3115/v1/d14-1162

Rawat, S., Rawat, A., Kumar, D., & Sabitha, A. S. (2021). Application of Machine Learning and Data Visualization Techniques for Decision Support in the Insurance Sector. *International Journal of Information Management Data Insights*, *1*(2), 100012. https://doi.org/10.1016/j.jjimei.2021.100012

Rawte, V., & Anuradha, G. (2015). Fraud detection in health insurance using data mining techniques. *2015 International Conference on Communication, Information & Computing Technology (ICCICT)*, Mumbai, India. https://doi.org/10.1109/iccict.2015.7045689

Waghade, S. S., & Karandikar, A. M. (2018). A Comprehensive Study of Healthcare Fraud Detection based on Machine Learning. *International Journal of Applied Engineering Research*, *13*(6), 4175–4178.

Wang, M., Xu, L., & Guo, L. (2018). Anomaly Detection of System Logs Based on Natural Language Processing and Deep Learning. *2018 4th International Conference on Frontiers of Signal Processing (ICFSP)*., Poitiers. https://doi.org/10.1109/icfsp.2018.8552075

Ying-lan, F., & Bing, H. (2009). Design and Implementation of ETL Management Tool. *2009 Second International Symposium on Knowledge Acquisition and Modeling*, Wuhan, China. https://doi.org/10.1109/kam.2009.105