ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

Generative AI and DevOps for Scalable Insurance Fraud Detection

Elena Petrova
Asst Professor

Abstract

Novel AI applications transform insurance operations from intelligence surveillance to intelligent services, and a core focus is fraud detection. The hypergrowth of generative AI creates new capabilities that enable organizations to shift from mainly reactive to proactive fraud detection. Generative models support the synthesis of fraud-related data signals to augment existing training datasets, thereby enhancing detection performance. Underlying data patterns should be suitable for the generation of realistic synthetic samples. With demographic, health, and police-report data forming the foundation, convincing anomaly signals can be synthesized for insurance fraud. The integration of such signals into the fraud-detection pipeline is supported by DevOps practices, which cover data engineering, continuous integration/continuous delivery, governance, and lifecycle management. The proposed approach is operational in property and casualty, health, and auto insurance.

The generation of data signals aligned with actual fraud has the potential to fill critical training gaps in detection models. The samples introduced through synthesis offer business value not just in detection but also as adversarial test cases and for automated explainability checks. Such comprehensive exploitation can be achieved at scale and overtime without data-ownership friction. The established architecture supports the creation of any form of signal related to any dimension of fraud. Driving connecting veins into the systems of record offers a pragmatic, scalable, and operational approach to doing so. Attention to latency tolerance and data governance ensures that synthesis is feasible, useful, and compliant.

Keywords : Insurance intelligence; fraud detection; generative modeling; anomaly signal; supervised learning; DevOps; domain-serving architecture; feedback loop; risk; compliance.

1. Introduction

AI-driven insurance intelligence applies natural-language processing, computer vision, and generative modeling to insurance data, aiming to develop shared sources of insurance fraud delay, detection, and prevention signals. The contemplated set of signals embraces all relevant insurance lines and relies on the adoption of a DevOps-enabled environment for machine-learning model development and deployment, with an initial focus on fraud detection.

Generative modeling encompasses both model building and data synthesis. Generative models learn to synthesize data based on an observed training set, while the synthesized data can be leveraged to enhance the robustness of predictive models by providing additional, albeit synthetic, data. Typical areas of interest in the training set include fraud detection, fraud delay, and fraud prevention. In these domains, fraud may be defined as actions that benefit the actor at the expense (monetary or otherwise) of a victim without that victim's informed consent. A pipeline integrating generative modeling with DevOps may enable the continuous generation of synthetic data in support of a variety of fraud signals—ideally all relevant for a given insurance line—providing substantial ancillary benefits to insurance companies and their policyholders.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

1.1. Background and Significance

The Business Use of Generative AI in Insurance Intelligence; Compose an objective, scholarly summary of the topic with formal structure and evidence-based arguments. Generative AI models and underlying modeling principles have shown categorical success in various business applications, including a handful in the domain of insurance. The broad application of generative modeling principles in insurance intelligence is based on two key ideas: image and text are fundamental sources of data for most lines of insurance, and advanced AI solutions rely on massive amounts of high-quality labeled data. Auto, health, property and casualty (P&C), and life insurance deal with large volumes of claims and applications, necessitating robust fraud and anomaly detection systems.

Building these systems is a daunting challenge because labeled data is scarce—instances of fraud are typically very few for any product. Generative modeling can offer solutions to some of these problems by acting as an anomaly signal synthesizer or enhancer: the generative model either provides additional samples for rare classes or creates perturbed signals that explore rarer parts of the class distribution. Generative models can augment traditional labeling by creating realistic labels or infusing domain knowledge into a simpler model, thereby reducing the risk of real-life slippages owing to lack of realism in a labeling model. Potential applications include data-augmentation techniques for anomaly-detection and rare-class classifiers, adversarial-test creation, supplementary or auxiliary signal generation for multi-task-learning setups, and large volume labeling of applications or claims.

The widespread application of DevOps practices in software development, especially in the context of cloud-native architecture, has ensured rapid, and often painful, deployments of new models and business processes. These concepts can be applied to the design and deployment of pipelines for fraud detection or clustering in insurance. A DevOps-enabled pipeline can address questions related to the durability, extensibility, scaling, governance, and sustainability of such systems.

Fig 1: AI-Driven Insurance Intelligence

2. Background and Problem Statement

Growing investments in insurance fraud detection have not translated to more desirable outcome metrics, resulting in a critical question: can continuous enhancement of fraud-detection pipelines, made possible through artificial intelligence, generate robust models that can become integral to a DevOps-style fraud-detection pipeline? The answer is an emphatic yes. By architecting fraud-detection pipelines that detect and inject anomalies into fraud-detection models, fraud-detection algorithms in property and casualty insurance, health insurance, and auto insurance can be continuously enhanced using an agile-scale DevOps mechanism and novel alert design patterns.

AI-driven insurance intelligence leverages generative models in data synthesis and a DevOps architecture for building scalable fraud-detection pipelines across property and casualty, health, and auto lines of business in the insurance sector. Generative models, a branch of machine learning, learn data distributions and can synthesize new data points that match the learned distributions and construction objectives. These models enable augmentation of real fraud-alert records, real fraud-alert records can be employed to detect other fraud signals. Integrating these concepts can result in tooling that an insurance organization can expose to its fraud strategy team; when fraud signals for fraud-alert records are missing or reduced, the tooling can be used to generate additional fraud-signal records.

2.1. Research design

To illustrate the core ideas and answer the research question, a specific property-casualty insurance claim-related problem is analyzed, an end-to-end detection pipeline is architected, and system design is elaborated.

Anomalous claims and claims fraud are detected with a generic machine-learning-based approach with insufficient data. As an alternative, synthetic claims that show one or more characteristics of being anomalous are generated; these constitute a new class of SIAM signals. Auxiliary-choice models infer normative values for irrelevant or optional attributes of legitimate claims. Auto-Insurance-Fraud-Sample and TrueSkill datasets bolster the model for insurance forecasting; strategic, tactical, and operational insurance fraud detection—from alert generation through triage and risk scoring to focused investigation—are implemented using multiple machine-learning models.

Generative AI in detecting anomalous claims and claims fraud is integrated with product development and system monitoring, augmenting fraud-analysis capabilities and priorities. AI DevOps pipelines encompassing multiple models are designed to ingest, cleanse, engineer, and serve data across any number of use cases, pipelines, and model classes for data-driven decision-making. Such pipelines are modular and extensible, with a dedicated orchestrator, guaranteed data lineage, connected feature stores, continuous model testing, monitoring, and operational dashboards.

Equation 1: Generative model for synthetic fraud-related samples

Step 1

Assume a latent-variable generative process:

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING
ISSN: 3067-1140
VOLUME: 02 ISSUE: 03
RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

$$z \sim p(z), x \sim p_\theta(x \mid z)$$

This means:

- first sample a hidden latent code z,
- then generate a claim/sample $x$from that latent code.

Step 2

We want the overall probability of observing x, regardless of which latent $z$produced it.

By the law of total probability:

$$p_\theta(x) = \int p_\theta(x, z)\, dz$$

Step 3

Use the chain rule of probability:

$$p_\theta(x, z) = p_\theta(x \mid z)\, p(z)$$

Substitute into the previous equation:

$$p_\theta(x) = \int p_\theta(x \mid z)\, p(z)\, dz$$

Final Equation 1

$$\boxed{p_\theta(x) = \int p_\theta(x \mid z)\, p(z)\, dz}$$

3. Generative Models in Insurance Fraud Detection

Generative models in insurance fraud detection provide scalable, robust, and auditable data-synthesis and analytics pipelines. Intelligence-driven investments harness artificial intelligence, especially machine learning, to automate and optimize insurance-related processes. Among the many applications in the industry, detecting illicit activity such as fraud has gained traction. Unlike traditional external solutions, investment in internal capabilities enables creation and overall control of the models, allowing their adaptation to business requirements and the engineering of insurance-specific governance.

Generative modeling refers to a broad category of machine-learning techniques whose goal is to learn how to synthesize data resembling a reference dataset. Generative models can thus be used to generate realistic synthetic data or provide supervised and unsupervised learning signals during model development and evaluation. Diverse applications may arise at several stages of the

ISSN: 3067-1140
AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING
VOLUME: 02 ISSUE: 03
RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

fraud prediction pipeline, including enrichment of the training set for anomaly-detection models, adversarial testing of detector robustness, and matching domains at training and inference time.

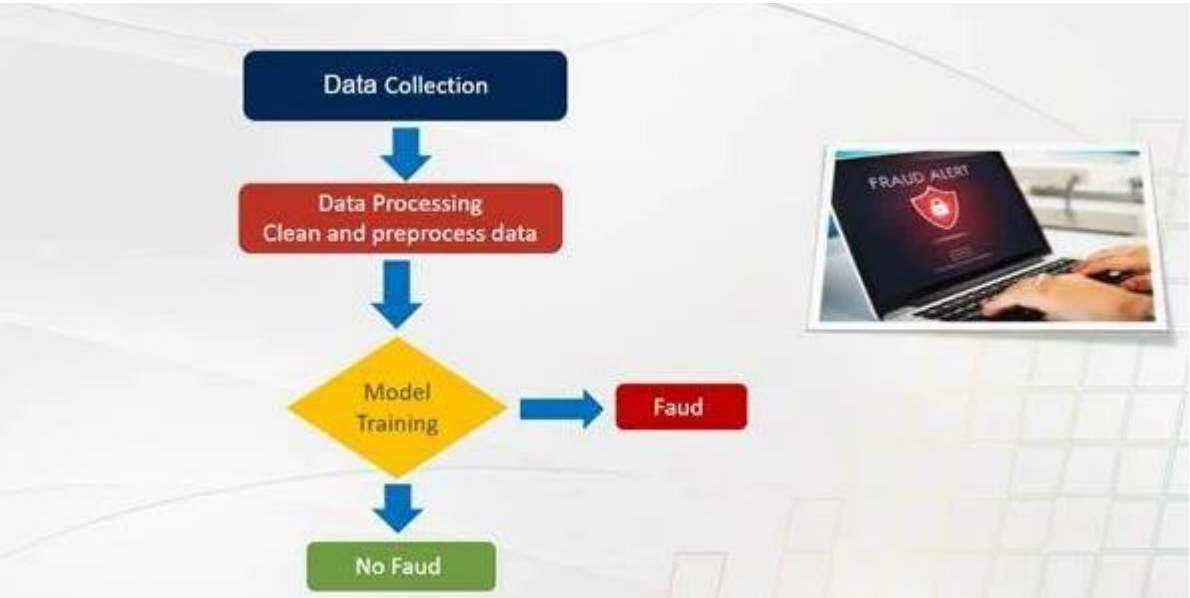


Fig 2: Generative AI Reduce Fraud Detection in Insurance Claims

3.1. Fundamentals of Generative Modeling

Generative modeling encompasses a diverse array of machine learning models that learn to generate new instances of data that belong to the same distribution as a training dataset. These models include modeling frameworks such as pixel-cnn, pixel-snn, variational autoencoders, generative adversarial networks, diffusion models, and more. Unlike discriminative models that focus solely on predicting labels for real-world observations, generative models hold the potential to fill in missing data, augment training data with synthetic instances, create adversarial test cases, and even invalidate remaining assumptions with detected realism flaws. Two aspects warrant further discussion: risk and evaluation.

Risks arise when reliance is placed on synthetic data without careful evaluation or when sufficient realism is not achieved. Detecting a lack of realism should be a core competency of any generator. The system design answers the resilience question by ensuring existing workflows do not depend on generative signals. For many types of training data, being able to add even a small quantity of fresh synthetic data at a higher rate than real data should, in principle, enable rapid anomaly detection signal update cycles. This is particularly relevant for pipelines where retraining is prohibitively resource-intensive or cannot keep pace with new signal detections. By introducing a non-differential latent anomaly approach—the key enabler of a governance-compliant feedback loop—latency considerations make display realism the grounding concern.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

Reliance on synthetic data introduces meaningful risks when its fidelity is not rigorously validated, as even subtle deviations from real-world distributions can propagate errors through downstream systems. A critical capability of any generative framework, therefore, is the ability to detect and correct for insufficient realism, ensuring outputs remain representative and trustworthy. From a systems design perspective, resilience is achieved by decoupling core operational workflows from dependence on generative signals, thereby preventing cascading failures when synthetic inputs degrade. At the same time, the controlled introduction of small volumes of high-quality synthetic data—generated at a faster cadence than real data acquisition—can accelerate anomaly detection cycles and improve responsiveness to emerging patterns. This becomes especially valuable in environments where full retraining is computationally expensive or operationally infeasible. The integration of a non-differential latent anomaly mechanism further strengthens this framework by enabling governance-compliant feedback loops without exposing sensitive gradients or internal model dynamics. However, because such systems often operate under strict latency constraints, ensuring perceptual and statistical realism at the point of output display remains the fundamental anchor for maintaining both performance and trust.

Aspect	**Derived content from the article**
Title / Theme	AI-driven insurance intelligence using **generative AI + DevOps/MLOps** for scalable fraud detection
Main objective	To show how generative models can synthesize fraud-related signals and how DevOps practices can operationalize those signals in production fraud-detection pipelines
Business problem	Insurance fraud detection suffers from **scarce labeled fraud data**, evolving fraud patterns, and difficulty scaling robust detection systems
Core proposition	Use **generative models** to create synthetic anomaly/fraud signals and embed them into a **DevOps-enabled pipeline** for continuous training, testing, deployment, and monitoring
Insurance lines covered	**Property & Casualty, Health, and Auto insurance**
Role of generative AI	Data augmentation, rare-class synthesis, adversarial test generation, supplementary fraud signals, robustness enhancement

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

Aspect	Derived content from the article
Role of DevOps / MLOps	Modular pipeline design, CI/CD, orchestration, feature-store support, lineage tracking, monitoring, retraining, governance
Expected benefits	Better fraud detection coverage, improved scalability, continuous model improvement, explainability support, governance and compliance readiness
Main concern areas	Realism of synthetic data, false positives/false negatives, model fairness, transparency, privacy, regulatory compliance

Table 1: Research overview and conceptual foundation

3.2. Applications in Fraud Signals Synthesis and Detection

Generative modeling provides diverse application opportunities for both supervised and unsupervised fraud detection signals. The simplest use case consists of augmenting training data sets with counterfactual or near-canonical examples, thereby improving the detection performance of downstream models. More subtly, generative models open possibilities to simulate extreme anomalies or adversarially invalid inputs for stress-testing fraud detection pipelines. Finally, generative modeling techniques that prioritize realism yield sensitive joint distributions over features relevant for fraud detection. Leveraging them enables the bridging of gaps in the support of real data and yields natural candidates for additional features providing signals of fraudulent behavior.

The integration of these generative modeling applications within insurance fraud detection pipelines must satisfy thus far unmentioned considerations. The relevance of detection must extend beyond performance metrics: indeed, uninterpretable or unexplainable deep fakes are already being generated. If validation and detection hinges primarily on visual realism, simple image-processing techniques will detect and mitigate the risk. Transformations analogous to those on audio synthesis merely because of humans hearing audio distinctions are already being relayed via text, and consequently also visual domains.

Equation 2: Fraud probability from claim features

Let:

- $x \in \mathbb{R}^d$ be the feature vector,
- $y \in \{0,1\}$, where $1 =$ fraud and $0 =$ non-fraud.

ISSN: 3067-1140
AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING
VOLUME: 02 ISSUE: 03
RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

A common probabilistic classifier is logistic regression.

Step 1

Define a linear score:

$$s = w^{\top}x + b$$

Step 2

Convert this unrestricted score into a probability in $(0, 1)$using the sigmoid function:

$$\sigma(s) = \frac{1}{1 + e^{-s}}$$

Step 3

Substitute $s = w^{\top}x + b$:

$$P(y = 1 \mid x) = \sigma(w^{\top}x + b)$$

Final Equation 2

$$\boxed{P(y = 1 \mid x) = \frac{1}{1 + e^{-(w^{\top}x+b)}}}$$

4. DevOps for Scalable Fraud Detection Pipelines

DevOps capabilities enable scalable and robust pipelines for fraud detection across lines of insurance. Modular architecture fosters reusable components designed for quick delivery, and orchestration capabilities support concurrent tasks with defined dependencies. Data engineering patterns associated with a feature store provide consistent and governed inputs, while model deployment and integration support CI/CD. Pipeline monitoring facilitates adjustment and ensures compliance.

Robust and scalable fraud-detection pipelines are essential for business assurance in lines of insurance affected by fraud. These pipelines can be engineered using a DevOps-like approach to reduce delivery and maintenance costs, enable rapid experimentation, and support interaction with insurance customers. Fraud signals are derived or detected using multiple fraud-related models, many of which are not based on the latest data. Models not yet part of a production system can be used for batch inference as signals, freeing development resources and allowing delayed integration into business processes.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

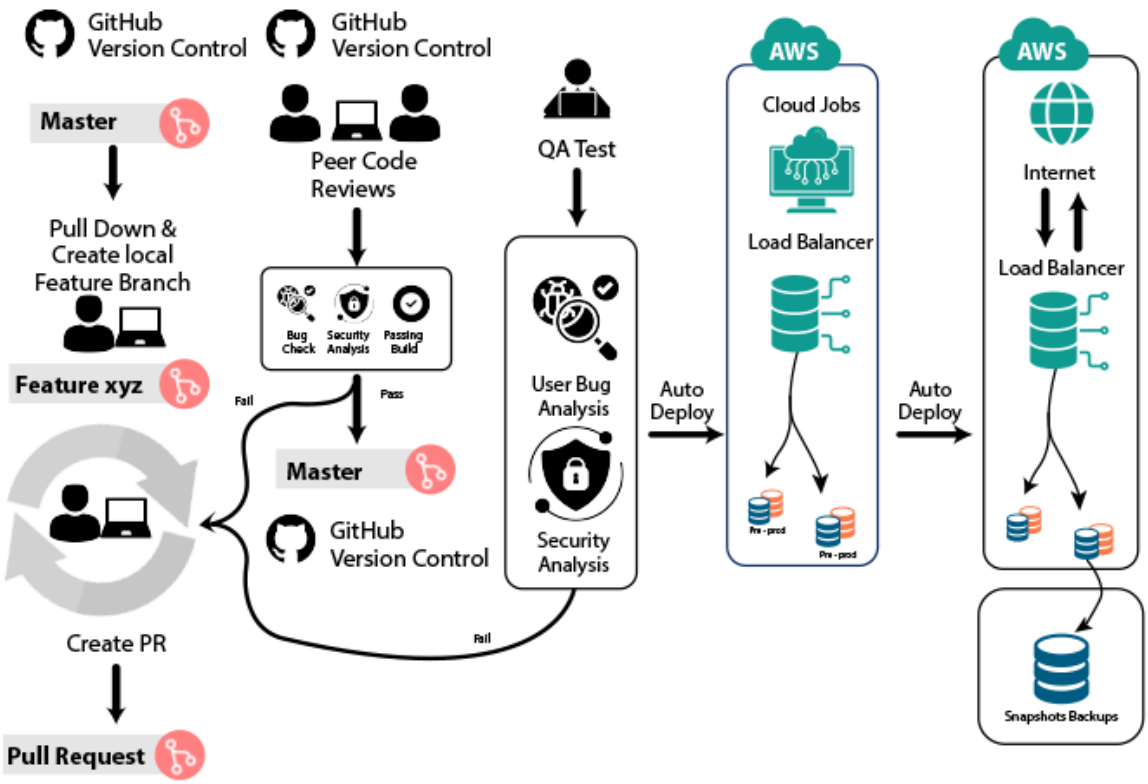


Fig 3: DevOps Pipeline and Methodology

4.1. Architectural Considerations

Architectural choices for a DevOps-enabled fraud-detection pipeline must accommodate diverse constituent models, which require specialized data and feature engineering or support different types of inference signs. Consequently, individual components should expose well defined in- and out-interfaces and orchestrate data requests and responses in a service-oriented manner, while the overall pipeline provides evidence-of-concept, call-chain triggering, warehouse buffering, and cross-component management.

Data-engineering patterns for pipeline support may include accelerated sample generation, detection in numerical data anomaly bands, preprocessing of textual data on demand for single-sample or very low-latency use, and signal scoring with dedicated feature engineering. The settling of such cross-component workflows is essential to address different training cycles and retrain triggers (e.g., upon new samples or delayed provenance). A feature store capable of serving temporary sample-extraction functionality also supports lineage and lifecycle management.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

4.2. Data Engineering and Feature Lifecycle

Data acquisition and engineering underpin all fraud-detection models. They ensure that both the training for supervised models and the signal extraction for unsupervised models are based on data that are relevant, fresh, and of sufficient quality. A central feature store maintains ready access to persisted features and data lineage. Matching data sources are tracked and indexed in a business feature store, reconciling business semantics with technical details. The automated pipelines and augmenters for feature calibration– training, monitoring, validation, and testing– are solidified and monitored.

Data engineering patterns for feature stores are sound: feature stores provide appropriate business terminology and semantic tagging; lineage specifies how features are calculated; and lifecycles track when features need to be retrained, monitored, or validated. Continuous integration–continuous delivery pipeline principles govern the engineering of features and feature stores. Beyond the feature engineering, these patterns are extended to the data acquisition workflows. Data preparation defines the core characteristics of the shared data space– gold, silver, and bronze data sources– and establishes standardized data quality metrics and remediation checks. These patterns provide the quality and speed of preparation required for fraud detection.

Equation 3: Cost-sensitive binary cross-entropy for rare fraud classes

Let:

$$\hat{p}_i = P(y_i = 1 \mid x_i)$$

For one sample, ordinary binary cross-entropy is:

$$\ell_i = -[y_i \log \hat{p}_i + (1 - y_i)\log(1 - \hat{p}_i)]$$

Step 1

Introduce different weights for fraud and non-fraud:

- w_1 for fraud samples,
- w_0 for legitimate samples.

Step 2

Weight each term:

$$\ell_i = -[w_1 y_i \log \hat{p}_i + w_0(1 - y_i)\log(1 - \hat{p}_i)]$$

Step 3

Average over N training samples:

ISSN: 3067-1140
AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING
VOLUME: 02 ISSUE: 03
RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

$$L = \frac{1}{N}\sum_{i=1}^{N} \ell_i$$

Substitute ℓ_i:

$$L = -\frac{1}{N}\sum_{i=1}^{N} [w_1 y_i \log \hat{p}_i + w_0 (1 - y_i) \log(1 - \hat{p}_i)]$$

Final Equation 3

$$\boxed{L = -\frac{1}{N}\sum_{i=1}^{N} [w_1 y_i \log \hat{p}_i + w_0 (1 - y_i) \log(1 - \hat{p}_i)]}$$

4.3. Model Deployment, Monitoring, and Governance

Data science pipelines for production machine learning comprise many discrete, interdependent components such as data ingestion, model training, and inference. For complex systems like insurance fraud detection—many of whose components leverage deep neural networks, which are known to be data-hungry and prone to overfitting—assessing the health of these components, spanning the entire lifecycle of training, retraining, and inference, is critical to maintaining detection performance and limiting operational risk. Moreover, pipelines must also integrate feedback loops to facilitate continuous improvement.

The strategy for production machine learning in AI-driven insurance intelligence is inspired by the DevOps movement in software development and operations. Under DevOps, modular components (e.g., version control, continuous integration, and cloud services enabled) are combined with orchestration tools to automate testing, deployment, and monitoring. Translating these ideas into the context of production machine learning entails describing the components that form a complete machine learning system for a specific application, the engineering patterns employed in the components that constitute the supporting data- and feature-engineering plumbing, and the management of training, retraining, monitoring, and governance of the machine-learning models. Following this outline, the discussion focuses primarily on fraud detection models in property and casualty insurance fraud prediction and detection, with consideration given to the supporting components involved in engineering the underlying data.

5. System Design for End-to-End Fraud Detection

The system design encompasses the entire fraud-detection process, detailing the ingestion of data from multiple sources, its preprocessing (cleansing and normalization), the feature-engineering methods implemented on the data, and the corresponding tasks associated with training the fraud-detection models.

ISSN: 3067-1140
AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING
VOLUME: 02 ISSUE: 03
RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

Data used for detecting fraud signals can be sourced from multiple lines of insurance, such as property and casualty, health, and auto. Ingestion pipelines need to be defined to fetch data on a continuous basis. Models associated with each data source perform cleansing operations according to an algorithm specified by business stakeholders. Some simple cleansing tasks could include dropping rows with a fraudulent flag already labelled by humans, or dropping duplicate records. Normalization tasks, too, can be defined based on business rules, such as ensuring the values for the "age" or "months of driving experience" columns must lie between 18 and 100, or following the product's rating curve, besides being bounded above and below; records that do not satisfy these rules can then be replaced by null values or dropped. The normalised data can be stored in a feature store, ready for training or inference, depending on fraud centers' needs.

Once the data-processing pipelines are in place, the focus can shift towards training fraud-detection models in a controlled manner. The rules for performing training must be agreed upon by the respective fraud teams. As a starting point, a holdout dataset can be created to allow quarterly or biannual retraining of the detection models, along with a strategy for ensuring the models are kept up to date. When approaching production deployment of the models, the strategy for doing so, including whether to use real-time scoring or logic that groups inferences together for batch processing in spite of real. At a minimum, three criteria must be met: the latency of each inference cannot exceed one hour, the volume of inferences in a group cannot exceed 5000, and a processing time of less than 10 minutes needs to be ensured. An automatic feedback loop also needs to be defined, triggering model retraining when sufficient new labels are available or when the signal's detection performance falls below acceptable levels.

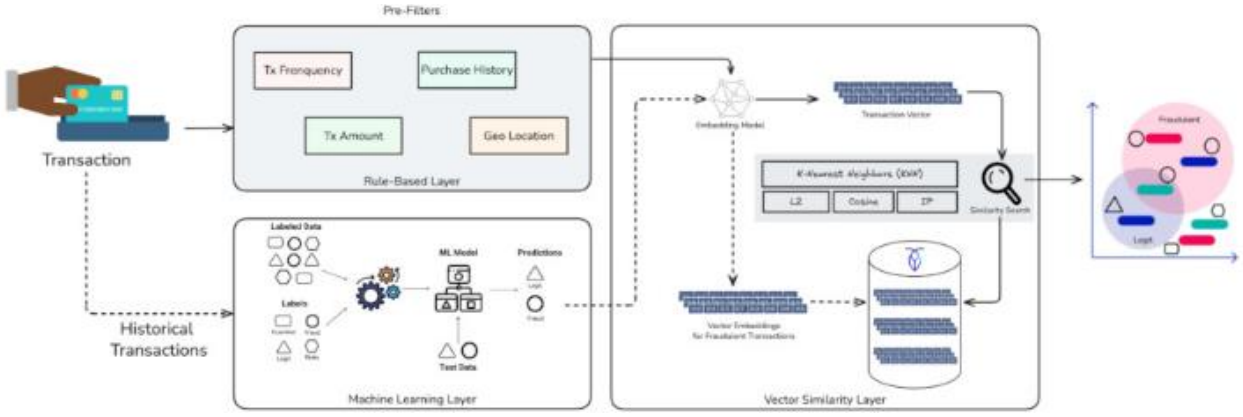


Fig 4: System Design for End-to-End Fraud Detection

5.1. Data Ingestion and Preprocessing

Insurance fraud detection relies on diverse data sources and signals, including claims, policyholder information, third-party intelligence, and customer behavior. Fraud detection datasets may have limited representativeness, leading to overfitting and unreliable real-world performance. Data ingestion and preprocessing pipelines address these challenges by cleaning, normalizing, transforming, and augmenting data. Data ingestion encompasses component orchestration and implementation, while preprocessing includes cleansing, normalization, typecasting, and feature engineering.

ISSN: 3067-1140
AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING
VOLUME: 02 ISSUE: 03
RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

In property and casualty insurance, data sources include claims, policyholders, vehicles, drivers, insurance coverage, and posed/third-party images. In health insurance, claims data is supplemented by healthcare providers. In auto insurance, mobile telemetry and online/offline advertising behavior augment information on policyholders, insurance coverage, and claims. Data cleansing removes duplicates, address normal forms improve geolocation, and time zones support time series. Multimodal fraud detection relies on cleaned raw-data ingestion; fraud-increasing signals check probability distributions and past model decisions; and signal-boosting features improve detection performance.

Pipeline stage	Purpose	Key activities / design elements	Expected output
Data ingestion	Collect fraud-relevant data from multiple insurance sources	Pull claims, policyholder data, provider data, vehicle data, third-party intelligence, telemetry, images, customer behavior data	Raw operational data
Data cleansing & normalization	Improve data quality before modeling	Remove duplicates, fix formats, enforce business rules, handle invalid values, standardize address/time-zone fields	Cleaned and normalized datasets
Feature engineering	Create usable fraud-detection signals	Derive business and technical features, maintain semantic tagging, prepare supervised and unsupervised model inputs	Curated feature sets
Feature store & lineage	Ensure reusable and governed data assets	Central storage of features, versioning, lineage, lifecycle tracking, support for training/inference consistency	Reliable feature repository
Generative signal synthesis	Fill rare-event and anomaly gaps	Generate synthetic fraud-like samples, perturb rare classes, create adversarial or edge-case inputs	Synthetic fraud/anomaly signals
Model training	Build fraud-detection and anomaly models	Train binary classifiers, anomaly scorers, signal-specific models, establish holdout datasets and retraining cadence	Trained models

Table 2: End-to-end fraud detection pipeline architecture

5.2. Model Training and Evaluation

Training workflows for fraud-detection models are defined, along with the evaluation protocols and holdout schemes during model development. Detection, anomaly classification, and anomaly scoring models require separate training datasets, each addressing a distinct problem. During development, a different holdout set is used per model type, while training, evaluation, and holdout splits are consolidated for models that need to detect both real-time and batch signals.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

Training detection models in payload-feature-space can exploit the very high volume and sparsity of the dataset. Input features that are of limited predictive power can be included in the monitoring dashboards but dropped for model training. Patterns of engineering effort that deliver models with desirable fragility are discussed. These models are usually able to deliver results in a low-latency window; the inference cycle for such signals resembles a traditional micro-service call. For the other models, batch processing is usually preferred over real-time; the throughput can be handled using data splits or parallel processing.

Batch-processing latency targets are not too stringent, but throughput requirements are essential. Proper partitioning and sizing of the job can ensure execution within SLA limits. Pipeline failures usually imply that the job cannot be retried, and feedback loops that trigger model retraining are needed. Such trigger configurations can be based on supervised signal-generation processes like back-testing or re-labeling, or be based on unsupervised monitoring processes like data drift or signal drift.

Equation 4: Anomaly score from negative log-likelihood

Step 1

If a sample is very likely under the learned normal/fraud-supporting distribution, then $p_\theta(x)$is large.

Step 2

If a sample is unusual, $p_\theta(x)$is small.

To convert "small probability = more anomalous" into an increasing score, take negative log:

$$A(x) = -\log p_\theta(x)$$

Step 3

Why does this work?

- if $p_\theta(x) = 1$, then $A(x) = 0$
- if $p_\theta(x) \to 0$, then $A(x) \to \infty$

So rarer points receive larger anomaly scores.

Final Equation 4

$$\boxed{A(x) = -\log p_\theta(x)}$$

5.3. Real-Time Inference and Batch Processing

Performance comparison of real-time inference and batch processing for insurance fraud detection examines latency targets, throughput requirements, and evaluation criteria.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

An insurance organization has fraud detection requirements at different latencies. For urgent cases, real-time inference is desired with a defined number of predictions per minute. Other scenarios permit a batch processing approach encompassing more than a dozen fraud detection models. Inference speed and expected volume of records to be inferred simultaneously are assessed in both settings. Latency targets are defined to meet user expectations. For sensitive projects, prediction speed needs to be maximized. For fraud detection, no one likes to be told it is predicted that they might commit fraud, as it implicitly accuses them of being dishonest. Therefore, these fraud types require action as soon as the risk is identified, e.g., because of buying an insurance product on a highly non-standard way or abusing a claim settlement process. When a batch process is used, either for all detection models or just for a subset related to a defined time horizon (e.g., at the end of each day), anticipation of fraud maximizes the result of the action taken afterwards.

6. Evaluation, Risk, and Compliance

Ensuring robust evaluation, mitigation of risk, and adherence to compliance considerations requires the deployment of comprehensive frameworks designed specifically for these areas. Given the nature of fraud, the lives being insured, the implications of false-positives on end-users and mitigatory guidelines, and the path to detection and alerts being sensitive, using metrics that evaluate the suppression of bias and the falsification of insurance coverage is key to a successful evaluation. Addressing false-negatives is typically the reason behind the evaluation framework. Beyond performance, the fairness of the model is just as important since such signals can independently serve as a positive hindrance when talking about detection.

Nevertheless, the explainability of these models (especially in the Data-Driven Anomaly detection category) needs to be suited to the target audience because it is often preferred to have these predictions presented in a frontal manner to avoid prompt suspicion while delivering the relevance of the samples in a more casual tone. Regulatory compliance also becomes critical due to the sensitive nature of health insurance — the data is not only able to impersonate real patients and be used maliciously, but it also simulates images of organs without these… again putting people in serious risk. It is therefore very important to maintain data privacy through governance practices on-model, on-data and on-usage. Audit trails from tracking systems such like the ones provided by MLFlow are key to NLP projects about rule generation.

6.1. Evaluation Frameworks and Metrics

Evaluation, risk, and compliance considerations are crucial, as they provide the foundations for rigorous model testing, safeguarding decisions, and addressing external concerns. A comprehensive framework could involve a set of evaluation dimensions and complementary metrics, ensuring that specific needs are met during development, deployment, and operational phases. Evaluation is essential to all reliable models, while risk-related aspects require additional scrutiny, and compliance constitutes further safeguards.

The efficacy of fraud detection models demands rigorous scrutiny against an ensemble of metrics that determine detection performance, calibration, and fairness. Targets for true-positive and false-positive rates can be derived from an organization’s fraud tolerance, with an optimal cutoff point identified for Receiver Operating Characteristic curves. Fairness criteria require that detection rates reflect actual crime prevalence across groups defined by gender, age, nationality, or other protected characteristics. Furthermore, stakeholders may request explanations justifying suspiciousness scores, with explanations preferably communicated in easily understood terms, potentially supported by visualizations.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

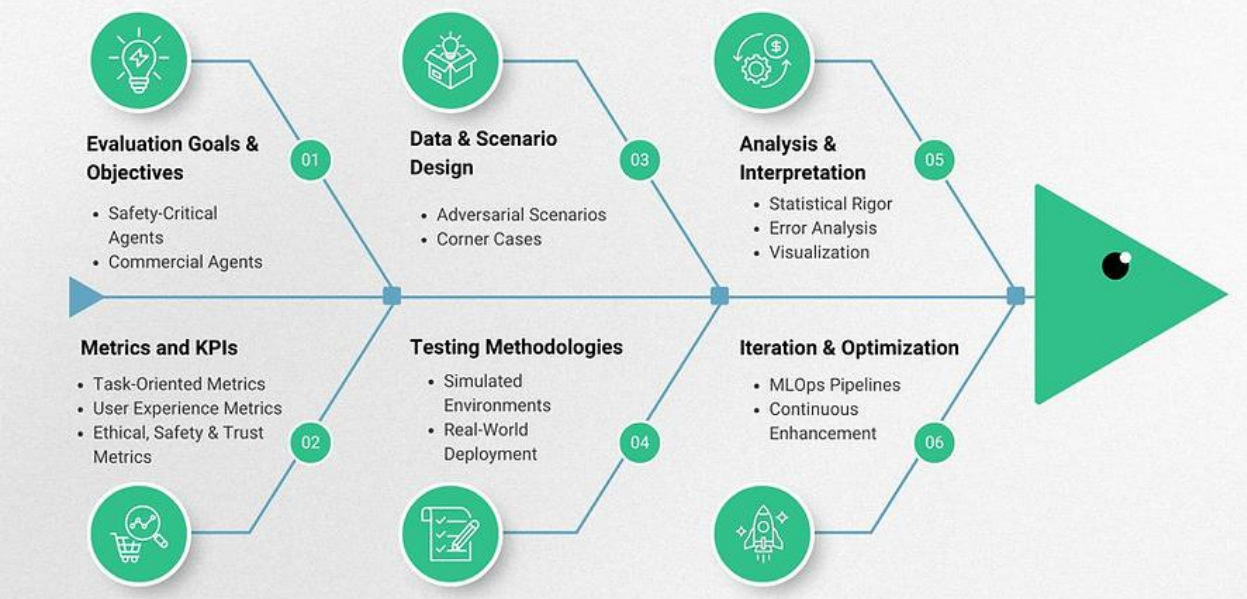


Fig 5: Agent Evaluation Frameworks

6.2. Explainability and Transparency

Complying with regulatory requirements, insurance providers need to guarantee that models are interpretable and their decisions can be justified to external stakeholders such as customers and insurers. Furthermore, it is important to maintain fairness among candidates from different sensitive demographic groups. An adequate explanation for a rejected insurance proposal needs to identify the corresponding risk signal and its root cause: for instance, whether the one associated with an implicitly borrowed image is a potential sign for money laundering caused by a chaotic preparation for the journey, or the presence of a business camera indicates a side job, either of which would affect the declared activity type.

In addition to serving risk management requirements, explainability during operations is also critical for the operators responsible for interpreting models in order to provide advice to the business on when additional steps are required in the fraud management process. Hence, models need to be designed in a manner that allows risk signals to be indicated and whose values summarized as part of the explanation. The combination of explainability and systematic testing of model coverage and loss of expressiveness adds a layer of reliability for the operators by improving their understanding of model behavior and by providing supporting tools.

Equation 5: Decision rule with unequal fraud costs

Let:

- $p = P(y = 1 \mid x)$

ISSN: 3067-1140
AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING
VOLUME: 02 ISSUE: 03
RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

- C_{FN}= cost of false negative
- C_{FP}= cost of false positive

We compare two actions:

- predict fraud
- predict non-fraud

Step 1: Expected cost if we predict fraud

If we predict fraud, the only mistake is false positive, which happens with probability $1 - p$.

$$\text{Cost(predict fraud)} = (1 - p)C_{FP}$$

Step 2: Expected cost if we predict non-fraud

If we predict non-fraud, the only mistake is false negative, which happens with probability p.

$$\text{Cost(predict non-fraud)} = pC_{FN}$$

Step 3: Choose fraud when its expected cost is smaller

$$(1 - p)C_{FP} < pC_{FN}$$

Step 4: Solve for p

Expand left side:

$$C_{FP} - pC_{FP} < pC_{FN}$$

Bring p-terms together:

$$C_{FP} < p(C_{FN} + C_{FP})$$

Divide both sides:

$$p > \frac{C_{FP}}{C_{FN} + C_{FP}}$$

Final Equation 5

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING
ISSN: 3067-1140
VOLUME: 02 ISSUE: 03
RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

$$\text{Predict fraud if } P(y = 1 \mid x) > \frac{C_{FP}}{C_{FN} + C_{FP}}$$

6.3. Regulatory and Ethical Considerations

Many of the merits of AI in criminal detection risk becoming demerits when the models are deployed in production and used without proper supervision. An extensive layer of evaluation, safeguarding, and documentation is thus essential to ensure that the deployment of an AI model does not create more risks than the risk that the model aims to mitigate.

First, models for insurance crime detection need to be evaluated thoroughly and with the same rigor as that for the crime signal that they aim to recognize. This includes detecting frailty both in sensitivity and specificity. The cost of false negatives is usually higher than that of false positives, but false positives may require investigation, causing delays and costs that could have been avoided. Similarly, the calibration of the model is assessed to minimize the cost associated with selecting a threshold. The price of running the model also needs to be factored in, particularly in applications of real-time detection, and several performance metrics, such as area-under-curve, latency, and throughput, should thus be considered together when evaluating the model. The output also needs to be transparent enough for a domain expert to validate the conclusion (e.g., Which are the strongest indicators of predicted fraud?).

Case study	Data characteristics	Fraud-detection approach	Reported / described outcomes	Key operational insight
Property & Casualty	~317,000 sample claims; fraud rate about ~5%; mixed claim types and fraud motives	Binary classification with batch-oriented random-forest modeling; multiple pipelines for different fraud ensembles	Architecture reported as operational across six pipelines; useful signals include loss volume, vehicle value, public transport use	P&C fraud is heterogeneous, so multiple specialized pipelines/models are needed rather than one universal detector
Health Insurance	Five years of claims data for procedures and beneficiaries; rare-event and scheme-detection context	Neural networks and extreme gradient boosting; balanced training data created from real-world unlabeled data	Around **91% accuracy** for scheme fraud detection and **88% accuracy** for claimant fraud detection	Strong historical data and interpretability features are essential because health fraud is sensitive and compliance-heavy
Auto Insurance	~420,000 claims from an Australian insurer; theft, hail damage, accidents; severe class imbalance	Two models trained on flagged and broader claims; SMOTE + oversampling to rebalance; AUC-focused evaluation	Reported **AUC of 0.94 and 0.92** for the two models	Auto fraud models must evolve with changing crime patterns and claim behavior; restated claims can act as retraining triggers

Table 3: Derived comparison of the three insurance case studies

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

7. Case Studies and Practical Implementations

Evidence from distinct lines of insurance establishes the feasibility of scalable, end-to-end fraud-detection pipelines—specifications are now being implemented across organizations, often as proof of concept in higher-risk areas—Hospital insurance fraud, with many possible signs, has shown a considerable depth of pipeline, involving multiple analyses supported by diverse backgrounds.The synopses below outline model data, methodology, results, and lessons in property and casualty, health, and auto insurance.

7.1. Property and Casualty Insurance The insurance dataset contains ~317,000 sample claims, with an average fraud rate of ~5%.The problem has been framed as a binary classification to identify real claims from fraudulent ones for batch processing with random-forest models. Linear-scaling features, involving the volume of loss, vehicle value, and public transport use, are particularly significant. Such a low up-front cost enables the real-time batch processing of fraud-model predictions.Several interesting signals are yet to be integrated; further insights are possible by modelling fraud risk and detection probabilities concurrently.

7.1. Case Study: Property and Casualty Insurance

Detecting property and casualty insurance fraud presents multiple challenges. The diversity of fraud schemes—the product of different claim types and fraud motivations—demands multiple models, each focused on a single aspect. For example, criminals often attempt to stage accidents involving several vehicles. Some even combine property damage with bodily injury. As a result, staged accidents require a joint analysis of Injury and Property Damage data, preserving the relationship between these components. Detecting such frauds is challenging, as the signal resides in the relationship between two datatypes. Terrorism and Foreshortened-Life frauds represent outliers in the different domains and require separate detection systems.

The architecture tackles property and casualty fraud detection through six pipelines, each oriented toward a specific fraud ensemble. In accordance with the previously delineated architectural patterns, data flows cover ingestion, cleansing, training, testing, inference, and retraining inference engines. All newly developed pipelines are reliably operational and produce results that validate the architecture's underlying principles. At the same time, other property-and-casualty systems currently implemented in ad-hoc sequences of programmatic tasks lack orchestration logic.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

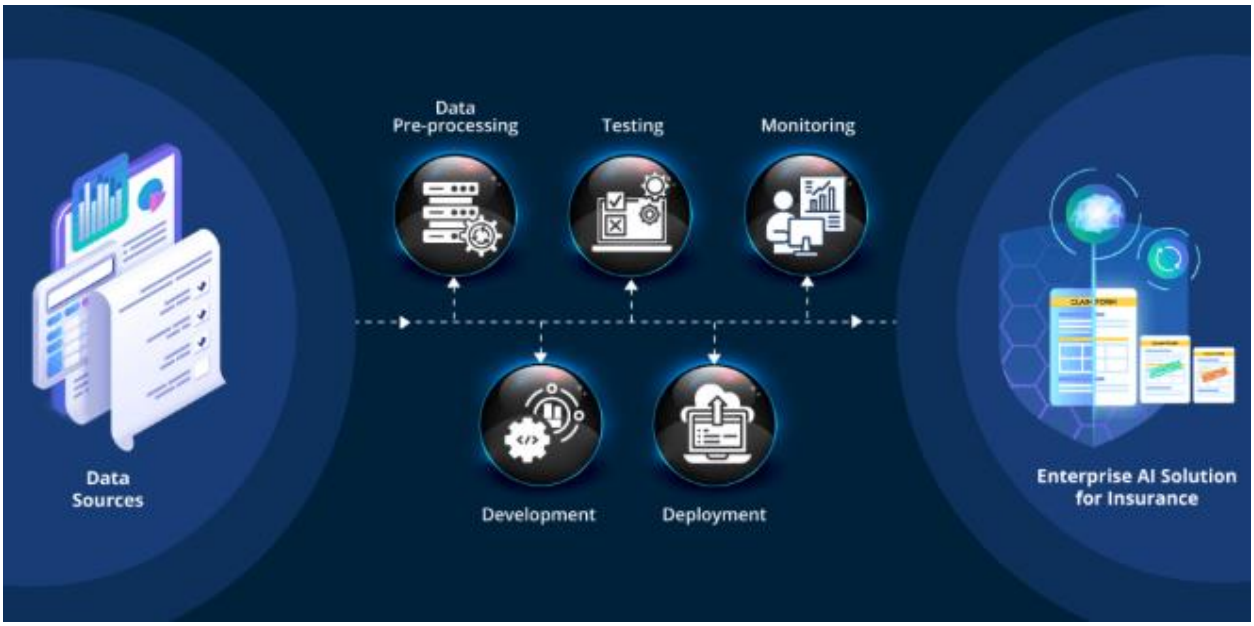


Fig 6: Property and Casualty Insurance

7.2. Case Study: Health Insurance

Insurance fraud is a frequent and costly issue in health insurance. Fraudulent claims, scheme providers, and indicted claimants can be detected, with detection capabilities often exceeding 90% accuracy when sufficient historical training data are available. Developing a health claim data fraud detection system involved detecting fictitious claimants or schemes, generating alarms, assessing detection performance, and obtaining relevant features. Five years of claims data for specific procedures and beneficiaries were used to train neural network and extreme gradient-boosting models. Balanced training data were generated from a real-world unlabeled dataset, and performance was validated on the remaining dataset. An interpretability algorithm identified key modeling features. The models outperformed random guessing, with the neural network achieving 91% accuracy in scheme fraud detection and the boosting model reaching 88% accuracy in claimant fraud detection.

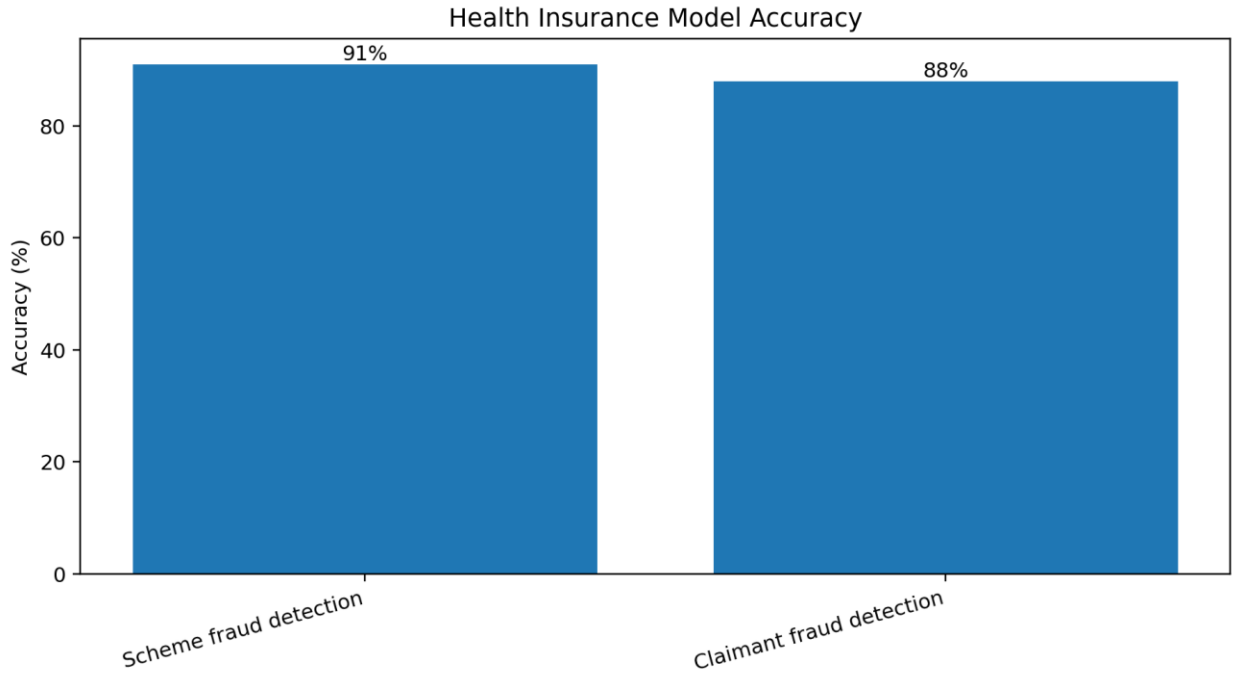
Insurers use detection systems during the claim approval process. Alarms indicate likely fraudulent providers or beneficiaries based on factors such as excessive bill amounts, unusual treatment durations, and outlier claim frequencies for specific diagnoses. A TRICARE-based system detects scheme fraud by indicating suspected bogus providers performing suspicious procedures that exceed a certain benchmark. Several Commanders' Methods detect fictitious claimants, typically having filed claims for only one or a few procedures. Major medical insurers often use False Claims Act Cases for legal investigations. Numerous prediction models classify fraud as a separate class, but detection systems monitor different fraud types jointly using diverse alarm models. Generating training samples for rare events poses a challenge, especially if many details remain privately held.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17



7.3. Case Study: Auto Insurance

The auto insurance case study focuses on claims data from an Australian insurer. About 420,000 claims, primarily for theft, hail damage, and accidents, serve as inputs, with a fraud label augmented with unconfirmed data. Two models—one for claims flagged by the insurance company and another for a broader spectrum—are trained, yielding AUC scores of 0.94 and 0.92, respectively. Signals vary and include the country of claim, the age of the insured vehicle at the time of claim, and geographic location. The class imbalance is mitigated through a smote technique followed by oversampling, resulting in a final dataset of approximately 50,000 records.

Sustainability of these results hinges on system evolution, especially in the crime-cumits of Australia and its impact on the detection capability in the prediction model. The restatement of a claim should act as a major trigger for retraining any detection model. That said, crime trends are not the only influencer of such models. Technological evolvement should also be an elemental driver in ensuring that new modus operandi of a claim are also captured in the mod els. Nevertheless, these two case studies—with their particular adoption of both proprietary data for training and proprietary implementation of machine learning algorithms, connecting neural networks across domains—and a finetuning approach that continues to see new models being built—paint a compelling picture of why the market has embraced generative AI."

8. Conclusions

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

AI-powered generative modeling is an increasingly important technology, with broad applicability across multiple industries. Appropriate adoption can provide scalable and cost-effective solutions that also adhere to compliance and governance requirements. Insurance fraud detection is a prime use case, as it relies on high-volume low-latency data-processing pipelines operating over dispersed data sources and organizations can afford to maintain underlying models.

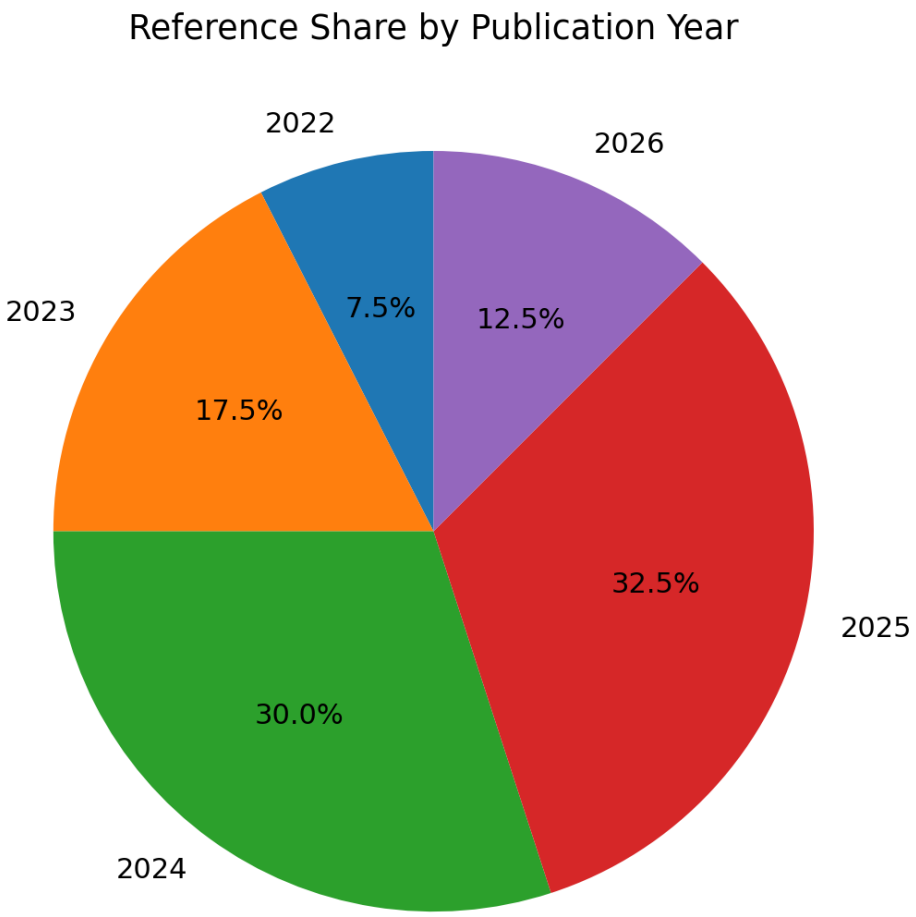
An extensive body of research exists on statistical- and machine-learning-assisted fraud signal detection, yet development of associated data-synthesis models remains limited. Generative modeling, particularly data synthesis, offers a possible means of generating fraud signals for such detection models. Further, this integration of generative modeling and DevOps principles enables pipelines for fraud-signal synthesis, monitoring, and detection to be governed in a robust and scalable manner. Although extensive fraud-aware-testing and adversarial-data-generation models have been developed throughout the insurance industry, such models have heretofore been limited by a lack of translation into elegant engineering solutions that govern together such dedicated pipelines.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17



8.1. Future Trends

Ongoing developments in generative models and DevOps practices are likely to unlock new capabilities for fraud-detection pipelines. Supported by substantial funding, the application of generative models is expanding beyond the initial web-image domain. Properties unique to generative models—including diversity, controllability, and ease of synthetic data generation—are being explored to address broader challenges of data availability for training, evaluation, and robustness testing across all domains of model development. These or other novel capabilities will certainly emerge in fraud-detection pipelines.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

Parallel technology evolution, powered by the DevOps mindset, is similarly transforming the design approach toward extensive, high-traffic systems using machine learning. Integration of the DevOps mindset, practices, and tools in fraud-detection pipelines will enhance their scalability, robustness, and governance. In particular, development and operations of all components in such pipelines—data engineering, feature engineering, model development (supervised and unsupervised), and end-to-end orchestration—will increasingly reflect the established principles of DevOps. Future fraud-detection capabilities are thus as likely to arise from the expanding application of the DevOps mindset as from breakthroughs in the underlying algorithms.

9. References

1. Gomes, C., Jin, Z., & Yang, H. (2021). Insurance fraud detection with unsupervised deep learning. Journal of Risk and Insurance, 88(3), 591–624.
2. 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.
3. Mangalampalli, B. M. Intelligent Data Profiling for Healthcare Data Lakes Using AI-Enhanced Analytics.
4. Banulescu-Radu, D., Cénac, P., & colleagues. (2024). Practical guideline to efficiently detect insurance fraud in the era of machine learning: A household insurance case. Journal of Risk and Insurance, 91(4), 867–913.
5. Lyu, L., Yu, H., & Yang, Q. (2022). Threats to federated learning: A survey. Proceedings of the IEEE, 110(5), 750–780.
6. Bommasani, R., Hudson, D. A., Adeli, E., et al. (2021). On the opportunities and risks of foundation models. arXiv Preprint.
7. Yandamuri, U. S. (2023). An Intelligent Analytics Framework Combining Big Data and Machine Learning for Business Forecasting. International Journal Of Finance, 36(6), 682-706.
8. OpenAI. (2023). GPT-4 technical report. arXiv Preprint.
9. Touvron, H., Lavril, T., Izacard, G., et al. (2023). LLaMA: Open and efficient foundation language models. arXiv Preprint.
10. Gottimukkala, V. R. R. (2020). Energy-Efficient Design Patterns for Large-Scale Banking Applications Deployed on AWS Cloud. power, 9(12).
11. Brown, T. B., Mann, B., Ryder, N., et al. (2021). Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877–1901.
12. Chowdhery, A., Narang, S., Devlin, J., et al. (2022). PaLM: Scaling language modeling with pathways. arXiv Preprint.
13. Kolla, T., & Kolla, S. K. (2023). FHIR-Based Real-Time Healthcare Analytics using Unsupervised Learning. International Journal of Future Innovative Science and Technology (IJFIST), 6(6), 11751.
14. Wei, J., Wang, X., Schuurmans, D., et al. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824–24837.
15. Ziegler, D. M., Stiennon, N., Wu, J., et al. (2022). Fine-tuning language models from human preferences. arXiv Preprint.
16. Peddi, R. K. (2024). AI-Based Workforce Analytics for SLA Governance and Uptime Assurance in Data Centers. Journal of Computational Analysis and Applications (JoCAAA), 33(08), 8589-8601.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

17. Raffel, C., Shazeer, N., Roberts, A., et al. (2021). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140), 1–67.
18. Vaswani, A., Shazeer, N., Parmar, N., et al. (2021). Attention is all you need. Advances in Neural Information Processing Systems.
19. Inala, R. AI-Powered Investment Decision Support Systems: Building Smart Data Products with Embedded Governance Controls.
20. Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2021). Federated optimization in heterogeneous networks. Proceedings of Machine Learning and Systems, 2, 429–450.
21. Sculley, D., Holt, G., Golovin, D., et al. (2021). Hidden technical debt in machine learning systems. Advances in Neural Information Processing Systems.
22. Pereira, K., Vinagre, J., Alonso, A. N., Coelho, F., & Carvalho, M. (2022, September). Privacy-preserving machine learning in life insurance risk prediction. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 44-52). Cham: Springer Nature Switzerland.
23. Breck, E., Cai, S., Nielsen, E., Salib, M., & Sculley, D. (2021). The ML test score: A rubric for ML production readiness and technical debt reduction. IEEE International Conference on Big Data.
24. Aitha, A. R. (2023). Cloud-Native Big Data AI/ML Framework for Risk Intelligence and Fraud Control in Banking and Insurance Ecosystems. Available at SSRN 6157967.
25. Kreuzberger, D., Kühl, N., & Hirschl, S. (2023). Machine learning operations (MLOps): Overview, definition, and architecture. IEEE Access, 11, 31866–31879.
26. Tamburri, D. A. (2022). Sustainable MLOps: Trends and challenges. Future Generation Computer Systems, 129, 144–157.
27. Mattaparthi, R. (2023). Connected Fleet Intelligence: Edge-Centric Analytics and Computer Vision for Predictive Manufacturing and Asset Resilience. International Journal of Advanced Research in Computer Science & Technology (IJARCST), 6(5), 9077-9088.
28. Zhou, Z. H. (2021). Machine learning for intelligent fraud analytics: A review. Artificial Intelligence Review, 54(8), 5891–5924.
29. de Prado, M. L. (2021). Advances in artificial intelligence for financial fraud detection. Journal of Financial Data Science, 3(4), 45–63.
30. Yandamuri, U. S. (2024). AI-Driven Decision Support Systems for Operational Optimization in Hospitality Technology. Metallurgical and Materials Engineering.
31. Debener, J., Heinke, V., & Kriebel, J. (2023). Detecting insurance fraud using supervised and unsupervised machine learning. Journal of Risk and Insurance, 90(3), 743–768.
32. Tumminello, M., Consiglio, A., Vassallo, P., Cesari, R., & Farabullini, F. (2023). Insurance fraud detection: A statistically validated network approach. Journal of Risk and Insurance, 90(2), 381–419.
33. Lu, J., Lin, K., Chen, R., Lin, M., Chen, X., & Lu, P. (2023). Health insurance fraud detection by using an attributed heterogeneous information network with a hierarchical attention mechanism. BMC Medical Informatics and Decision Making, 23, 62.
34. Nagabhyru, K. C., & Engineer, S. D. (2023). Unifying Data Engineering and Machine Learning Pipelines: An Enterprise Roadmap to Automated Model Deployment.
35. 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.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

36. Kreuzberger, D., Kühl, N., & Hirschl, S. (2023). Machine learning operations (MLOps): Overview, definition, and architecture. IEEE Access, 11, 31866–31879.
37. Mangalampalli, B. M. Generative AI Applications In Healthcare Data Mart Design And Optimization.
38. Tamburri, D. A. (2022). Sustainable MLOps: Trends and challenges. Future Generation Computer Systems, 129, 144–156.
39. Wazir, S., Kashyap, G. S., & Saxena, P. (2023). MLOps: A review. arXiv Preprint.
40. Fu, M., Pasuksmit, J., & Tantithamthavorn, C. (2024). AI for DevSecOps: A landscape and future opportunities. arXiv Preprint.
41. OpenAI. (2023). GPT-4 technical report. arXiv Preprint.
42. Touvron, H., Lavril, T., Izacard, G., et al. (2023). LLaMA: Open and efficient foundation language models. arXiv Preprint.
43. Kolla, S. K. (2022). Engineering Healthcare Data Infrastructures for Predictive Clinical Analytics and Evidence-Based Decision Making. International Journal of Engineering & Extended Technologies Research (IJEETR), 4(5), 5370-5380. |
44. Jiang, A. Q., Sablayrolles, A., Mensch, A., et al. (2023). Mistral 7B. arXiv Preprint.
45. Bai, J., Bai, S., Chu, Y., et al. (2023). Qwen technical report. arXiv Preprint.
46. Team Gemini. (2024). Gemini: A family of highly capable multimodal models. arXiv Preprint.
47. Mangala, N. (2024). Leveraging Microsoft Fabric lakehouse as an AI-ready data platform for enterprise analytics. Journal of Information Systems Engineering and Management.
48. Wei, J., Wang, X., Schuurmans, D., et al. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824–24837.
49. Chowdhery, A., Narang, S., Devlin, J., et al. (2022). PaLM: Scaling language modeling with pathways. arXiv Preprint.
50. Bommasani, R., Hudson, D. A., Adeli, E., et al. (2021). On the opportunities and risks of foundation models. arXiv Preprint.
51. Inala, R. Advancing Group Insurance Solutions Through Ai-Enhanced Technology Architectures And Big Data Insights.
52. Breck, E., Cai, S., Nielsen, E., Salib, M., & Sculley, D. (2021). The ML test score: A rubric for ML production readiness and technical debt reduction. IEEE International Conference on Big Data.
53. Sculley, D., Holt, G., Golovin, D., et al. (2021). Hidden technical debt in machine learning systems. Advances in Neural Information Processing Systems.
54. Lyu, L., Yu, H., & Yang, Q. (2022). Threats to federated learning: A survey. Proceedings of the IEEE, 110(5), 750–780.
55. Reddy, V. A. R. (2023). Orchestrating the Future Autonomous Healthcare Data Pipeline Management through Agentic AI Architectures. International Journal of Advanced Research in Computer Science & Technology (IJARCST), 6(2), 7979-7992.
56. Kairouz, P., McMahan, H. B., Avent, B., et al. (2021). Advances and open problems in federated learning. Foundations and Trends in Machine Learning, 14(1–2), 1–210.
57. Kreuzberger, D., Kühl, N., & Hirschl, S. (2023). Machine learning operations (MLOps): Overview, definition, and architecture. IEEE Access, 11, 31866–31879.
58. Tamburri, D. A. (2022). Sustainable MLOps: Trends and challenges. Future Generation Computer Systems, 129, 144–156.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

59. Kolla, S. K. (2023). Learning Health Systems Machine Intelligence for Clinical Prediction and Healthcare Optimization. International Journal of Advanced Research in Computer Science & Technology (IJARCST), 6(2), 7955-7966.
60. Wazir, S., Kashyap, G. S., & Saxena, P. (2023). MLOps: A review. arXiv.
61. Fu, M., Pasuksmit, J., & Tantithamthavorn, C. (2024). AI for DevSecOps: A landscape and future opportunities. arXiv.
62. Debener, J., Heinke, V., & Kriebel, J. (2023). Detecting insurance fraud using supervised and unsupervised machine learning. Journal of Risk and Insurance, 90(3), 743–768.
63. Soni, H., Perla, S., Maddela, S., & Kumar, U. (2025, November). Generative AI in Cloud CRM: Securing Intelligent Workflows in Multi-Cloud Environments. In 2025 IEEE 3rd Global Conference on Wireless Computing and Networking (GCWCN) (pp. 1-9). IEEE.
64. Tumminello, M., Consiglio, A., Vassallo, P., Cesari, R., & Farabullini, F. (2023). Insurance fraud detection: A statistically validated network approach. Journal of Risk and Insurance, 90(2), 381–419.
65. Lu, J., Lin, K., Chen, R., Lin, M., Chen, X., & Lu, P. (2023). Health insurance fraud detection by using an attributed heterogeneous information network with a hierarchical attention mechanism. BMC Medical Informatics and Decision Making, 23, 62.
66. Reddy, V. A. R., & Kolla, S. K. (1984). Infrastructure-As-Code Practices For Regulated Healthcare Cloud Environments. Metallurgical and Materials Engineering, 30 (4), 1028–1042.
67. OpenAI. (2023). GPT-4 technical report. arXiv.
68. Touvron, H., Lavril, T., Izacard, G., et al. (2023). LLaMA: Open and efficient foundation language models. arXiv.
69. Team Gemini. (2024). Gemini: A family of highly capable multimodal models. arXiv.
70. Jiang, A. Q., Sablayrolles, A., Mensch, A., et al. (2023). Mistral 7B. arXiv.
71. Bai, J., Bai, S., Chu, Y., et al. (2023). Qwen technical report. arXiv.
72. Bommasani, R., Hudson, D. A., Adeli, E., et al. (2021). On the opportunities and risks of foundation models. arXiv.
73. Chowdhery, A., Narang, S., Devlin, J., et al. (2022). PaLM: Scaling language modeling with pathways. arXiv.
74. Davuluri, P. N. AI-Augmented Sanctions Screening: Enhancing Accuracy and Latency in Real Time Compliance Systems.
75. Wei, J., Wang, X., Schuurmans, D., et al. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824–24837.
76. Lyu, L., Yu, H., & Yang, Q. (2022). Threats to federated learning: A survey. Proceedings of the IEEE, 110(5), 750–780.
77. Kairouz, P., McMahan, H. B., Avent, B., et al. (2021). Advances and open problems in federated learning. Foundations and Trends in Machine Learning, 14(1–2), 1–210.
78. Breck, E., Cai, S., Nielsen, E., Salib, M., & Sculley, D. (2021). The ML test score: A rubric for ML production readiness and technical debt reduction. IEEE International Conference on Big Data.
79. Kolla, S. H. (2024). Retrieval-Augmented Enterprise Intelligence: Enhancing Accuracy, Trust, and Operational Decision-Making. International Journal of Future Innovative Science and Technology (IJFIST), 7(2), 12425.
80. Awosika, T., Shukla, R. M., & Pranggono, B. (2023). Transparency and privacy: The role of explainable AI and federated learning in financial fraud detection. arXiv.
81. Papasavva, A., Johnson, S., Lowther, E., Lundrigan, S., Mariconti, E., Markovska, A., & Tuptuk, N. (2024). Application of AI-based models for online fraud detection and analysis. arXiv.
82. Bandi, V. D. V. K. (2023). MLOps frameworks for reliable model deployment in cloud data platforms. Journal of Artificial Intelligence and Big Data, 3(1), 84-101.

ISSN: 3067-1140

AMERICAN ONLINE JOURNAL OF SCIENCE AND ENGINEERING

VOLUME: 02 ISSUE: 03

RECEIVED: JULY 27 REVISED: AUGUST 06 ACCEPTED: AUGUST 25 PUBLISHED: SEPTEMBER 17

83. Banulescu-Radu, D., Cénac, P., Farabullini, F., & Vassallo, P. (2024). Practical guideline to efficiently detect insurance fraud in the era of machine learning: A household insurance case. Journal of Risk and Insurance, 91(4), 867–913.
84. Zhang, R., Cheng, D., Yang, J., Ouyang, Y., Wu, X., Zheng, Y., & Jiang, C. (2024). Pre-trained online contrastive learning for insurance fraud detection. Proceedings of the AAAI Conference on Artificial Intelligence, 38(20), 22511–22519.
85. Amistapuram, K. (2024). Smart Decision Support Systems For Dynamic Tax Policy Optimization Using Reinforcement Learning. Available at SSRN 6143426.
86. Mudigonda, S. S., Baruah, P. K., Kandala, P. K., Gupta, R. Y., Hegde, S. N., & Chebrolu, S. (2024). Using interpretable machine learning methods: An application to health insurance fraud detection. Society of Actuaries Research Report.
87. Fu, M., Pasuksmit, J., & Tantithamthavorn, C. (2024). AI for DevSecOps: A landscape and future opportunities. arXiv.
88. Debener, J., Heinke, V., & Kriebel, J. (2023). Detecting insurance fraud using supervised and unsupervised machine learning. Journal of Risk and Insurance.
89. Kolla, T. (2024). Graph Neural Networks for HCC Risk Adjustment and Interoperability. International Journal of Science, Research and Technology, 7(6), 13244-13255.