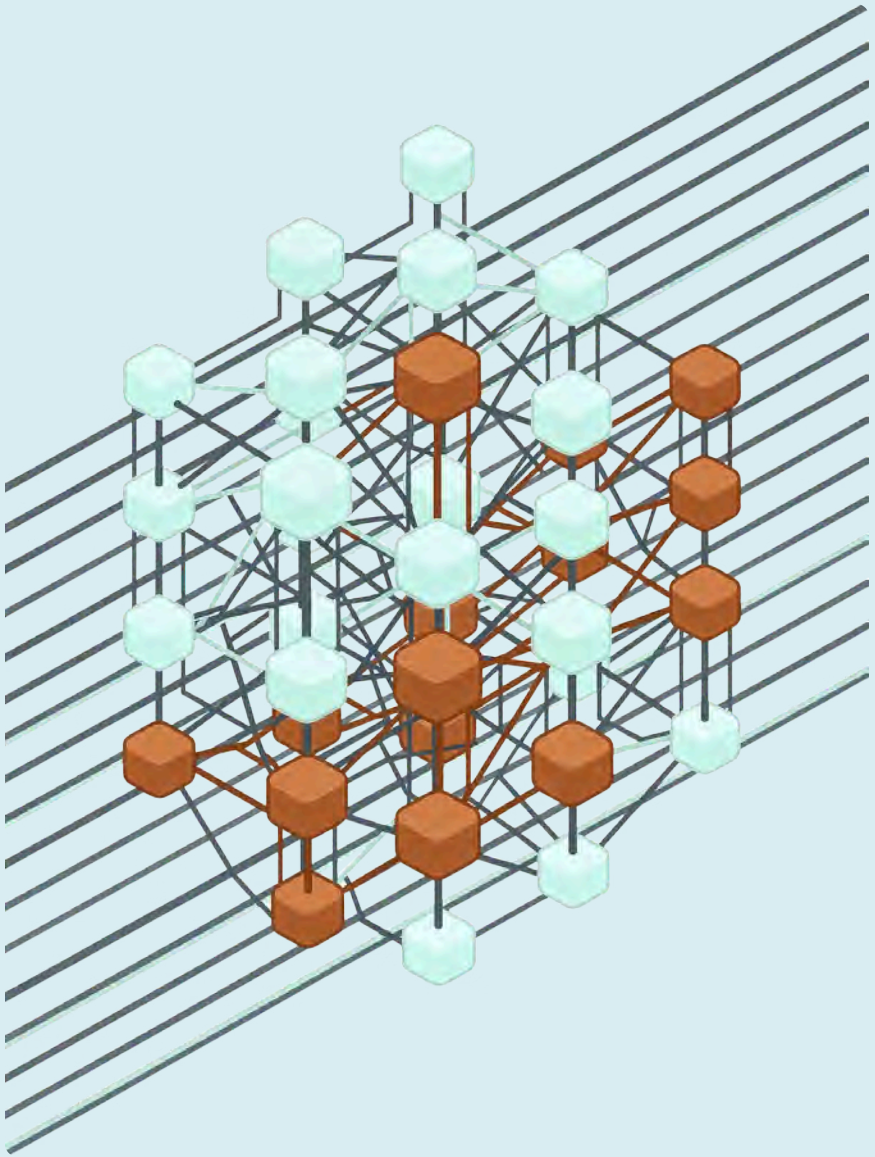


AI-Driven Fraud Detection & Personalization in Scalable Go Fintech Systems

By Sridhar Rao Muthineni
Product Tech Lead, Capgemini
Conf42 Golang 2026

The Challenge

Fintech Is Growing So Is the Threat Surface

Account Takeovers

Credential stuffing and session hijacking target high-value accounts at scale.

Synthetic Identities

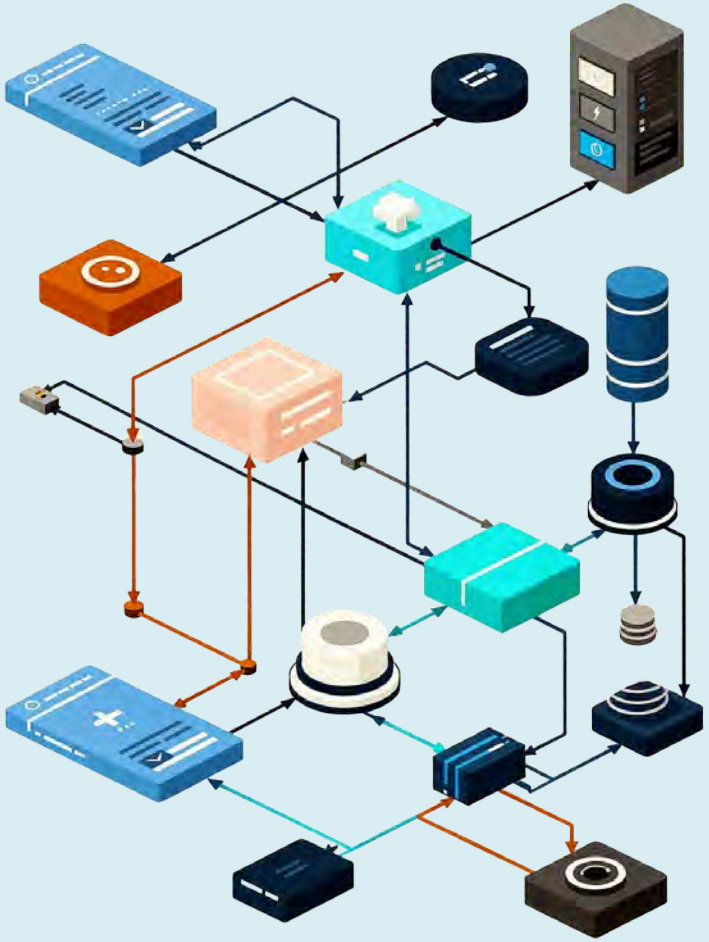
Fabricated profiles exploit onboarding gaps in KYC and credit pipelines.

Money Laundering

Layered transactions obscure illicit fund flows across distributed networks.

User Expectations

Customers demand personalized, real-time financial experiences alongside security.



A Unified AI-Driven Approach

This session presents a single, coherent framework that addresses both fraud prevention and personalization two historically siloed problems within high-performance Go-based fintech systems.

- **Detect**

  Real-time anomaly detection on high-frequency transaction streams

- **Personalize**

  Adaptive recommendations, credit scoring, and conversational agents

- **Govern**

  Explainability, fairness, and privacy-preserving deployment

Chapter 1

ML Architectures for Fraud Detection

This chapter reviews the main machine learning architectures used for real-time fraud detection in high-volume transaction streams, focusing on the tradeoffs between accuracy, latency, interpretability, cost, and resilience in production fintech systems.

- **Model families**

  GBDTs, RNNs, GNNs, and Transformers each fit different fraud signals.

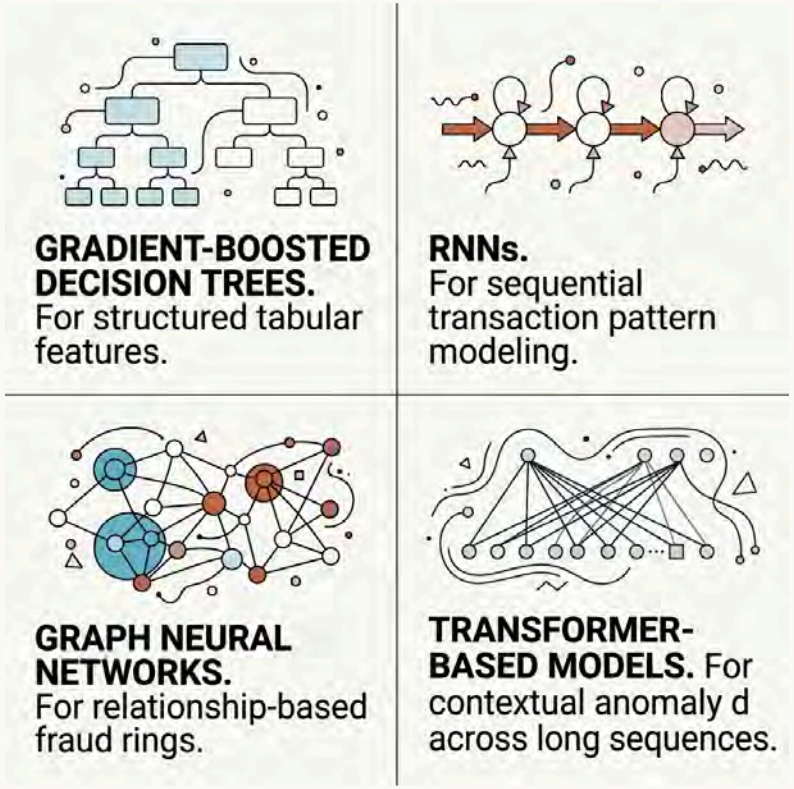
- **Streaming constraints**

  Models must infer quickly and reliably as events arrive.

- **Production tradeoffs**

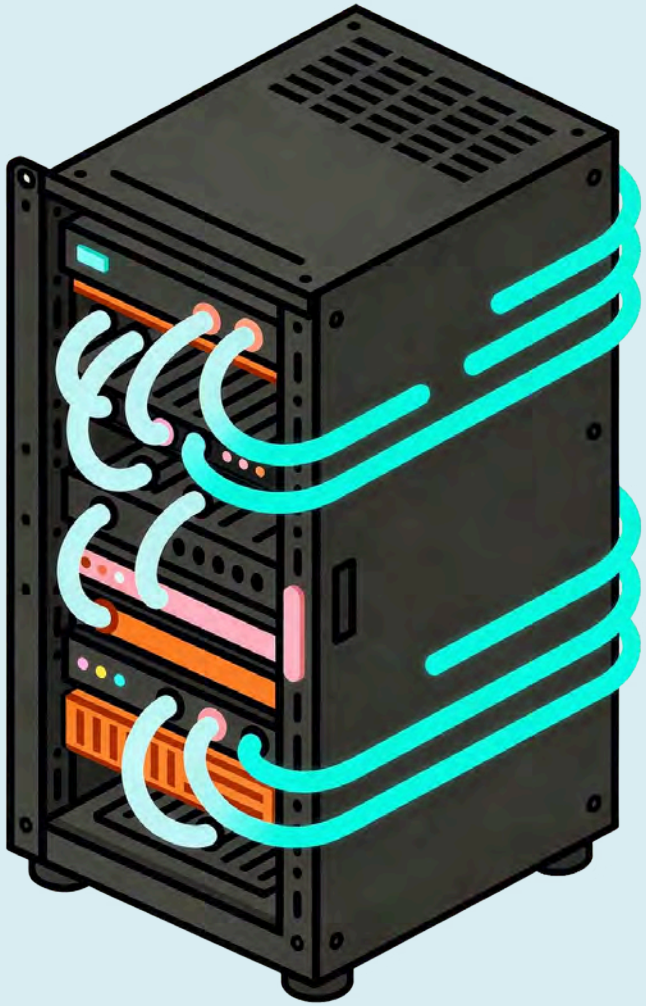
  The best choice balances accuracy, explainability, and operational cost.

Machine Learning Models for Fraud Detection



Choosing the Right Architecture

- **GBDTs** excel at structured tabular features with fast inference
- **RNNs** capture temporal patterns in sequential transaction histories
- **Graph Neural Networks** expose fraud rings hidden in account relationships
- **Transformers** contextualize anomalies across long behavioral sequences



Why Go Powers the Inference Pipeline

- **Low-Latency Execution**

  Go's compiled runtime delivers sub-millisecond decision paths critical for payment authorization flows.

- **Goroutine Concurrency**

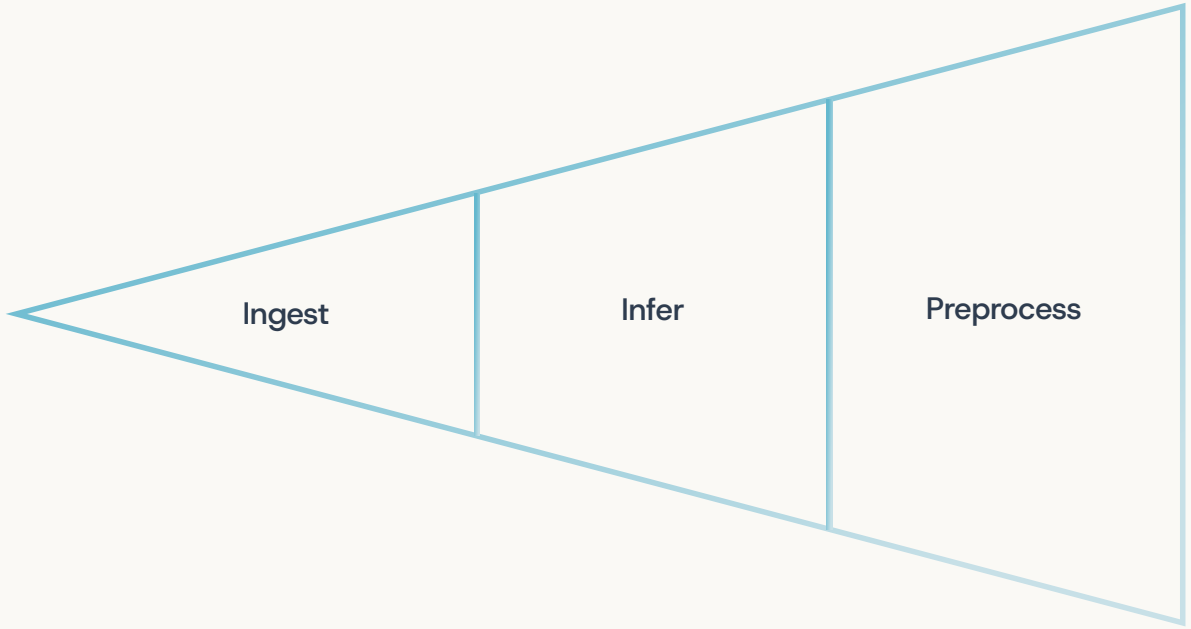
  Thousands of concurrent inference requests handled efficiently via goroutines and channels.

- **Efficient Memory Management**

  Controlled GC pauses and stack allocation minimize latency spikes under high transaction load.

Architecture Deep Dive

Streaming Inference Pipeline in Go



Go's standard library and ecosystem including Kafka clients, gRPC bindings, and protobuf make building low-latency streaming fraud pipelines idiomatic and production-ready.

WHAT WE MEASURE | THE CORE TRADE OFF

Key Production Metrics & Trade offs

Detection Accuracy

Precision and recall across fraud classes, including rare synthetic identity patterns.

False-Positive Rate

High false positives erode user trust and increase operational review costs.

Inference Latency

End-to-end p99 latency must remain within authorization SLA windows.

> Optimizing purely for recall catches more fraud but increases false positives eroding the customer experience. Production systems must balance both, continuously.

Chapter 2

Personalization at Scale

Recommendation systems, reinforcement learning, and LLM-powered agents make fintech experiences more relevant in real time, while staying aligned with trust and regulatory constraints.

- ## Recommendation Systems

  Rank the next best action based on user behavior, context, and historical patterns. In fintech, this can surface timely offers, alerts, or product suggestions that feel relevant without being intrusive.

- ## Reinforcement Learning

  Optimize decisions from feedback over time, learning which actions lead to better outcomes. It is especially useful when the right choice depends on changing user responses and long-term impact.

- ## LLM Conversational Agents

  Deliver personalized guidance naturally through chat and voice interactions. They can explain complex financial topics, answer questions, and help users take action in a more human way.

Personalization Layers in Fintech

Recommendation Systems

Collaborative and content-based filtering surfaces relevant financial products based on behavioral signals.

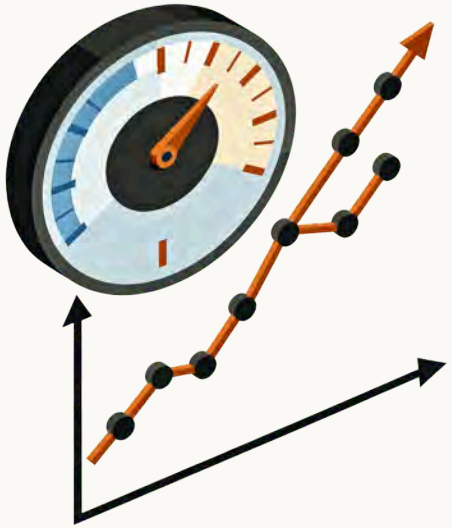
Reinforcement Learning

RL agents optimize long-term engagement by dynamically adjusting credit offers and spending nudges.

LLM Conversational Agents

Large language models power context-aware financial advisors that respond to natural-language queries.

Adaptive Credit Scoring & Behavior-Aware Insights



Beyond Static Credit Models

Traditional credit scoring relies on point-in-time snapshots. AI-driven systems model dynamic behavioral trajectories spending velocity, payment consistency, and lifestyle signals to produce continuously updated risk profiles.

- Behavior-aware scoring adjusts in near real-time
- Tailored product offerings matched to financial readiness
- Proactive insights surface before users ask

Chapter 3

Safeguards & Responsible AI

Explainability, fairness, privacy, and compliance keep AI auditable, defensible, and safe to deploy in fintech.

- ### Explainability

  Make decisions interpretable with feature attribution, reason codes, and traceability.

- ### Fairness & Bias Auditing

  Detect disparate impact with bias metrics, slice-based evaluation, and ongoing monitoring.

- ### Privacy-Preserving Techniques

  Protect sensitive data with minimization, access controls, anonymization, and privacy-aware design.

Regulatory & Ethical Constraints

Compliance Is Architecture

Regulations are not afterthoughts they shape model selection, data pipelines, and deployment decisions from the ground up.

Explainability

SHAP values and LIME provide decision justifications required by GDPR Article 22.

Fairness in Lending

Bias audits across protected attributes must be embedded in model evaluation cycles.

Privacy-Preserving Techniques in Go Deployments

Federated Learning

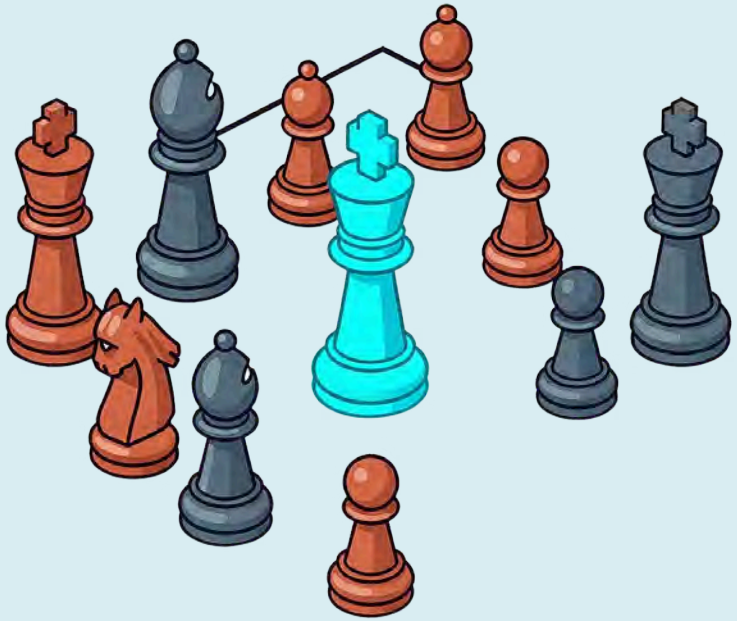
Models train across distributed devices without centralizing raw transaction data, preserving user privacy at the source.

Differential Privacy

Calibrated noise injection ensures individual records cannot be reverse-engineered from model outputs or gradients.

On-Device Inference

Lightweight Go-compiled models running on-device enable fraud scoring without transmitting sensitive data to the cloud.



Emerging Challenges & Frontier Problems

Adversarial Robustness

Fraudsters probe and adapt to model boundaries defenses must evolve continuously.

Real-Time Graph Analytics

Streaming GNNs on live transaction graphs require novel Go-native infrastructure patterns.

Responsible AI Adoption

Deploying high-stakes models in financial systems demands rigorous governance and auditability.

Key Takeaways

What to Bring Back to Your Team

1 **Go is a first-class ML infrastructure language**

Concurrency, low latency, and efficient memory make it ideal for streaming inference pipelines — not just microservices.

2 **Fraud detection and personalization share infrastructure**

Unified feature stores, streaming layers, and model serving reduce duplication and improve system coherence.

3 **Regulation is a design constraint, not a compliance checkbox**

Build explainability, fairness, and privacy into your architecture from day one — retrofitting is costly and fragile.

4 **Monitor for drift, not just errors**

Production ML systems degrade silently. Continuous behavioral monitoring is as critical as model accuracy at launch.

Thank you!