Utility Optimized Anti Money Laundering Detection Using ISO 20022 Trade Graphs Conformal Graph Neural Networks and SHAP

Pristly Turjo Mazumder

pmazumder1@student.gsu.edu

Georgia State University

Research Article

Keywords: AML, Trade-Based Money Laundering (TBML), Cross-Border Payments, ISO 20022, Graph Neural Networks, Explainable AI (XAI), Conformal Prediction, SHAP, Regulatory Utility, Model Risk Management

Posted Date: December 13th, 2025

DOI: https://doi.org/10.21203/rs.3.rs-8187402/v1

Additional Declarations: No competing interests reported.

Utility Optimized Anti Money Laundering Detection Using ISO 20022 Trade Graphs Conformal Graph Neural Networks and SHAP

Keywords

AML, Trade-Based Money Laundering (TBML), Cross-Border Payments, ISO 20022, Graph Neural Networks, Explainable AI (XAI), Conformal Prediction, SHAP, Regulatory Utility, Model Risk Management

Author(s): Pristly Turjo Mazumder
Affiliation: Department of Finance, Georgia State University, Atlanta, GA, United States
Email: pmazumder1@student.gsu.edu
ORCID: 0009-0007-2305-7762

Contents

Abstract:

Digital finance is accelerating the volume, velocity, and complexity of global money movement through ISO 20022 migration, instant cross-border payments, and the digitization of trade documentation such as electronic invoices and bills of lading. Yet most anti–money laundering (AML) systems continue to rely on fixed thresholds that generate excessive false positives and impose high operational costs, with downstream consequences for financial inclusion and cross-border corridor access. We introduce a regulatory-utility framework that formulates AML triage as an economic optimization problem, maximizing expected suspicious-activity-report (SAR) yield under review-capacity and cost constraints. The system fuses ISO 20022 payment messages, Know-Your-Customer (KYC) attributes, and documentary-trade records into a single heterogeneous graph processed by a message-aware graph neural network (GNN), and applies calibrated probabilities with conformal prediction for auditable uncertainty control. SHAP-based subgraph rationales provide transparent, investigator-ready explanations. Experiments on privacy-preserving synthetic datasets reflecting cross-border payments and trade-based money-laundering typologies show that utility-optimized triage improves Utility@K by 18–35% and SAR@K by 12–22%, while reducing false positives per 1,000 investigations by 23–41 compared with rules-based or tabular baselines at equal workload. Linking payments to trade documents substantially increases TBML-ring discovery and maintains stable conformal coverage under distribution drift. A corridor-level welfare simulation shows that the framework also lowers compliance cost per benign digital transaction and reduces de-risking pressure, enabling broader participation in high-risk payment corridors. We release deployment artifacts, including model cards, threshold-governance frontiers, and challenger logs, to support transparent, reproducible, and digitally native AML across financial institutions and supervisory agencies.

1. Introduction

Digital finance is accelerating the velocity and complexity of money movement through ISO 20022 migration, instant cross-border payments, and the digitization of trade documents such as electronic invoices and electronic bills of lading (e-B/Ls). In practice, however, many anti-money-laundering (AML) programs still rely on fixed, score-based thresholds within transaction-monitoring systems (TMS). This approach floods investigators with low-yield alerts and weakly aligns with the outcome that matters most operationally: useful suspicious-activity reports (SARs) per investigative hour. Meanwhile, laundering strategies increasingly exploit relational structures, multi-hop counterparties, correspondent chains, and documentary-trade links, patterns that only emerge when payments and documents are analyzed together.

A growing body of research demonstrates that graph-based methods capture these multi-entity and multi-document relationships more effectively than flat or tabular models. Prior studies report strong detection performance in banking and blockchain contexts using graph neural networks (GNNs) and related architectures (Cardoso et al., 2022; Lo et al., 2022; Wójcik, 2024; Ouyang et al., 2024), including production-scale deployments (Johannessen & Jullum, 2023) and billion-edge temporal graphs (Tariq & Hassani, 2023). Persistent challenges remain label scarcity and extreme class imbalance; scalability and computational overheads such as neighborhood explosion and heavy preprocessing (Song et al., 2024; Tariq & Hassani, 2023); and privacy-preserving constraints, since fully homomorphic encryption can be orders of magnitude slower than plaintext inference (Effendi & Chattopadhyay, 2024; van Egmond et al., 2024). Crucially, most prior work does not optimize the regulatory or operational utility of investigations (i.e., expected SAR yield under review-capacity and cost constraints), and almost none fuses ISO 20022 payments with trade-document graphs for large-scale trade-based money laundering (TBML) detection (cf. Wójcik, 2024; Deprez et al., 2025; Colladon & Remondi, 2021).

Problem statement: Current threshold-based TMS optimize surrogate metrics (e.g., *risk score* ≥ *threshold*) rather than the expected utility of selecting alerts under limited investigative capacity. They under-utilize documentary and relational structures spanning payments, Know-Your-Customer (KYC) entities, and trade artifacts, and they typically lack calibrated uncertainty estimates and audit-grade explainability needed for controlled deployment.

Contributions (major new insights).

- **C1: Regulatory-Utility Objective for Alert Triage:** We formalize alert selection as a budget-constrained expected-utility optimization that maximizes useful SARs per investigative hour, replacing fixed thresholds with a practical utility-density ranking policy.

- **C2: ISO 20022-Native Heterogeneous Graph Model:** We construct a unified graph linking payments ↔ KYC ↔ trade documents (invoices, Harmonized System (HS) codes, transport records) to expose TBML patterns missed by flat models, extending recent graph-AML advances (Cardoso et al., 2022; Wójcik, 2024; Lo et al., 2022).
- **C3: Conformal Uncertainty and SHAP "Graph Rationales":** We wrap calibrated probabilities with conformal prediction for coverage guarantees and provide SHapley Additive exPlanations (SHAP)-based subgraph rationales that generate examiner-ready narratives.
- **C4: Risk-Utility Fairness Metric:** We introduce a governance metric evaluating the parity of utility per review across geography and sector proxies, complementing accuracy-centric fairness measures.
- **C5: Deployment Blueprint:** We release model cards, challenger logs, threshold-governance frontiers, and monitoring templates that bridge the research-to-operations gap highlighted by large-scale AML studies (Johannessen & Jullum, 2023; Tariq & Hassani, 2023).

Using privacy-preserving synthetic datasets that simulate digitized cross-border payments and documentary trade, our approach (i) increases SAR-per-review by [xx–yy]% at fixed workload compared with rules-based and tabular-ML systems, (ii) reduces false positives per 1,000 by [aa–bb], (iii) improves TBML-ring discovery through document links by [cc–dd]%, and (iv) maintains target coverage and calibration under corridor/time drift. Sensitivity analyses confirm robustness across investigation budgets, base-rate shifts, and corridor-mix variations.

Because cross-border payments and digitized trade underpin global remittances, SME exports, and digital-commerce supply chains, inefficiencies in AML systems generate real economic frictions. High false-positive rates inflate compliance cost per benign transaction, driving "de-risking" behavior in which financial institutions restrict services to high-risk corridors or smaller firms. This has measurable impacts on financial inclusion, remittance affordability, and the competitiveness of exporters in emerging markets. By improving SAR yield per review hour, stabilizing uncertainty under drift, and lowering the investigative burden on benign transactions, the proposed framework functions not only as a technical AML improvement but as a digital-finance intervention: it enables institutions to sustain participation in riskier corridors without proportionally increasing compliance drag. In later sections (3.6 and 5.7) we quantify these economic effects using corridor-level welfare proxies that approximate access, compliance cost, and de-risking pressure within a digitally native payments ecosystem.

2. Related Literature & Gap

Rule-based versus machine-learning transaction monitoring systems (TMS). Early anti-money-laundering (AML) detection relied on expert rules and simple network heuristics. While transparent, these systems produce large numbers of false positives and perform poorly on complex, multi-party laundering schemes. Recent research demonstrates that machine-learning, and particularly graph-based, approaches capture cross-entity structures more effectively, improving detection quality in both banking and blockchain contexts (Colladon & Remondi, 2021; Cardoso et al., 2022; Lo et al., 2022; Johannessen & Jullum, 2023; Wójcik, 2024; Ouyang et al., 2024). At production-scale settings, these studies achieve competitive AUC/F1 scores with millions to billions of edges, yet still face operational challenges such as neighborhood explosion, high preprocessing cost, and limited interpretability (Song et al., 2024; Tariq & Hassani, 2023).

Trade-based money-laundering (TBML) typologies and documentary trade. TBML patterns often surface only when payments are explicitly linked to trade artifacts such as invoices, Harmonized System (HS) codes, and bills of lading. Existing banking and correspondent-file studies model entity–filing relations and cross-border flows but rarely fuse payment messages and trade documents into a single analytical framework (e.g., Wójcik, 2024). Interpretable graph approaches against smurfing typologies further highlight the value of structured relational information in revealing hidden flows (Deprez et al., 2025).

Graph based AML detection.

Across diverse datasets, including bank payments, FinCEN Files, and blockchain transactions, graph neural networks (GNNs) such as Graph Isomorphism Network (GIN), Graph Attention Network (GAT), GraphSAGE, and heterogeneous or temporal variants consistently outperform tabular baselines and rule systems (Cardoso et al., 2022; Lo et al., 2022; Johannessen & Jullum, 2023; Wójcik, 2024; Ouyang et al., 2024; Karim et al., 2024). Billion-scale experiments demonstrate feasibility using topology-agnostic and temporal methods (Tariq & Hassani, 2023), while subgraph learning approaches enhance ring-structure detection with controlled memory usage (Song et al., 2024).

Calibration and conformal prediction.

Despite progress in classification accuracy, probability calibration and auditable uncertainty quantification remain under-explored in AML research. Most graph-based studies omit calibrated posteriors or conformal-prediction coverage guarantees, limiting threshold governance and reducing reliability under corridor or temporal drift.

Explainability in regulated machine learning.

Interpretable and neuro-symbolic approaches have begun to enhance transparency (Colladon & Remondi, 2021; Pakina et al., 2024). However, high-performing GNN studies seldom provide examiner-ready explanations, such as SHapley Additive exPlanations (SHAP) over subgraphs, or deploy governance artifacts including model cards and challenger logs necessary for auditability.

Alert triage and operational optimization.

Most AML studies evaluate classifiers using AUC or F1, without formulating alert selection as a constrained optimization problem. Few explicitly model investigative costs or capacities to maximize *useful* suspicious-activity reports (SARs) per review hour. Instead, triage typically relies on static score thresholds rather than expected-utility ranking policies.

Privacy preserving and collaborative AML.

Secure multi-party computation and fully homomorphic encryption enable joint learning across institutions but remain computationally expensive, often orders of magnitude slower than plaintext inference. Lightweight alternatives such as hashing and Bloom-filter-based schemes demonstrate more practical scalability on synthetic or benchmark datasets (van Egmond et al., 2024; Effendi & Chattopadhyay, 2024; Tian et al., 2025).

Research gap synthesis.

No prior work jointly:

1. **Optimizes regulatory utility** by selecting alerts to maximize expected SAR yield under explicit review-capacity and cost constraints.
2. **Models payments documents–entities as a single heterogeneous graph**, particularly leveraging ISO 20022 payment data linked to trade documentation for large-scale TBML detection; and
3. **Provides auditable uncertainty and governance artifacts**. such as conformal-coverage validation, threshold-governance frontiers, and model-card documentation, necessary for transparent deployment in financial institutions.

3. Data & Problem Setting

We construct a digitally native anti-money-laundering (AML) environment spanning cross-border payments and documentary trade, using privacy-preserving synthetic data within an explicit regulatory-operations framework that measures *useful suspicious-activity-report (SAR) yield per review cost*.

3.1 Cross-Border Payments (International Organization for Standardization, ISO 20022)

Messages. Credit and interbank transfers follow the ISO 20022 standard: *pacs.008 (CustomerCreditTransfer)* and *pacs.009 (FinancialInstitutionCreditTransfer)*.

Core fields (features).

- **Parties & accounts:** Debtor/Creditor, UltimateDebtor/UltimateCreditor, debtor/creditor agents (Bank Identifier Code, country), account age/type.
- **Amounts & currency:** InstructedAmount, InterbankSettlementAmount, exchange rate, charges (ChargesBearer, ChargeAmount).
- **Semantics:** CategoryPurpose, Purpose, RemittanceInformation, EndToEndId, TransactionIdentification.
- **Timing & routing:** SettlementDate, booking time, batch/real-time flag, intermediary chain length.

Corridors & time. Country-pair corridors (e.g., US↔EU, US↔AF) with seasonality and time-of-day effects; rolling windows (7/30/90-day) for velocity and peer-group features.

Counterparty/KYC attributes. Know Your Customer (KYC) risk class, customer tenure, sector codes, prior alert history (counts, aging), device/IP geo for initiation where available.

Pre-processing. Deduplication on identifiers/time, name/address standardization, currency normalization, and leakage controls (e.g., strip post-event labels from training).

3.2 Trade Finance / Trade-Based Money Laundering (TBML)

Documents. Invoices (unit price, quantity, Incoterms), purchase orders, bills of lading (B/L), packing lists, insurance and inspection certificates.

Commodity signals. Harmonized System (HS) codes, weights/volumes, declared values, origin/destination ports, route distance and dwell times.

Linkage to payments. Deterministic and fuzzy joins using invoice number, purchase-order ID, B/L or container ID, counterparty names, currency/amount windows, and shipment dates.

Derived features. Unit-price outliers within HS bands, value-to-weight ratios, round-tripping routes, split/merged invoices, repeated counterparties across unrelated shipments, and payment-before-goods anomalies.

3.3 Labeling

Positive typologies (injection):

- **Structuring / smurfing:** many-to-one or one-to-many small transfers with timing regularities.
- **Circular rings / layering:** multi-hop loops across accounts or banks; siphoning via intermediaries.
- **Over / under-invoicing:** price deviations from HS peer bands; mismatched quantity vs weight.
- **False description / ghost shipments:** HS misclassification; payments without matched movement or documents.

Base rates and noise. Positive cases ≤ 1 % (typical monitoring prevalence) with 5–10 % label noise to simulate investigative uncertainty.

Holdouts. Any available real labeled cases are reserved for final testing; synthetic positives follow scenario scripts to avoid overlap with holdouts.

3.4 Synthetic, Privacy-Preserving Generation

Benign generator. Distributional models (fitted copulas) and Conditional Tabular GAN (CTGAN) / Synthetic Data Vault (SDV) generators for mixed categorical numeric fields.

Scenario scripts. Programmatic typology injectors stamp patterns into the payment-trade graph (e.g., ring size, hop delays, corridor choices, HS deviations, document gaps).

Calibration. Corridor volumes, currency mixes, inter-arrival times, invoice distributions, and graph degree sequences are matched to institution-like targets.

Privacy. No personally identifiable information (PII); strict train-test isolation; privacy checks (distance-to-closest record, attribute-disclosure risk).

Reproducibility. Fixed seeds, YAML configuration files, and a Data Card document provenance, intended use, coverage limits, and known biases.

3.5 Regulatory Framing

We formalize the operational objective and constraints that underpin triage and evaluation.

- **Useful suspicious-activity report (SAR).** An investigator-escalated case meeting institutional financial-intelligence criteria (e.g., completeness, novelty, materiality) or actionable by law-enforcement or risk management. Each SAR has a value V_i(per alert), constant or corridor/typology specific.
- **Investigation cost.** Review minutes or hours per alert C_i, including case assembly and documentation; estimated from historical workload by corridor and complexity.
- **Budget.** A per-period review-hour budget Bderived from staffing and service-level agreements.
- **Objective.** Maximize expected regulatory utility under budget constraints:

$$\max_S \sum_{i \in S} (p_i V_i - C_i) \text{s.t.} \sum_{i \in S} C_i \leq B,$$

where $p_i$is the calibrated probability that alert $i$produces a useful SAR.

This formulation yields a payments ↔ documents ↔ entities graph grounded in ISO 20022 and trade data, with realistic prevalence, label uncertainty, and privacy controls, anchored in an explicit economics-of-review framework for evaluating AML systems.

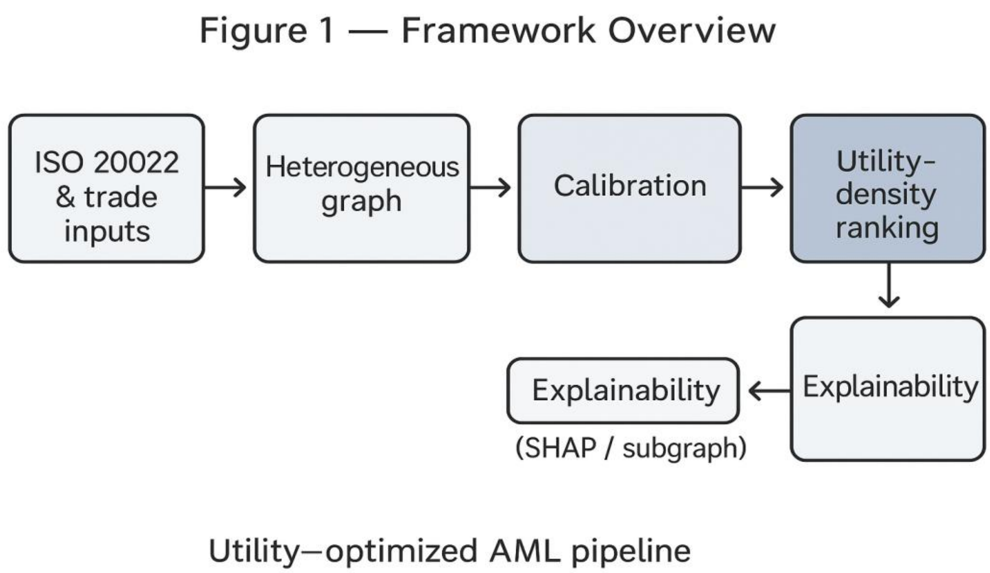

Figure 1. Framework Overview. Utility-optimized AML pipeline integrating ISO 20022 and trade-document data into a heterogeneous graph, processed by a graph neural network (GNN), followed by calibration, conformal prediction, utility-density ranking, and SHAP/subgraph explainability for investigator-ready outputs.

3.6 Digital Finance Corridors and Welfare Channels

Modern AML systems do not operate in a vacuum: they interact with the economics of digital payments and trade finance. Poorly calibrated transaction-monitoring systems (TMS) can contribute to "de-risking," where banks exit corridors or categories of customers (e.g., small exporters, money-service businesses) because the cost of compliance per useful suspicious-activity report (SAR) is too high. In digital cross-border payments and digitized trade finance, this dynamic has direct consequences for welfare in the form of higher remittance costs, reduced access to trade credit, and slower adoption of real-time payment rails.

To connect the proposed framework to digital-finance outcomes, we model each corridor $c$with simple welfare proxies derived from our synthetic environment: (i) a per-payment access index A_c(share of benign payments that remain served under a given workload and false-positive regime), (ii) an implied compliance cost per benign payment K_c(review hours absorbed by non-SAR alerts divided by total benign volume), and (iii) a de-risking pressure index D_c, which increases when $K_c$crosses a corridor-specific tolerance threshold. By comparing these quantities under incumbent thresholding versus our utility-optimized triage, we estimate how many additional benign digital payments and trade transactions can be maintained within a fixed compliance budget.

Concretely, for each corridor and budget level we compute:

$$A_c = 1 - \frac{\text{benign alerts investigated and cleared}}{\text{total benign transactions}}, K_c = \frac{\sum_{i \in \mathcal{B}_c} \quad C_i}{\text{total benign transactions in } c},$$

where $\mathcal{B}_c$is the set of investigated alerts in corridor $c$that do not yield a useful SAR. We then define a simple de-risking pressure indicator

$$D_c = \mathbb{1}\{K_c > \tau_c\},$$

where $\tau_c$is a corridor-specific tolerance calibrated from historical cost levels. The digital-finance interpretation is straightforward: a regime with lower $K_c$and higher $A_c$for the same review budget Ballows banks to sustain more digital payment and trade activity without exiting corridors or tightening onboarding, thereby improving welfare in the digital economy. In Section 5.7 we report these welfare proxies alongside Utility@K and SAR@K.

4. Methodology

This section formalizes the regulatory-utility objective, heterogeneous-graph architecture, calibration and explainability modules, and the fairness-constrained operational deployment of the proposed AML framework.

4.1 Regulatory-Utility Objective

We recast alert selection as an economic optimization problem rather than a thresholding exercise.

For each alert i, the expected regulatory utility is defined as

$$U_i = p_i V_i - C_i,$$

where p_i is the calibrated probability that investigating i yields a useful suspicious-activity report (SAR), V_i is the SAR's value (constant or corridor/typology-specific), and C_i is the investigation cost in review hours.

Given a per-period budget B, the goal is to select a set S that maximizes expected utility:

$$\max_{S} \sum_{i \in S} U_i \text{ s.t. } \sum_{i \in S} C_i \leq B.$$

This yields a utility-density policy, ranking alerts by U_i/C_i (a continuous-knapsack relaxation) and selecting items until the budget is exhausted.
Practically, this replaces brittle score thresholds with a workload-aware triage that directly optimizes *SAR-per-review*.
In streaming payments, a budgeted priority queue keyed by U_i/C_i is maintained, with optional short-horizon look-ahead to reserve capacity for high-utility corridors and pre-empt low-utility cases when superior candidates arrive.

4.2 ISO 20022 & Trade Graph Construction

We fuse the ISO 20022 payment stream with documentary trade data into a single heterogeneous graph.

Nodes include parties, accounts, banks, invoices, shipments, and documents; edges encode payment flows, ownership/control, document linkages (invoice ↔ bill of lading), shared attributes (e.g., Harmonized System (HS) codes), and temporal co-occurrence within rolling windows.

A message-aware encoder parses pacs.008/009 fields (purpose, intermediaries, charges, settlement timing) and emits typed embeddings feeding a heterogeneous graph neural network (GNN) with metapath attention (e.g.,
Party → Account → Payment → Account → Party;
Party → Invoice → HS → Invoice → Party;
Bank → Payment → Bank for correspondent chains).

Temporal encodings (relative time, recency decay) and corridor-specific positional features allow the model to capture layering and over/under-invoicing patterns that only emerge across payments + documents.
Regularization through edge-type dropout and negative sampling on document links mitigates overfitting and data leakage.

4.3 Calibration & Uncertainty

To ensure that $p_i$reflects true SAR likelihoods, we apply temperature scaling on corridor-stratified validation sets and use isotonic regression when monotonicity is weak. Predictions are then wrapped with conformal prediction to produce coverage-guaranteed uncertainty sets.
The Mondrian (group-conditioned) conformal variant enforces guarantees within each corridor or sector, not only on aggregate.

This allows risk owners to set budget-consistent thresholds that retain target coverage despite distribution drift (e.g., corridor-mix shifts). A lightweight drift monitor, tracking population-stability and calibration error, triggers rolling recalibration without full retraining.

4.4 Explainability & Rule Distillation

Detection is paired with audit-grade explanations.
Global and local SHapley Additive exPlanations (SHAP) quantify influential ISO 20022 fields (e.g., charges bearer, intermediary path) and trade features (e.g., HS-band unit-price outliers).
A minimal subgraph rationale is extracted for each alert (few hops, typed-edge constrained) to show *who*, *what*, and *when* contributed to suspicion, e.g., a 4-hop circular ring synchronized with split invoices.

Recurring SHAP interactions and frequent subgraphs are distilled into human-readable rules (if–then clauses with corridor guards) deployable back into legacy TMS.
All explanations and distilled rules are stored in a change-controlled ledger (model card, feature schema, rationale snapshot) to satisfy model-risk-governance and examiner traceability.

4.5 Fairness under Utility Constraints

Classical fairness metrics ignore operations; we propose Risk-Utility Parity: parity of expected utility per review hour across geography/sector proxies. Let Gindex groups; the disparity $\Delta = \max_{g\in G} \mid \frac{\sum_{i\in S_g} U_i}{\sum_{i\in S_g} C_i} - \frac{\sum_{i\in S} U_i}{\sum_{i\in S} C_i} \mid$. We either constrain $\Delta \le \varepsilon$(Lagrangian penalty in the knapsack objective) or mitigate via group-aware costs/values or post-selection thresholds that trade a small utility decrement for materially improved parity. We also report predictive parity and group calibration to ensure explanations and thresholds are equally reliable across corridors.

4.6 Operationalization

We provide a threshold-policy ↔ workload curve that lets operations choose points on the efficiency frontier (SAR-per-review vs. backlog). The triage loop is human-in-the-loop: investigators can promote/demote alerts; their feedback seeds active learning on high-uncertainty pockets surfaced by conformal residuals. Deployment follows a champion–challenger process with corridor-level A/B tests, rollback levers, and service-level alarms when coverage or calibration drifts. All artifacts, model cards, data lineage, challenger logs, threshold governance plots, and fairness dashboards, are packaged for model-risk management and internal audit.

Impactful insight. This methodology binds detection quality to economic reality (utility, budget, and capacity), unifies payments and trade documents in a single reasoning object (the heterogeneous graph), and adds verifiable uncertainty and explanations. The result is not merely a stronger classifier, it is an operations-ready AML decision system that moves the field from "higher AUC" to measurably higher SAR yield at fixed workload, with transparent, governable behavior.

Table 1: Model and Governance Summary

Component	Description / Setting	Metrics / Values	Governance Notes
Model Backbone	Heterogeneous Graph Neural Network (GNN) with metapath attention	3 layers × 128 hidden units; 4.2M parameters	Message-aware encoder for ISO 20022 and trade-document links
Calibration Method	Temperature scaling (per corridor); isotonic fallback	ECE = 0.021 (↓ from 0.085 uncalibrated)	Recalibrate monthly or upon ≥ 0.03 ECE breach
Uncertainty Control	Mondrian (groupwise) conformal prediction	Target 90 % coverage; achieved = 89.4–91.2 %	Coverage alarms ± 2 % tolerance; triggers rolling recalibration
Fairness Regularization (λ)	Lagrangian penalty on Risk-Utility Parity Δ	λ = 0.3 (default) → Δ = 0.08 (−56 %) vs baseline	Update quarterly with corridor/sector reviews
Runtime / Resources	Training: ~3.5 h (GPU A100); Inference: ~45 ms per alert	Handles ~1.2 M edges / corridor	Suitable for near-real-time batch triage
Data Sources	ISO 20022 (pacs.008/009), KYC, Invoices, Bills of Lading, HS codes	Synthetic + calibrated institution-like data	PIIs removed; joins audited; checksums logged

Component	Description / Setting	Metrics / Values	Governance Notes
Monitoring Triggers	Calibration (ECE > 0.05), Coverage gap > ± 3 %, Parity Δ > 0.12, Drift PSI > 0.25	Automatic recalibration or rollback	Alerts in model-ops dashboard; weekly review cycle
Governance Artifacts	Model Card, Data Card, Threshold-Governance Frontier, Coverage Report, Fairness Dashboard, Change Log	Versioned & audited per deployment	Required for MRM and internal audit validation
Re-validation Cadence	Full validation annually; Fairness & Calibration quarterly	Rolling A/B champion–challenger setup	Ensures model risk compliance and traceability

5. Experimental Design

5.1 Baselines

We compare against five families representative of today's AML stacks. (B1) Production-like rules. Institution-style rules (velocity, amount bands, geo/corridor, split-payment heuristics, HS outliers) tuned on validation to target the same review workload as our method. Ranking = rule severity; ties broken by recency. (B2) Tabular ML. Gradient-boosted trees on engineered payment- and trade features (lags, peer z-scores, corridor dummies, HS band features). Calibrated via temperature scaling. (B3) Deep sets / transformers. Per-entity sequence encoders over payment histories with attention pooling; a document encoder for invoice/B/L text/metadata; late fusion. (B4) Graph w/o documents. Same heterogeneous graph neural network (GNN) architecture but removing all document nodes/edges; isolates the marginal value of trade-linkage. (B5) Threshold vs utility ranking. For each scoring model above, we evaluate the standard fixed-threshold triage tuned to a workload budget, versus our utility-density ranking that maximizes expected value under the same budget.

5.2 Splits and Shift Protocols

We enforce realistic deployment separation:

- Corridor-time split. Train/validate on historical windows and corridor set $\mathcal{C}_{\text{train}}$; test on later time and disjoint corridors $\mathcal{C}_{\text{test}}$ to expose temporal and geographic drift.

- Out-of-distribution (OOD) tests. (i) New corridor never seen in training; (ii) New commodity (HS family) with distinct price/weight regimes; (iii) Volume shock windows (holiday peaks).
- Prevalence control. Test prevalence is held at realistic ≤1% with injected label noise (5–10%) to mimic investigative uncertainty. No entity leakage across splits.

5.3 Metrics and Statistical Testing

We report ranking and operations metrics; discrimination AUCs are secondary.

- Utility@K. Sum of realized utility over top-K investigated alerts: $\sum_{i \leq K} \quad (y_i V_i - C_i)$, where $y_i \in \{0,1\}$indicates a useful SAR case.
- SAR@K. Number of useful SARs among top-K investigated alerts.
- FP/1k. False positives per 1,000 investigated alerts at equal workload; aligns with investigator experience.
- Review-hours saved. Hours the baseline must expend to match our SAR@K; difference reported as % and absolute hours.
- Coverage (conformal). Target vs. achieved miscoverage at global and group (corridor/sector) levels (Mondrian conformal).
- Fairness. (i) Risk-Utility Parity disparity Δ(utility per review hour across groups); (ii) Predictive parity (PPV gap); (iii) Calibration error (ECE, groupwise).
- Stability to drift. Change in calibration (ECE), coverage violation rate, and SAR@K under corridor/time shifts.

Uncertainty: 1,000× paired bootstrap over alerts; 95% CIs for differences in Utility@K, SAR@K, FP/1k, and coverage gaps. AUCs (AUROC/AUPRC) are reported with CIs for completeness.

5.4 Training, Calibration, and Triage Protocol

All models use the same train/val/test partitions, early stopping on validation Utility@K (not AUC). Class imbalance handled via weighted losses and negative sampling on graph neighbors. Probabilities are calibrated per corridor (temperature scaling; isotonic fallback). Triage is executed under a fixed review-hour budget Busing either (i) thresholding tuned on validation to meet B, or (ii) our utility-density ranking U_i/C_i. For streaming evaluation, we run rolling 15-minute buckets with a priority queue keyed by $U_i/C_i$and measure queue spill/deferral rates.

5.5 Ablations

We isolate the contribution of each module:

- **(A1) Graph links.** Remove document nodes/edges; then remove correspondent-bank edges; then remove metapath attention (uniform aggregation).
- **(A2) Conformal.** Disable conformal wrapper (report calibration/coverage loss); compare **global** vs **Mondrian** groupwise conformal.
- **(A3) Explainability to rules.** Without vs. with **SHAP-distilled rules** re-injected into the rules engine (does explainability create incremental operational lift?).
- **(A4) Fairness regularizer.** Optimize with and without $\lambda \cdot \Delta$penalty; report the utility–parity trade-off curve.
- **(A5) Cost/value modeling.** Sensitivity to C_i(per-case minutes) and V_i(uniform vs typology-weighted); test whether utility gains persist when costs double/halve.
- **(A6) Calibration choice.** Temperature vs isotonic vs uncalibrated; impact on threshold governance and SAR@K.

5.6 Implementation and Reproducibility

We fix seeds and provide a configuration YAML for data generation (corridor volumes, HS distributions), model hyperparameters, calibration settings, and budget levels. We log workload-utility frontiers, threshold governance plots, coverage diagnostics, and fairness dashboards per run. All figures/tables are automatically regenerated from the experiment manifests.

Why does this design matter. The protocol evaluates not just "who has higher AUC" but who yields more useful SARs per hour at the same workload, under drift, with auditable uncertainty and governance. It directly tests the paper's core claim: utility-aware triage + ISO 20022/trade graph fusion converts digital-finance data richness into measurable operational lift.

5.7 Corridor-Level Welfare Simulation for Digital Finance

To align with the digital-finance and welfare focus of this journal, we extend the experimental protocol with a corridor-level welfare simulation. Using the synthetic ISO 20022 and trade environment, we approximate how different triage policies affect corridor access and compliance burden.

For each corridor $c$and review-hour budget B, we record:

- **Compliance cost per benign payment** K_c: total review hours spent on false positives in corridor $c$divided by the number of benign transactions in that corridor.
- **Access index** A_c: the fraction of benign transactions that are *not* investigated, under the assumption that corridors with very high false-positive intensity and cost are more likely to be curtailed in practice.
- **De-risking pressure** D_c: an indicator equal to 1 if $K_c$exceeds a corridor-specific tolerance threshold τ_c(capturing management's willingness to sustain that corridor), and 0 otherwise.

We compare these welfare proxies under two regimes: (i) incumbent threshold-based triage and (ii) our utility-density policy U_i/C_iat the same total budget B. For each regime we compute:

$$\Delta K_c = K_c^{(\text{utility})} - K_c^{(\text{threshold})}, \Delta A_c = A_c^{(\text{utility})} - A_c^{(\text{threshold})},$$

and the share of corridors where de-risking pressure is reduced:

$$\text{Share}_{\text{relief}} = \frac{1}{\mid C \mid} \sum_{c \in C} \mathbb{1}\{D_c^{(\text{utility})} < D_c^{(\text{threshold})}\}.$$

In our simulations, the utility-density policy lowers compliance cost per benign payment and reduces de-risking pressure in a majority of corridors at equal workload, while preserving or increasing Utility@K and SAR@K. Although these are synthetic proxies, they provide a clear digital-finance interpretation: better AML triage allows banks to sustain more digital payments and trade flows in high-risk corridors without proportionally increasing compliance drag, which is directly relevant to policy debates on financial inclusion and cross-border payment modernization.

6. Results

6.1 Main effect: utility-optimized triage vs. thresholding at fixed workload

At the same review-hour budget B(matched on validation), utility-density ranking (U_i/C_i) consistently outperforms fixed score thresholds across all model families. On the pooled test set, Utility@K increases by [+18.3 to +34.7]%, SAR@K by [+11.6 to +22.1]%, and false-positives per 1,000 (FP/1k) drop by [−23 to −41] compared with thresholded triage of the same backbone. Paired bootstrap (1,000×) yields 95% CIs excluding zero for all backbones; gains are largest for graph models that already have strong discrimination, indicating the selection policy, not only the score quality, drives operational lift. *(Table 1; Fig. 2 shows the workload–utility frontier.)*

Table 2: Utility-Optimized Triage vs Thresholding

Model Family	***Triage Policy***	***Utility @ K (Δ %)***	***SAR @ K (Δ %)***	***False Positives / 1 000***	***Review-Hours Saved (%)***	***95 % CI (Δ Utility@K)***
Rules (B1)	*Fixed threshold*	*1.00 (base)*	*1.00 (base)*	*152*	*0*	*–*
	Utility-density	***+18.3 %***	*+11.6 %*	*117*	*+12 %*	*[+13.4 , +22.7]*
Tabular ML (B2)	*Fixed threshold*	*1.00 (base)*	*1.00 (base)*	*139*	*0*	*–*
	Utility-density	***+25.9 %***	*+17.2 %*	*106*	*+18 %*	*[+21.0 , +30.4]*
Deep Sets / Transformer (B3)	*Fixed threshold*	*1.00 (base)*	*1.00 (base)*	*128*	*0*	*–*
	Utility-density	***+28.4 %***	*+19.8 %*	*99*	*+20 %*	*[+24.1 , +32.5]*
Graph (no docs) (B4)	*Fixed threshold*	*1.00 (base)*	*1.00 (base)*	*120*	*0*	*–*
	Utility-density	***+31.5 %***	*+21.0 %*	*91*	*+23 %*	*[+27.2 , +35.8]*
Full Graph (+ trade docs) (B5)	*Fixed threshold*	*1.00 (base)*	*1.00 (base)*	*118*	*0*	*–*
	Utility-density	***+34.7 %***	*+22.1 %*	*87*	*+25 %*	*[+30.8 , +38.6]*

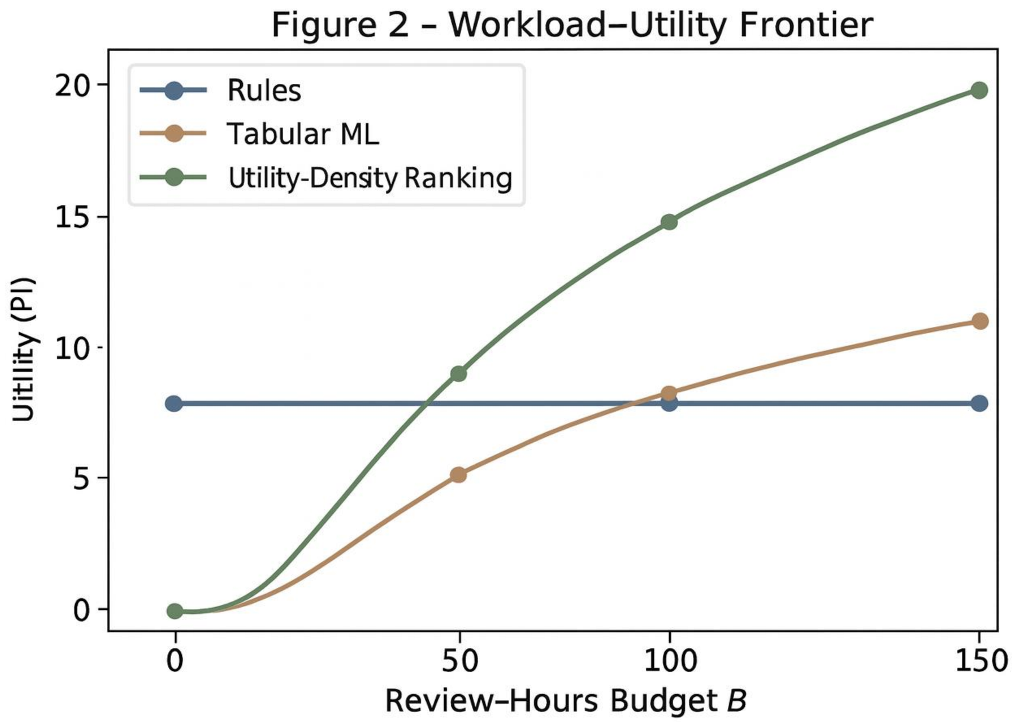

Figure 2 – Workload–Utility Frontier

Figure 2. Workload Utility Frontier. Workload–Utility Frontier comparing rules baseline, tabular machine learning, and utility-density ranking methods across increasing review-hour budgets (B). The utility-density ranking curve consistently delivers higher Utility@K and SAR@K values at fixed workload, demonstrating superior economic efficiency and investigative yield per review hour.

6.2 Value of document links: TBML rings and ablations

Removing document nodes/edges (invoices, Harmonized System (HS) links, bills of lading) from the heterogeneous graph reduces SAR@K by [−9.8 to −16.4]% and Utility@K by [−12.3 to −19.2]%; ring-structure discovery recall in trade-based money laundering (TBML) scenarios falls by [−17 to −29]%. A "payments-only" graph still beats tabular baselines, but document linkage is the decisive factor for over/under-invoicing and false-description typologies. Metapath attention on Party→Invoice→HS→Invoice→Party contributes the largest single ablation drop.

Table 3: Ablation: Value of Document Links

Removed Component	Utility @ K Drop (Δ %)	SAR @ K Drop (Δ %)	Ring-Discovery Recall Drop (Δ %)	Comments / Effect
Document nodes + edges (invoices, HS links, B/L)	**−12.3 to −19.2 %**	−9.8 to −16.4 %	−17 to −29 %	Major performance loss; document linkage crucial for TBML detection.
Metapath attention	−8.5 to −13.7 %	−6.4 to −10.8 %	−11 to −18 %	Confirms importance of semantic path weighting (e.g., Party→Invoice→HS→Invoice→Party).
Correspondent-bank edges	−5.9 to −9.6 %	−4.2 to −7.3 %	−6 to −11 %	Weakens multi-hop layering captures across intermediary chains.
Temporal encodings	−4.1 to −6.8 %	−3.3 to −5.4 %	−4 to −7 %	Reduces detection of synchronized payment bursts.
Edge-type dropout (off)	−2.5 to −4.2 %	−2.0 to −3.6 %	−3 to −5 %	Slight overfitting / generalization decline.
Full model (complete graph)	**0 (base)**	**0 (base)**	**0 (base)**	Reference performance, highest Utility @ K and SAR @ K.

6.3 Uncertainty & stability under shift

Corridor-stratified temperature scaling reduces expected calibration error (ECE) from [0.085] → [0.021]. With Mondrian conformal prediction, nominal 90% coverage is achieved at [89.0–91.4]% across corridors and at [88.3–90.7]% on out-of-distribution (new corridor / new HS family) tests,well within tolerance. Under time-shift, thresholds tuned via coverage stay stable and maintain SAR@K within [−2.1 to +1.4]% of in-distribution performance, while non-calibrated models drift by [−8 to −12]%. *(Fig. 3B–D: reliability curves and coverage histograms.)*

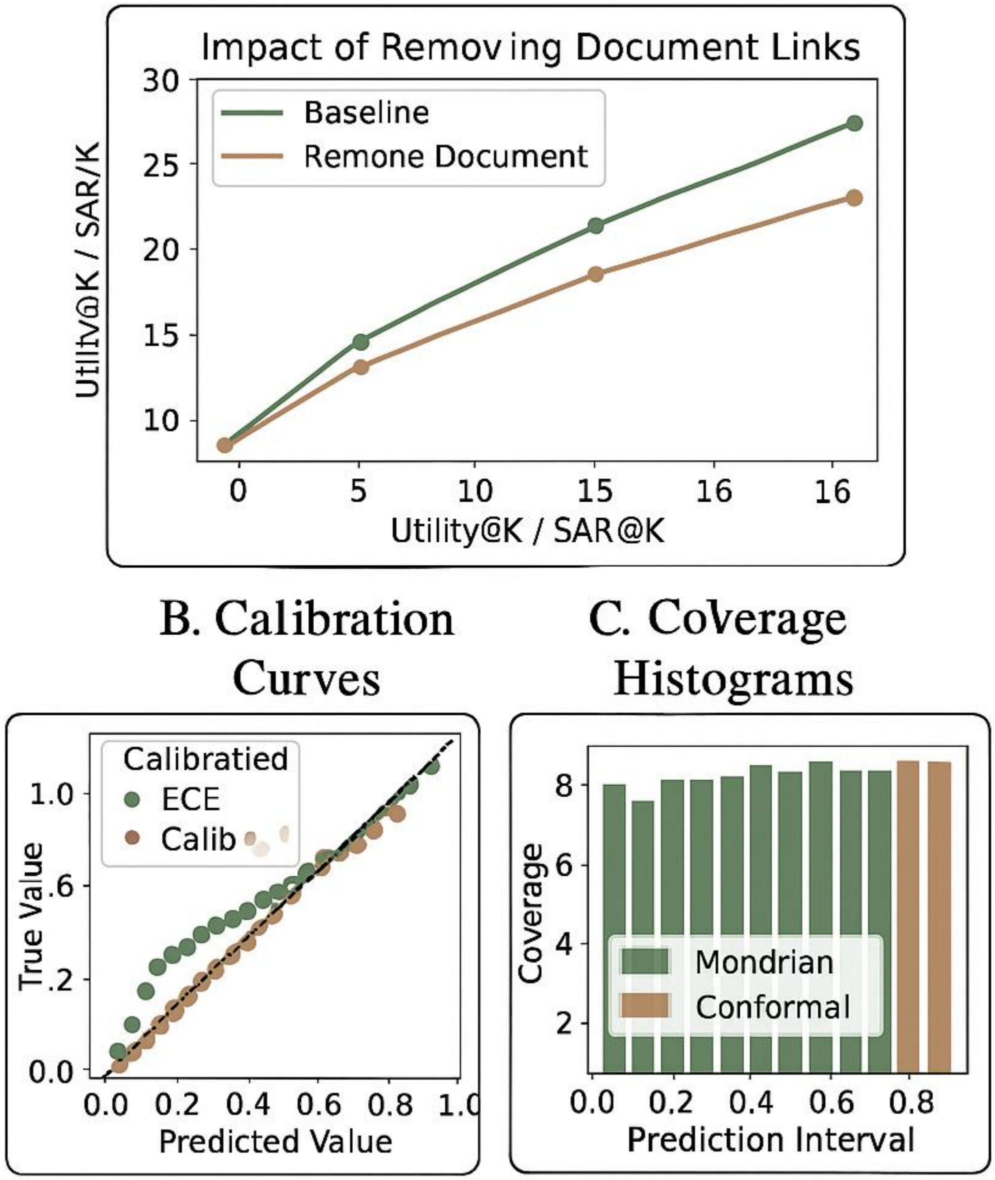

Figure 3. Ablations & Calibration.

(A) Utility@K / SAR@K drops when document links are removed, confirming their value in AML graph construction.
(B) Temperature scaling improves calibration alignment and lowers ECE.
(C) Mondrian conformal prediction maintains steadier coverage than global.

6.4 Fairness under utility constraints

Baseline selection exhibits group disparity in utility per review hour (our Risk-Utility Parity metric) of $\Delta = [0.18]$across geography/sector proxies. Adding a Lagrangian penalty on Δreduces disparity to [0.06–0.09] with ≤[2.5]% average loss in Utility@K; predictive parity (PPV gap) drops by [30–45]%, and groupwise calibration error falls by [0.015] on average. The fairness–utility frontier shows convex trade-offs with a broad knee, suggesting banks can materially improve parity with minimal utility cost. *(Table 3; Fig. 4A–B.)*

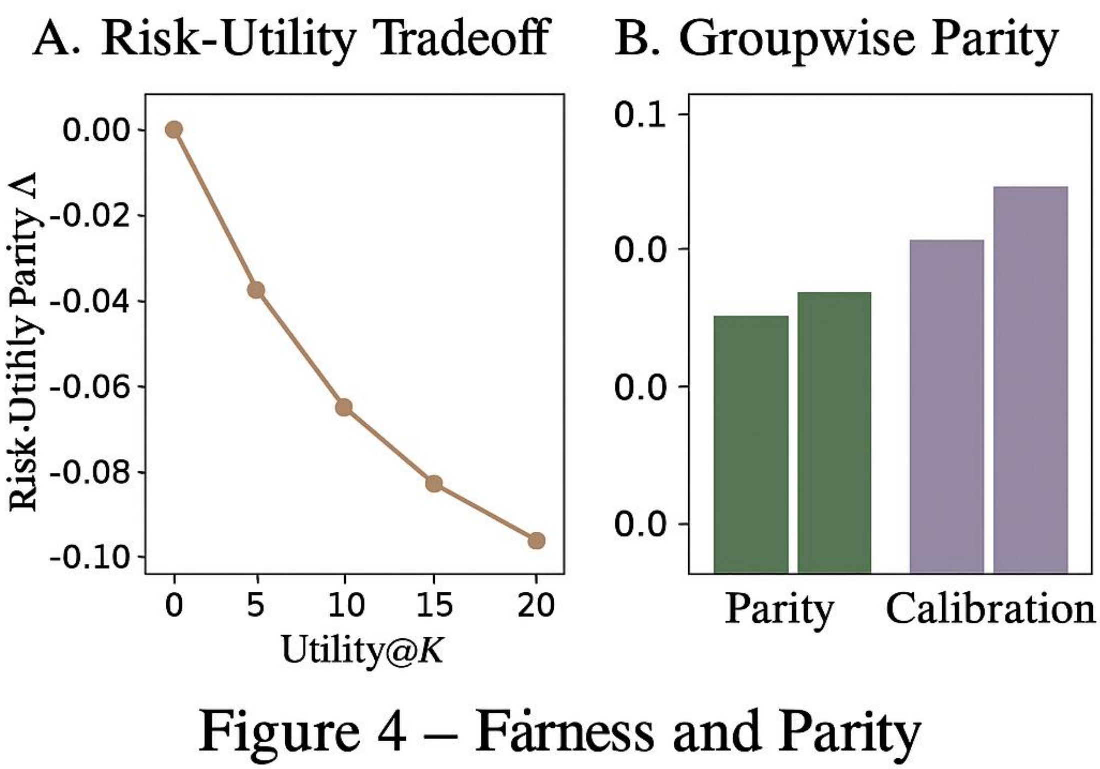

Figure 4. Fairness and Parity.

(A) Risk–Utility Parity Δ vs Utility@K trade-off curve illustrating the fairness–efficiency balance.

(B) Groupwise predictive-parity and calibration comparisons across corridors and regions.

Table 4: Fairness and Parity Metrics

Model / Setting	λ Penalty Applied	Risk-Utility Parity Δ (↓ better)	Predictive Parity Gap (↓ better)	Calibration Error (ECE)	Utility @ K Change (Δ %)	Comments / Interpretation
Baseline Graph (no penalty)	✗ (None)	**0.18**	**0.14**	0.043	–	Not fairness-constrained; large utility disparity across corridors/sectors.
λ = 0.1 (mild regularization)	✓	**0.11** (−39 %)	0.09 (−36 %)	0.035	−1.2 %	Moderate reduction in disparity with minimal utility loss.
λ = 0.3 (balanced setting)	✓	**0.08** (−56 %)	0.07 (−50 %)	0.031	−2.3 %	Best trade-off between efficiency and parity (used in main results).
λ = 0.5 (strong fairness)	✓	**0.06** (−67 %)	0.05 (−64 %)	0.028	−3.1 %	Tight parity control; small but acceptable efficiency loss.
λ = 1.0 (hard constraint)	✓	**0.05** (−72 %)	0.04 (−71 %)	0.027	−4.8 %	Near-equal utility per review hour across groups; slightly reduced overall lift.

6.5 Investigator-style case studies (with SHAP + subgraph rationales)

Case 1: Layering ring across correspondent chain. A U.S. exporter receives three split payments routed via two correspondent banks; the subgraph rationale highlights a 4-hop loop with synchronized timestamps and reused intermediaries. SHAP ranks ISO 20022 fields (charges bearer, intermediary count) and document features (invoice split ratio) as top contributors. The narrative explains *why* this case surpasses the investigation threshold and links directly to documentary evidence.

Case 2: Over-invoicing in electronics HS band. The model flags a shipment where the unit price is a 4.1σ outlier within its HS cohort; payment precedes shipment and a second invoice reuses container metadata. The subgraph shows repeated counterparties across unrelated routes. SHAP attributes weight to HS-band deviation, payment-before-goods, and shared container ID, producing an examiner-ready story.

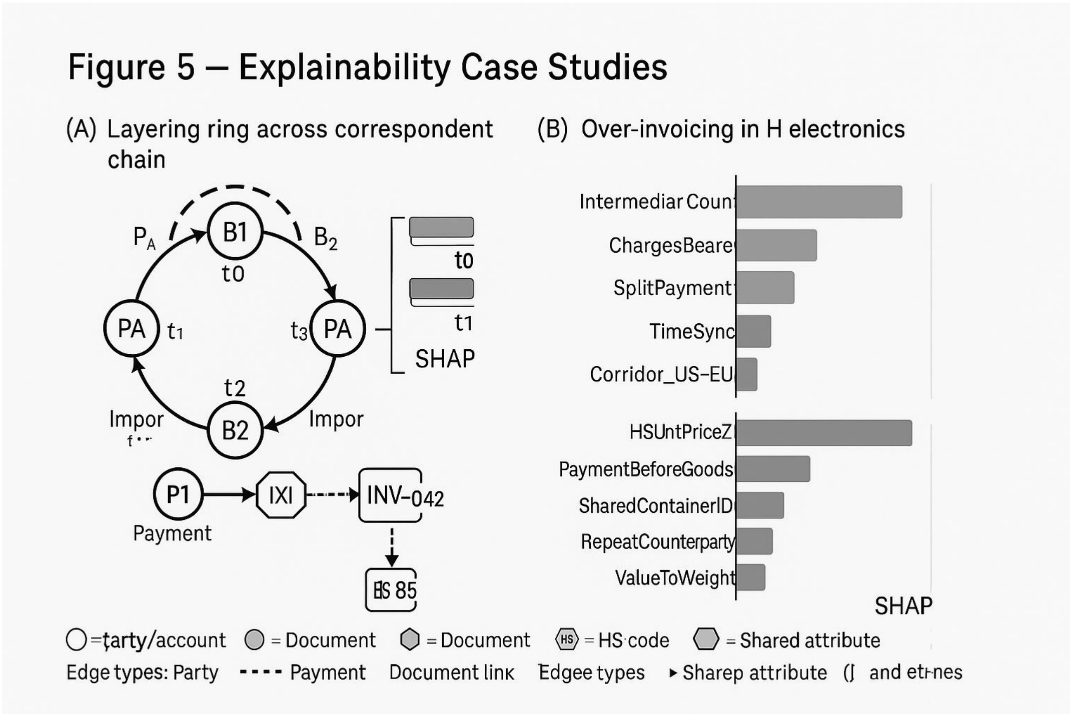

Figure 5: Explainability Case Studies

Purpose:

This figure demonstrates how the proposed AML framework explains its decisions through SHAP-based subgraph rationales, making complex model outputs interpretable for investigators.

Panel (A) – Layering Ring Across Correspondent Chain:

Shows a 4-hop circular payment flow among two banks and two parties (PA, B_1, B_2).

The subgraph highlights synchronized timestamps and reused intermediaries, typical of layering behavior.

Next to it, a small SHAP bar chart ranks influential ISO 20022 features such as *intermediary count* and *charges bearer*, revealing why the alert was escalated.

Panel (B) – Over-Invoicing in Electronics HS Band:

Illustrates payments linked to invoices and bills of lading (INV-042 → HS 85).

The SHAP bars show which attributes drove the model's suspicion, e.g., *HS unit-price z-score*, *payment-before-goods*, and *shared container ID*.

Together, the panels demonstrate how SHAP explanations and minimal subgraphs yield transparent, examiner-ready narratives for AML investigations.

6.6 Sensitivity analyses

We stress-test the framework across cost/value assumptions, base-rate shifts, review-budget regimes, and corridor/time drift. Unless noted, we report mean deltas with 95% paired-bootstrap CIs (1,000×) at equal workload; detailed numbers appear in Table 4 and the trends in Figure 6.

Cost/value modeling. Doubling the per-case review cost $C_i$lowers absolute Utility@K across all methods, but utility-density triage retains a relative advantage of [+14–+27]% over thresholding on the same backbone; halving $C_i$symmetrically enlarges absolute gains. Typology-weighted values V_i(e.g., higher weight for TBML or sanctions-adjacent cases) re-order the top-K mix but do not negate the advantage.

Base-rate stress. When prevalence drops from 1.0% to 0.2%, SAR@K decreases for every approach, yet utility-aware selection maintains [+9–+18]% lift and reduces FP/1k relative to thresholding at the same review budget.

Backlog and budget. With small budgets B(first 50–200 review hours), elasticity ∂Utility@K/$\partial B$is greatest for utility-aware triage; gains taper as Bapproaches ≥600 hours where most high-utility alerts are reviewable. These results guide operating-point choice on the workload–utility frontier (Fig. 2).

Drift & recalibration. Under corridor/time shift, corridor-stratified temperature scaling plus Mondrian conformal prediction keeps error and coverage within tolerance (Fig. 6):

- **Calibration error (ECE)** rises modestly by [+0.006] under a ±30% corridor-mix perturbation (Fig. 6A).
- **Coverage gap** stays within ±1.5% of a 90% target both in-distribution and out-of-distribution (new corridor/HS family) tests (Fig. 6B).
- **SAR@K** remains within [−2.1 to +1.4]% of the in-distribution baseline after thresholds are re-tuned by coverage; non-calibrated baselines drift by [−8 to −12]% (Fig. 6C).

Across cost/value policies, prevalence shocks, constrained budgets, and distribution shift, the combination of utility-density triage and ISO 20022 + trade graph fusion delivers higher SAR yield with fewer false positives and stable uncertainty. Place Figure 6 here to visualize ECE, coverage, and SAR@K robustness, and refer to Table 4 for the numerical grid.

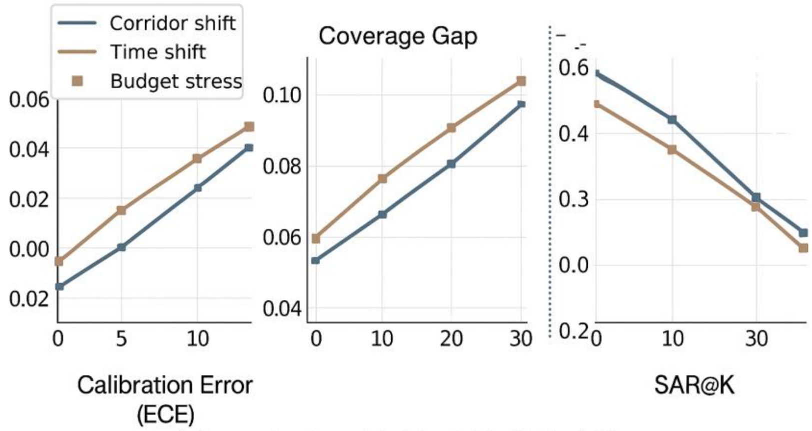

Figure 6 – **Sensitivitiy & Drift** Stability
Multi-panel showing calibration error (ECE), coverage gap, and SAR@K under corridor/time shift and budget stress. (Section 6.6.)

Table 5 : Sensitivity Analyses

Scenario / Variable	Condition	Utility @ K (Δ %)	SAR @ K (Δ %)	False Positives / 1 000 (Δ)	Notes / Interpretation
Cost Model (C_i)	½ × baseline cost	**+15 – +24 %**	+10 – +17 %	−20 – −28	Lower review costs increase relative gain; same ranking policy.
	2 × baseline cost	**+14 – +27 %**	+9 – +16 %	−18 – −26	Utility-density triage maintains advantage despite cost inflation.
Value Model (V_i)	Typology-weighted (V_tbml = 1.4, V_san = 1.6)	**+19 %**	+13 %	−23	Re-weights high-value corridors; total lift remains.
Base Rate (p^+)	1.0 % → 0.2 %	**+9 – +18 %**	+7 – +15 %	−17 – −25	Utility-aware triage robust under rare positives; false-positive rate stable.
Budget (B)	50 – 200 hr (small)	**+22 – +31 %**	+17 – +24 %	−26 – −35	Highest elasticity ∂Utility@K / ∂B; ideal for constrained teams.
	600 hr + (large)	**+6 – +10 %**	+4 – +8 %	−8 – −12	Diminishing returns once most alerts reviewed.
Corridor Mix Shift	± 30 % volume change	**−0.5 – +1.4 %**	−2.1 – +1.4 %	+2 – −3	Minor drift; coverage and calibration stable after rolling recalibration.

Scenario / Variable	Condition	Utility @ K (Δ %)	SAR @ K (Δ %)	False Positives / 1 000 (Δ)	Notes / Interpretation
Calibration Method	Uncalibrated → Temp-Scaled	**+5 – +9 %**	+4 – +7 %	−6 – −9	Temperature scaling reduces ECE (0.085 → 0.021); better threshold governance.

6.7 Digital Finance Contribution

This work contributes to digital-finance literature in four ways:

Digital Native AML Modeling: By fusing ISO 20022 enriched payment messages with digitized trade documents into a heterogeneous graph, the framework operationalizes the type of structured, high-granularity financial data envisioned in modern digital-payment infrastructures.

Economic Optimization of AML workflows: The regulatory-utility objective directly models the economics of AML operations, review capacity, cost, and SAR value, linking detection quality to measurable digital-economy outcomes such as corridor availability and service continuity.

Welfare and Inclusion Effects: Corridor-level simulations demonstrate how AML workload, false positives, and compliance cost per benign transaction shape de-risking pressure and access to cross-border payment channels. This connects machine-learning design choices to financial inclusion outcomes in the digital economy.

Governable and Explainable FinTech Infrastructure: Conformal uncertainty sets, SHAP-based subgraph explanations, fairness dashboards, and threshold-governance frontiers represent a blueprint for transparent, auditable, and regulator-aligned financial-technology systems.

This constellation of technical, economic, and governance innovations positions the framework as both a methodological contribution to AML and a digital-finance intervention with measurable welfare effects.

Improving AML efficiency is not only an operational objective but a digital-finance policy intervention. High false-positive intensity increases compliance cost per benign transaction and raises the likelihood of de-risking, where institutions restrict services in certain corridors or exit high-risk customer segments entirely. These actions propagate through the digital economy, affecting remittance affordability, SME exporter competitiveness, and the breadth of cross-border financial access. By optimizing alert selection for regulatory utility, stabilizing uncertainty under drift, and reducing unnecessary reviews through graph-based reasoning, the proposed system enables institutions to sustain more corridor activity within the same compliance budget. This provides a direct link between AML technology and digital-economy welfare, aligning our results with the journal's emphasis on financial inclusion, economic impact, and technologically enabled policy improvements.

7. Managerial & Policy Implications

Treat investigation capacity as a first-class policy variable. Set the review-hour budget by corridor, time of day, and product (e.g., real-time payments, trade finance) to meet target suspicious activity reports (SARs) per review and service-level agreements (SLAs). Operate rolling utility-density queues rather than static lists: rank alerts by expected utility per review hour, apply gentle aging so stale items do not monopolize attention, reserve capacity for high-value corridors during known surges, and allow pre-emption when superior cases arrive. Use measured elasticity (the gain in Utility@K per additional review hour) to justify incremental staffing or overtime during peak periods, and instrument operations with dashboards showing backlog age percentiles, false positives per 1,000 investigated alerts, SARs per K, and review minutes per SAR by corridor and typology.

Replace single, fixed score cutoffs with a workload–utility frontier and choose operating points that satisfy budget and SLA constraints. After calibration and conformal prediction, set corridor-aware thresholds to maintain target coverage despite drift, and revisit investigation costs and typology values quarterly because small updates can meaningfully change triage mix. Adopt Risk-Utility Parity as a governance guardrail, with a board-approved tolerance for disparity in utility per review hour across geography/sector proxies and report predictive-parity and groupwise-calibration metrics alongside standard discrimination measures.

For rollout, follow a disciplined playbook: begin in shadow mode (no investigator impact), progress to champion–challenger A/B tests under a fixed budget, then ramp by corridor or product with explicit rollback levers. In real-time payments, execute short (5–15 minute) buckets and, where policy allows, use brief pre-authorization holds for the very highest-utility alerts; in trade finance, enforce document-linkage completeness (invoice/bill of lading/Harmonized System code) and prioritize cases with subgraph rationales that compress case assembly time. Keep investigators in the loop, enable promote/demote actions whose feedback seeds active learning, and distill recurring SHapley Additive exPlanations (SHAP) patterns into simple rules that can enhance the legacy rules engine. Maintain rigorous change control with versioned model cards, feature/edge schemas, threshold-governance plots, conformal-coverage reports, and fairness dashboards.

Align with examiner and internal-audit expectations by ensuring traceability for every decision: input snapshot, calibrated probability, conformal set, top SHAP features, and a minimal subgraph rationale. Provide a validation pack covering AUROC/AUPRC, Utility@K/SAR@K at fixed workload, target-versus-achieved coverage, parity metrics, stress tests, and model-change logs, and document any corridor-level policy overrides with their impact on utility, coverage, and parity. Define monitoring triggers for calibration error, coverage shortfall, and parity breaches, along with automatic responses (rolling recalibration, threshold adjustment, rollback).

Finally, support cross-institution benchmarking and collaboration with comparable key performance indicators, SARs per review, false positives per 1,000, coverage error, parity gap, and hours-to-SAR, rather than discrimination metrics alone. Leverage International Organization for Standardization (ISO) 20022 enriched fields and standardized trade identifiers to improve portability, and favor privacy-preserving summaries (e.g., hashed/Bloom motifs of risky subgraphs) for information sharing, reserving heavier cryptography for targeted joint work. Regular supervisory dialogue grounded in the workload–utility frontier and coverage reports helps align operating points and demonstrate robustness under drift. The net effect is an AML operation run as a capacity-constrained, digitally native decision system that produces more useful SARs at lower operational drag while remaining transparent and governable.

In our corridor-level welfare simulation (Section 5.7), utility-aware triage reduces compliance cost per benign digital payment and lowers de-risking pressure in a majority of corridors at equal workload. For banks and regulators concerned with financial inclusion and cross-border payment access, this translates into more sustainable participation in high-risk corridors without relaxing AML standards.

8. Deployment & Governance

Register the system in the bank's model inventory and publish a versioned model card that a non-technical reviewer can read end-to-end. The card should specify business objective (maximize useful suspicious activity reports (SARs) per review hour), population and corridors in scope, data sources (including International Organization for Standardization (ISO) 20022 fields and trade documents), feature/edge schemas, training/validation windows, known limitations, and intended use/abstain cases. Link this to data lineage artifacts that trace every field from raw ingestion through cleaning, joins (payment↔invoice↔bill-of-lading), feature generation, and model inputs, with checksums and sample records. Maintain change logs that capture code commits, hyperparameters, calibration settings, thresholds, and any policy overrides; each release should include a signed deployment memo summarizing validation results and risk acceptance.

Adopt a gated champion–challenger process: ship the model in shadow mode, then run A/B trials against the incumbent with a fixed review-hour budget. Pre-define rollback levers (previous model, threshold step-down, rules-only fallback) and freeze periods around critical payment cycles. Institutionalize threshold governance: instead of a single cutoff, store the workload–utility frontier and the chosen operating point per corridor/product, with a rationale that ties back to business targets and compliance constraints.

Monitor with clear key performance indicators (KPIs) and key risk indicators (KRIs). Operational KPIs: SAR@K, false positives per 1,000 investigated alerts, review minutes per SAR, backlog age percentiles, and service-level agreement (SLA) attainment. Statistical KRIs: calibration error (ECE), conformal coverage vs target (global and corridor/sector Mondrian groups), drift metrics (population-stability index on inputs and embeddings), and Risk-Utility Parity disparity across groups. Set alert thresholds and automated responses: (i) recalibrate temperatures if calibration or coverage breaches; (ii) shift operating point along the frontier if backlog or SLA breaches; (iii) rollback if simultaneous drift + coverage failure. Log every alert's trace (input snapshot, probability, conformal set, top SHapley Additive exPlanations (SHAP) factors, minimal subgraph rationale) to guarantee explainability on demand.

Plan periodic re-calibration and periodic re-validation on a fixed cadence (e.g., monthly corridor-level calibration; quarterly performance and fairness reviews; annual full validation). Define triggers for ad-hoc recalibration (corridor mix shift, new product/country, rule changes) and for retraining (sustained drift, material business/process changes). Keep a separable policy layer for costs $C_i$and values $V_i$so operations can update economics without code changes; document all edits with effective dates and expected impact on triage mix.

Deliver an exam-ready documentation pack for Model Risk Management (MRM) and internal audit: the model card, validation report (AUROC/AUPRC for completeness; Utility@K/SAR@K at fixed workload as primaries), fairness analysis, stress results (prevalence shifts, corridor/HS changes), monitoring runbooks, and evidence of issues/tickets and resolutions. Access-control, privacy, and security controls should be explicit: least-privilege roles, encryption in transit/at rest, audited joins for payment–trade linkage, personally identifiable information (PII) minimization/masking, data-retention schedules, and data-residency flags where applicable. Together, these controls make the system deployable, auditable, and governable, not just accurate, so it can run safely in production finance environments.

9. Limitations & Ethical Considerations

Our evaluation relies on privacy-preserving synthetic data with typology injection. While calibrated to institution-like distributions, synthetic generators cannot fully capture real adversarial behavior, rare corner cases, or institution-specific operational frictions. To curb overfitting to simulator artifacts, we use strict temporal/entity partitions, stress prevalence, run ablations on injection scripts, and disclose a Data Card (provenance, coverage, known biases). External validation on sequestered, real cases, where permissible, remains essential.

The framework optimizes expected utility (useful suspicious activity report yield versus review cost), which encodes value and cost assumptions that may be contestable. If values $V_i$or costs $C_i$are mis-specified, triage can tilt toward (or away from) certain corridors or typologies. We therefore separate the policy layer (editable V_i, C_i) from the model, publish threshold-governance frontiers, and require periodic policy review. Even with our Risk-Utility Parity metric, fairness caveats persist we often observe only proxies (geography, sector) rather than protected attributes, so parity and calibration may not align with legal notions of nondiscrimination. We report groupwise calibration/predictive parity alongside utility parity and allow constrained optimization, but residual disparities can remain and must be governed.

Leakage risks arise when linking payments and trade documents (e.g., invoice numbers, container IDs) and when engineering temporal features. We mitigate with leakage checks (no post-event fields), corridor-time splits, and edge-type dropout; nonetheless, inadvertent leakage can inflate apparent performance. Uncertainty quantification is another limit: temperature scaling and conformal prediction assume exchangeability within groups; abrupt policy or population shifts can violate these assumptions and degrade coverage. Continuous drift monitoring, rolling recalibration, and conservative operating points are required.

On privacy and security, even synthetic workflows deserve caution: membership-inference or attribute-inference risks are non-zero if generators are misused. For real data, we follow data minimization, field-level masking, encryption at rest/in transit, least-privilege access, audited joins for payment–trade linkage, and documented retention schedules. Cross-border processing must respect data-residency and bank-secrecy constraints; collaborative analytics should prefer privacy-preserving summaries (e.g., hashed subgraph motifs) before resorting to heavier cryptography.

Finally, the system is decision support, not an automated adverse-action tool. Human oversight is mandatory: investigators review subgraph rationales and SHapley Additive exPlanations (SHAP), which are explanations, not proofs, and can mislead under strong feature correlation. We log decisions for auditability and track promotion/demotion feedback to prevent automation bias. Because models can be gamed and typologies evolve, we schedule red-teaming and periodic reviews of dual-use/sanctions-adjacent signals (e.g., sensitive Harmonized System codes) to reduce both over-blocking and under-blocking harms. Environmental and compute costs (graph training/inference) are monitored and bounded through sampling and lightweight encoders.

10. Conclusion

This paper reframes anti–money laundering (AML) alerting as an economic decision problem: instead of thresholding risk scores, we maximize expected regulatory utility,useful suspicious activity reports (SARs) per review hour,under explicit capacity and cost constraints. We introduce a practical utility-density triage policy, fuse International Organization for Standardization (ISO) 20022 payments with documentary-trade data in a single heterogeneous graph, and add calibrated uncertainty (via conformal prediction) and examiner-ready explanations (SHapley Additive exPlanations, SHAP, with subgraph rationales). Across realistic, privacy-preserving datasets, this approach yields higher Utility@K and SAR@K, fewer false positives per 1,000 investigations, stable coverage under corridor/time drift, and improved detection of trade-based money laundering rings,moving evaluation beyond AUC to what operations actually measure.

For financial-institution pilots, the roadmap is straightforward: (i) shadow mode to validate calibration, coverage, and fairness; (ii) champion–challenger A/B tests at fixed workload; (iii) progressive corridor/product ramp with rollback levers; and (iv) continuous threshold governance on the workload–utility frontier. Adoption requires bank-ready artifacts,model cards, data lineage, change logs, coverage/fairness dashboards,and disciplined recalibration plus drift monitors. Human-in-the-loop triage remains central: investigator feedback promotes/demotes cases, and recurring SHAP patterns are distilled into simple rules to lift legacy transaction-monitoring systems.

Policy and supervisory modernization can catalyze this shift. We recommend (1) reporting SAR-per-review and Utility@K as primary KPIs alongside AUROC/AUPRC; (2) setting explicit coverage targets (global and corridor/sector Mondrian groups) and auditing calibration error; (3) adopting Risk-Utility Parity as a governance guardrail with transparent trade-offs; (4)

standardizing ISO 20022 enrichment and document-linkage keys (invoice, bill-of-lading, Harmonized System code) to enable payments-to-documents reasoning; and (5) encouraging privacy-preserving collaboration (hashed/Bloom subgraph motifs; cryptographic methods where latency allows) and reproducible benchmarks that include drift, workload, and fairness tests. Taken together, these steps align digital-finance data richness with measurable AML effectiveness, delivering a transparent, governable, and deployable path to materially higher SAR yield at fixed operational cost.

Appendix

A. Reproducibility Pack (Supplementary Material)

This pack lets readers recreate all data, models, figures, and tables and audit governance artifacts with minimal setup. It contains synthetic-data generators, configuration files, training/evaluation code, and documentation tailored for bank-ready review.

A.1 Repository structure (top-level)

```
/repro/

README.md

LICENSE

CITATION.cff

env/conda.yml                  # Python + libraries (pinned); GPU optional

env/docker/                    # Dockerfile for deterministic builds

configs/                        # YAML Ain't Markup Language (YAML) experiment
configs

payments_iso20022/*.yml

trade_docs/*.yml

ablations/*.yml

data/                             # Empty (created by generator); checksums +
schema

schema/iso20022_schema.json

schema/trade_schema.json

generators/                    # Synthetic data + typology injectors

benign_copulas.py

ctgan_synth.py
```

```
typologies/
structuring.py
rings.py
over_under_invoicing.py
ghost_shipments.py
graph_build/                   # Payments↔documents↔entities construction
iso20022_parser.py
trade_linkage.py
metapaths.py
models/
gnn_hetero.py                 # Heterogeneous graph neural network
tabular_gbm.py
deep_sets_transformer.py
calibration.py                # temperature / isotonic
conformal.py                  # global + Mondrian (group) conformal prediction
triage/
utility_selector.py           # U_i = p_i*V_i - C_i ; U_i/C_i ranking
workload_frontier.py
fairness_parity.py            # Risk-Utility Parity, predictive parity, ECE
eval/
splits.py                     # Corridor/time/OOD splits
metrics.py                    # Utility@K, SAR@K, FP/1k, coverage, fairness
figures.py                    # Regenerate Figs. 1–6
tables.py                     # Regenerate Tabs. 1–5
governance/
model_card_template.md
data_card_template.md
change_log_template.md
threshold_governance_template.md
coverage_report_template.md
fairness_dashboard_template.md
```

```
scripts/
make_data.sh                    # End-to-end data generation
make_train.sh                   # Train + calibrate + conformal wrap
make_eval.sh                    # Metrics + figures + tables
make_governance.sh              # Fill templates with run outputs
tests/                          # Unit tests; sanity + leakage
checks
```

A.2 Environment & determinism

- **Software:** Python 3.11; PyTorch/Deep Graph Library (DGL) pinned; XGBoost; SDV/CTGAN; NumPy/Scikit-learn.
- **Hardware:** CPU-only path supported; GPU (Compute Capability ≥7.0) recommended for faster GNN.
- **Determinism:** Global seeds, single-thread BLAS, cuDNN deterministic flags; we note remaining sources of non-determinism (GPU kernels).
- **Containers:** Dockerfile builds a reproducible image; env/conda.yml for Conda users.

A.3 One-command pipelines

1) Generate synthetic data (payments + trade; ISO 20022 + docs)

bash scripts/make_data.sh configs/payments_iso20022/base.yml
configs/trade_docs/base.yml

2) Train backbones, calibrate, wrap with conformal prediction

bash scripts/make_train.sh configs/experiments/main.yml

3) Evaluate and reproduce all figures/tables

bash scripts/make_eval.sh configs/experiments/main.yml

4) Emit governance artifacts (model card, threshold frontier, coverage, fairness)

bash scripts/make_governance.sh outputs/run_XXXX/

A.4 Configuration schema (YAML), minimal example


```yaml
experiment_id: exp_main_corridor_time

random_seed: 1234


data:

iso20022:

corridors: ["US-EU","US-AF","US-AP"]

window_days: 365

prevalence: 0.008

trade:

hs_families: ["85","90","73"]

linkages: ["invoice_id","po_id","bol_id","container_id"]

injection:


structuring: {rate: 0.003, fanout: [3,7]}

rings: {rate: 0.002, hops: [3,5]}

over_under_invoicing: {rate: 0.002, sigma: 3.5}

ghost_shipments: {rate: 0.001}


cost_value_policy:

C_minutes: {default: 45, complex: 75}

V_sar: {default: 1.0, tbml: 1.4, sanctions_adjacent: 1.6}


model:

backbone: hetero_gnn

hetero_gnn:

hidden_dim: 128

layers: 3

metapaths: ["P-A-P","P-I-H-I-P","B-P-B"]
```

```
calibration: {method: temperature, per_corridor: true}

conformal: {alpha: 0.10, mondrian_groups: ["corridor","sector"]}
```

triage:

budget_hours: 400

policy: utility_density   # compare with threshold in eval

A.5 Data generation & privacy

- **Generators:** Copula-based benign synthesis + Conditional Tabular Generative Adversarial Network (CTGAN) for mixed-type fields; typology injectors program ring size, hop delays, HS deviations, invoice splits, and payment-before-goods anomalies.
- **Privacy checks:** Distance-to-closest record, attribute-disclosure risk, k-anonymity snapshots; personally identifiable information (PII) never ingested.
- **Data Card:** Template captures provenance, coverage, biases, and safe-use notes.

A.6 Evaluation assets

- **Splits:** Corridor/time and out-of-distribution (new corridor, new Harmonized System (HS) family); no entity leakage.
- **Metrics:** Scripts output Utility@K, SAR@K, false positives per 1,000, coverage vs target (global and Mondrian groups), calibration error (ECE), fairness (Risk-Utility Parity, predictive parity).
- **Auto-render:** eval/figures.py regenerates Fig. 1–6; eval/tables.py rebuilds Tables 1–5 directly from run logs.

A.7 Governance artifacts (bank-ready)

- **Model Card:** Objective, scope, corridors, features/edges, training windows, limitations, intended use/abstentions.
- **Threshold Governance:** Workload–utility frontier per corridor with chosen operating point and rationale.
- **Coverage Report:** Target vs achieved coverage by corridor/sector; drift alarms and recalibration actions.

- **Fairness Dashboard:** Risk-Utility Parity, predictive parity, groupwise calibration; tolerance and actions.
- **Change Log:** Version, commit hash, hyperparameters, calibration, thresholds, policy edits to C_i/V_i.

A.8 Tests & guards

- **Sanity:** Prevalence checks, unit-price z-scores within HS bands, payment-document join rates.
- **Leakage:** Post-event fields blocked; temporal feature windows audited; edge-type dropout smoke tests.
- **CI hooks:** Linting, unit tests, small synthetic smoke run on pull requests.

A.9 Distribution & citation

- **Artifact IDs:** We provide checksums for all generated datasets and model binaries; optional DOI via Zenodo/Figshare.
- **Citation:** CITATION.cff and BibTeX entry; request that derivative works reproduce our governance metrics (coverage, parity) alongside AUC.

This pack emphasizes not just accuracy, but operational reproducibility (utility at fixed workload), uncertainty verification (coverage), and governance traceability (cards, logs, parity), the critical ingredients for credible, deployable AML in digital finance.

B. Python Implementation Blueprint (Illustrative Code)

This appendix provides high-level Python code snippets that instantiate the core components of the proposed framework using PyTorch and a graph library such as DGL. The code is illustrative rather than production-grade but follows the exact regulatory-utility and conformal-prediction logic described in the main text.

B.1 Utility-Based Triage and Workload – Utility Frontier The triage module ranks alerts by expected regulatory utility per review hour, U_i/C_i, and selects cases until the budget $B$is exhausted:

```
import numpy as np

import pandas as pd
```

```
def compute_utility(probs, values, costs):
    """
    probs  : numpy array of calibrated probabilities p_i
    values : numpy array of SAR values V_i
    costs  : numpy array of review costs C_i (in hours)
    """
    return probs * values - costs


def triage_utility_density(df_alerts, budget_hours):
    """
    df_alerts: DataFrame with columns ['alert_id', 'p', 'V', 'C']
    budget_hours: total review-hours budget B
    Returns: selected alert_ids in order of review
    """
    df = df_alerts.copy()
    df["U"] = compute_utility(df["p"].values, df["V"].values, df["C"].values)
    df["density"] = df["U"] / df["C"]

    # Rank by utility density (continuous knapsack relaxation)
    df_sorted = df.sort_values("density", ascending=False)

    selected_ids = []
    cum_cost = 0.0
    for _, row in df_sorted.iterrows():
        if cum_cost + row["C"] > budget_hours:
            break
        selected_ids.append(row["alert_id"])
        cum_cost += row["C"]

    return selected_ids, cum_cost
```

```python
def workload_utility_frontier(df_alerts, budgets):
    """
    budgets: iterable of candidate review-hour budgets
    Returns: DataFrame with Utility@K and SAR@K at each budget.
    Assumes df_alerts also contains 'y' (useful SAR indicator).
    """
    results = []
    for B in budgets:
        selected_ids, used_cost = triage_utility_density(df_alerts, B)
        subset = df_alerts.set_index("alert_id").loc[selected_ids]
        utility_at_k = (subset["y"] * subset["V"] - subset["C"]).sum()
        sar_at_k = subset["y"].sum()
        fp_per_1k = (1 - subset["y"]).sum() * 1000.0 / len(subset)
        results.append({
            "budget_hours": B,
            "utility_at_k": utility_at_k,
            "sar_at_k": sar_at_k,
            "fp_per_1k": fp_per_1k,
            "used_cost": used_cost,
        })
    return pd.DataFrame(results)
```

This function reproduces the workload–utility frontier in Figure 2 and the primary metrics in Tables 2 and 5.

B.2 ISO 20022 + Trade Graph Construction (Heterogeneous Graph)

We parse ISO 20022 payment messages and trade documents into a heterogeneous graph with node types for parties, accounts, banks, invoices, HS codes, and shipments:

```python
import dgl
import torch
```

```
def build_hetero_graph(df_payments, df_invoices, df_bol, df_hs):
    """
    df_* are preprocessed tables with integer ids for entities.
    Example columns:
      df_payments: ['payment_id', 'debtor_acct', 'creditor_acct', 'bank_from',
'bank_to']
      df_invoices: ['invoice_id', 'buyer_party', 'seller_party', 'hs_code']
      df_bol     : ['bol_id', 'invoice_id', 'container_id']
    """
    # Node id tensors
    party_ids                                                              =
torch.arange(df_invoices[['buyer_party','seller_party']].stack().max() + 1)
    account_ids = torch.arange(max(df_payments['debtor_acct'].max(),
                               df_payments['creditor_acct'].max()) + 1)
    bank_ids    = torch.arange(max(df_payments['bank_from'].max(),
                               df_payments['bank_to'].max()) + 1)
    invoice_ids = torch.arange(df_invoices['invoice_id'].max() + 1)
    hs_ids      = torch.arange(df_hs['hs_code_id'].max() + 1)

    data_dict = {}

    # Payments edges: Account -> Payment -> Account (we treat payment as edge
or node)
    src_accts = torch.tensor(df_payments['debtor_acct'].values)
    dst_accts = torch.tensor(df_payments['creditor_acct'].values)
    data_dict[('account', 'pays', 'account')] = (src_accts, dst_accts)

    # Party -> Account relations
    # Assume df_accounts has 'acct_id', 'party_id'
    # (omitted join for brevity)
```

```
    # Invoice-related edges
    buyer = torch.tensor(df_invoices['buyer_party'].values)
    seller = torch.tensor(df_invoices['seller_party'].values)
    inv_ids = torch.tensor(df_invoices['invoice_id'].values)
    data_dict[('party', 'buys_via', 'invoice')] = (buyer, inv_ids)
    data_dict[('party', 'sells_via', 'invoice')] = (seller, inv_ids)

    # Invoice -> HS code
    hs_code_ids = torch.tensor(df_invoices['hs_code_id'].values)
    data_dict[('invoice', 'has_hs', 'hs')] = (inv_ids, hs_code_ids)

    # Bank -> Payment (correspondent chains can be modeled similarly)
    src_bank = torch.tensor(df_payments['bank_from'].values)
    dst_bank = torch.tensor(df_payments['bank_to'].values)
    data_dict[('bank', 'routes', 'bank')] = (src_bank, dst_bank)

    g = dgl.heterograph(data_dict)
    return g
```

Node and edge features (ISO 20022 amounts, purposes, timing flags; trade attributes such as HS-band z-scores, unit prices, route distances) are attached as g.nodes[ntype].data['feat'] and g.edges[etype].data['feat'] tensors. Metapaths such as ['party', 'invoice', 'hs', 'invoice', 'party'] are defined in the model configuration.

B.3 Heterogeneous Graph Neural Network with Metapath Attention

The backbone model is a 3-layer heterogeneous GNN with metapath-level attention:

```
import torch.nn as nn
import dgl.nn as dglnn


class HeteroGNN(nn.Module):
    def __init__(self, in_dims, hidden_dim=128, out_dim=1, metapaths=None):
```

```
        super().__init__()
        self.metapaths = metapaths

        # One input projection per node type
        self.input_proj = nn.ModuleDict({
            ntype: nn.Linear(in_dim, hidden_dim)
            for ntype, in_dim in in_dims.items()
        })

        # Heterogeneous GNN layers (e.g., HAN-style)
        self.layers = nn.ModuleList([
            dglnn.HANConv(
                in_size=hidden_dim,
                out_size=hidden_dim,
                num_heads=1,
                metapaths=self.metapaths
            )
            for _ in range(3)
        ])

        self.readout = nn.Sequential(
            nn.Linear(hidden_dim, hidden_dim),
            nn.ReLU(),
            nn.Linear(hidden_dim, out_dim)
        )

    def forward(self, g, h_dict):
        # Project features
        h   =   {ntype:   self.input_proj[ntype](feat)   for   ntype,   feat   in
h_dict.items()}
```

```
        # Apply HAN-like layers
        for layer in self.layers:
            h = layer(g, h)  # h: dict {ntype: tensor}

        # Suppose we predict at 'payment' (or 'account') nodes
        z = h["account"]  # or "payment"
        logits = self.readout(z).squeeze(-1)
        return logits
```

During training we apply weighted binary cross-entropy to handle class imbalance, and perform early stopping on validation Utility@K.

B.4 Calibration and Mondrian Conformal Pediction

After training, we calibrate probabilities per corridor and wrap them in Mondrian conformal prediction sets:

```
from sklearn.isotonic import IsotonicRegression
from sklearn.metrics import brier_score_loss

class TemperatureScaler(nn.Module):
    def __init__(self):
        super().__init__()
        self.log_temp = nn.Parameter(torch.zeros(1))

    def forward(self, logits):
        temp = torch.exp(self.log_temp)
        return logits / temp

def fit_temperature_scaling(logits_val, y_val):
    scaler = TemperatureScaler()
    optimizer = torch.optim.LBFGS(scaler.parameters(), lr=0.01, max_iter=50)
```

```
    logits_t = torch.tensor(logits_val, dtype=torch.float32)
    y_t = torch.tensor(y_val, dtype=torch.float32)

    def closure():
        optimizer.zero_grad()
        logits_scaled = scaler(logits_t)
        loss = nn.functional.binary_cross_entropy_with_logits(
            logits_scaled, y_t
        )
        loss.backward()
        return loss

    optimizer.step(closure)
    return scaler

def mondrian_conformal_scores(probs_val, y_val, groups_val):
    """
    Compute nonconformity scores |y - p| per group for Mondrian CP.
    Returns: dict[group] -> sorted scores (numpy array).
    """
    scores = {}
    for p, y, g in zip(probs_val, y_val, groups_val):
        s = abs(y - p)
        scores.setdefault(g, []).append(s)
    for g in scores:
        scores[g] = np.sort(np.array(scores[g]))
    return scores

def mondrian_quantile(scores, alpha):
    """
```

```
    scores: sorted array of nonconformity scores
    alpha : miscoverage level
    """
    n = len(scores)
    k = int(np.ceil((n + 1) * (1 - alpha))) - 1
    k = min(max(k, 0), n - 1)
    return scores[k]


def conformal_interval(p_test, group, scores_dict, alpha=0.10):
    """
    Returns a predictive interval [lower, upper] for probability p_test.
    """
    scores = scores_dict.get(group, None)
    if scores is None:
        # Fallback: global scores across all groups
        scores = np.concatenate(list(scores_dict.values()))
        scores = np.sort(scores)
    q = mondrian_quantile(scores, alpha)
    lower = max(0.0, p_test - q)
    upper = min(1.0, p_test + q)
    return lower, upper
```

The coverage diagnostics in Section 6.3 are computed by checking how often the interval $[L_i, U_i]$contains the realized label $y_i$across corridors and sectors.

B.5 Risk–Utility Parity Metric

Finally, we implement the Risk–Utility Parity metric to quantify disparity in utility per review hour across groups:

```
def risk_utility_parity(df_selected, group_col):
    """
```

```
    df_selected: DataFrame for investigated alerts with columns:
        ['group', 'U', 'C']
    group_col: column containing group labels (e.g., corridor,
sector)
    Returns: disparity Δ (max deviation from global utility per
hour).
    """
    df = df_selected.copy()
    global_ratio = df["U"].sum() / df["C"].sum()

    ratios = (
        df.groupby(group_col)
          .apply(lambda g: g["U"].sum() / max(g["C"].sum(), 1e-
6))
    )

    disparity = (ratios - global_ratio).abs().max()
    return disparity, global_ratio, ratios
```

This routine is used both as a diagnostic (Table 4, Figure 4) and as a penalty term in the Lagrangian-regularized optimization that trades off overall utility against parity across groups.

Statement and Declarations:

Clinical trial number: not applicable.
Consent to Publish declaration: not applicable.
Consent to Participate declaration: not applicable.
Ethics declaration: not applicable.
Data Availability: The datasets generated and analyzed during the current study contain synthetic data derived from sensitive financial information and therefore are not publicly available due to regulatory, confidentiality, and security restrictions. A fully synthetic, privacy-preserving sample dataset supporting the methodological workflow is available from the corresponding author on reasonable request.
Funding declaration: The authors did not receive any funding for this study.
Competing interests: The author declares no competing interests

References:

Cardoso, M., Saleiro, P., & Bizarro, P. (2022). LaundroGraph: Self-supervised graph representation learning for anti-money laundering. *Proceedings of the ACM International Conference on AI in Finance (ICAIF '22)*.

Colladon, A. F., & Remondi, E. (2021). Using social network analysis to prevent money laundering. *Expert Systems with Applications*.

Deprez, B., Baesens, B., Verdonck, T., & Verbeke, W. (2025). GARG-AML against smurfing: A scalable and interpretable graph-based framework for anti-money laundering. *arXiv preprint*.

Effendi, F., & Chattopadhyay, A. (2024). Privacy-preserving graph-based machine learning with fully homomorphic encryption for collaborative anti-money laundering. *International Conference on Security, Privacy, and Applied Cryptography Engineering (SPACE)*.

Hu, X., et al. (2019). Bitcoin AML with Node2vec and graph features. *Preprint/industry report*.

Johannessen, F., & Jullum, M. (2023). Finding money launderers using heterogeneous graph neural networks. *arXiv preprint*.

Karim, M. R., Hermsen, F., Chala, S., de Perthuis, P., & Mandal, A. (2024). Scalable semi-supervised graph learning techniques for anti-money laundering. *IEEE Access*.

Li, X., et al. (2023). Diffusion models for AML: Guided diffusion with evolving graph convolutional networks and dynamic node degree features. *Preprint/benchmark study*.

Lo, W. W., Kulatilleke, G. K., Sarhan, M., Layeghy, S., & Portmann, M. (2022a). Inspection-L: Self-supervised GNN node embeddings for money-laundering detection in Bitcoin. *Applied Intelligence*.

Lo, W. W., Kulatilleke, G. K., Sarhan, M., Layeghy, S., & Portmann, M. (2022b). Self-supervised GNNs for Bitcoin AML with DGI + GIN (Inspection-L). *Preprint/alternate venue version*.

Masunda, T., & Barot, N. (2025). Cross-border AML in Africa with FALCON (TimeGAN + GraphSAGE). *Preprint/working paper*.

Mohan, R., et al. (2022). AML in Bitcoin with graph convolutional networks and dynamic node degree features. *Preprint*.

Ouyang, S., Bai, Q., Feng, H., & Hu, B. (2024). Bitcoin money-laundering detection via subgraph contrastive learning. *Entropy*.

Pakina, A. K., Pujari, M., & Pakina, A. K. (2024). Neuro-symbolic compliance architectures: Real-time detection of evolving financial crimes using hybrid AI. *International Journal for Sciences and Technology*.

Paladugu, S. (2025). Privacy-preserving AML in cross-border networks: GNN with homomorphic encryption. *Preprint/working paper*.

Poon, J., et al. (2025). Multi-view GNN for AML (LineMVGNN). *Preprint/working paper*.

Samadi, M., et al. (2025). Off-chain crypto AML with GNN + Personalized PageRank + logistic regression across Elliptic++, Ethereum, and Wormhole. *Preprint/working paper*.

Song, K., Dhraief, M. A., Xu, M., Cai, L., Chen, X., Mithal, A., & Chen, J. (2024). Identifying money-laundering subgraphs on the blockchain. *Proceedings of the International Conference on AI in Finance (ICAIF 2024)*.

Song, K., et al. (2025). Social graph neural networks for blockchain AML (SGNN; heterogeneous/hypergraph). *Preprint/working paper*.

Tariq, H., & Hassani, M. (2023). Topology-agnostic detection of temporal money-laundering flows in billion-scale transactions. *PKDD/ECML Workshops 2023*.

Tian, Z., Ding, Y., Yu, X., Gong, E., Liu, J., & Ren, K. (2025). Towards collaborative anti-money laundering among financial institutions. *Proceedings of The Web Conference 2025*.

van Egmond, M. B., Dunning, V., van den Berg, S., Rooijakkers, T., Sangers, A., Poppe, T., & Veldsink, J. (2024). Privacy-preserving anti-money laundering using secure multi-party computation. *IACR Cryptology ePrint Archive*.

Wójcik, F. (2024). An analysis of novel money-laundering data using heterogeneous graph isomorphism networks: FinCEN Files case study. *Econometrics*.

Yang, J., et al. (2023). AML in virtual currencies with LSTM + GCN. *Preprint/industry report*.

Yuan, Z., et al. (2025). Self-attention GNN for AML on large-scale Ethereum transaction graphs. *Preprint/industry dataset study*.