Artificial Intelligence for Compliance and Regulatory Reporting - Automating Anti-Money Laundering (AML) Detection in Financial Services

Author: Lucia Fernández **Affiliation:** Department of Information Technology, University of Buenos Aires (Argentina) **Email:** lucia.fernandez@uba.ar

Abstract

Anti-Money Laundering (AML) detection is a critical regulatory and operational function for financial institutions. Traditional rule-based systems capture known typologies but produce high false positive rates, operational burden, and limited adaptive capacity against evolving threats. Recent advances in artificial intelligence (AI) and machine learning (ML) including supervised classifiers, unsupervised anomaly detection, graph learning, and explainable AI (XAI) provide the potential to transform AML operations: improving detection quality, prioritizing alerts, and automating regulatory reporting (e.g., suspicious activity reports, SARs). This paper presents a comprehensive, scholarly treatment of AI for AML: (1) a systematic problem formulation mapping AML tasks to ML objectives; (2) a detailed review of modeling approaches (statistical baselines, supervised learning, unsupervised methods, graph-based models, temporal sequence models, and hybrid systems); (3) mathematical formulations for core tasks (anomaly scoring, link prediction, temporal point process modeling, and risk scoring); (4) an end-to-end system architecture for production deployment with considerations for data engineering, latency, model governance, auditability, human-in-the-loop triage, and regulatory reporting workflows; (5) evaluation methodologies appropriate to highly imbalanced, non-stationary data (including offline metrics, backtesting, and controlled trials); and (6) ethical, legal, and operational concerns such as fairness, privacy, adversarial abuse, and interpretability. We include reproducible experiment blueprints, recommended feature sets, and practical recommendations for staged adoption. The manuscript synthesizes academic research and industry practice to deliver an actionable roadmap for institutions seeking to modernize AML through AI while maintaining regulatory compliance and operational resilience.

Keywords: anti-money laundering, anomaly detection, graph neural networks, explainable AI, transaction monitoring, regulatory reporting, suspicious activity reports (SAR), supervised learning, unsupervised learning, compliance automation

1. Introduction

Money laundering and financial crime impose serious economic and societal harms, while regulators worldwide require financial institutions to maintain effective AML programs, including transaction monitoring, customer due diligence, and timely suspicious activity reporting. Historically, many AML systems are rule-based: transaction rules based on thresholds (e.g., large cash deposits), velocity rules, or typology triggers (Financial Action Task Force, FATF, guidance). Such systems generate large

volumes of alerts with low precision, creating significant manual investigation overhead and the risk that true suspicious activity is missed due to alert fatigue.

Artificial intelligence promises to improve AML by learning patterns from historical labeled cases, detecting novel anomalies, and uncovering complex networked relationships (e.g., layering and structuring across multiple accounts) that transcend simple rule thresholds. However, AML presents distinctive technical and regulatory challenges: extremely high class imbalance (fraud/AML cases are rare), evolving adversarial behavior, strict privacy and data-retention constraints, the need for auditable explanations for regulatory filings, and legal requirements for timely reporting. This paper provides an in-depth treatment of AI methods mapped to AML workflows, showing how models and systems can be designed, validated, and governed to meet both technical performance and compliance obligations.

2. Problem Definition and AML Task Taxonomy

2.1 Core operational tasks in AML

We decompose AML into a set of operational tasks that AI can assist or automate:

1. **Transaction monitoring & alert generation (real-time / near-real-time).** Continuously analyze streams of transactions to detect anomalous or policy-violating activity and generate case alerts for investigation.
2. **Network and link analysis.** Identify suspicious clusters of accounts, beneficial owners, or transaction paths indicative of money-laundering typologies (e.g., layering).
3. **Customer risk scoring and segmentation.** Maintain dynamic customer risk profiles using transactional behavior and external data.
4. **Case prioritization and triage.** Rank alerts to focus human investigator effort where expected value (true positive probability × potential harm) is greatest.
5. **Regulatory reporting and SAR drafting.** Assist or automate the preparation of Suspicious Activity Reports (SARs) with auditable narratives and evidence.
6. **Typology discovery and adaptive detection.** Discover emerging patterns of abuse via unsupervised learning and update detection rules/models.

2.2 Formalizing AML as machine learning problems

AML tasks map to several ML problem classes:

- **Binary classification:** label transactions or account-periods as suspicious or benign given features; heavily imbalanced.
- **Anomaly detection / novelty detection:** learn normal behavior and flag deviations when labeled examples are scarce.

- **Link prediction / community detection:** operate over transaction graphs to find suspicious connections.
- **Temporal sequence modeling / point processes:** model event arrival times and sequences (e.g., sudden bursts of transfers).
- **Ranking / learning to rank:** prioritize alerts or cases for investigation.

These tasks often interplay; for example, graph features (derived from network analysis) serve as inputs to a supervised classifier or anomaly scorer.

3. Data Sources, Feature Engineering, and Data Quality

3.1 Data sources available to financial institutions

- **Core transaction data:** deposits, withdrawals, transfers (payer/payee IDs, amounts, timestamps, channels).
- **Customer data:** KYC attributes, beneficial ownership, risk ratings.
- **Account metadata:** account opening dates, product types, limits.
- **External data:** sanctions lists, politically exposed persons (PEP) lists, adverse media, corporate registries.
- **Case history:** labeled SARs and investigator outcomes (true positives, false positives).

3.2 Feature engineering principles

Construct features at multiple levels:

- **Transaction-level features:** amount, channel, country pair, merchant category code, time of day, rounding patterns.
- **Aggregated statistics:** moving averages, counts over sliding windows (e.g., last 24 hours, 30 days), standard deviation of amounts.
- **Behavioral features:** recency, frequency, monetary (RFM), velocity indicators.
- **Network features:** degree centrality, betweenness, PageRank, clustering coefficients, connected component size, shortest path to sanctioned entities.
- **Graph motifs:** counts of patterns (e.g., chains, loops) indicative of layering.
- **Temporal features:** burstiness measures, inter-transaction intervals can be modeled via Hawkes processes.
- **Derived features for explainability:** textual templates for reasons (e.g., “rapid outbound transfers to high-risk jurisdiction”).

Quality considerations: canonicalization of entity IDs, robust handling of missing or inconsistent KYC data, deduplication, and provenance tracking to enable audit trails.

3.3 Labeling and ground truth

Obtaining ground truth is challenging. Labels come from SARs and investigator outcomes, but these are noisy (investigator decisions can be subjective) and often delayed. Practices to manage labeling issues:

- **Label propagation & weak supervision:** combine multiple signals (rule triggers, external matches) to create training labels with estimated noise.
- **Active learning:** focus investigator labeling effort on uncertain examples to improve model training efficiency.
- **Time-aware labeling:** careful alignment of model training windows to avoid leakage (only use information available at detection time).

4. Modeling Approaches

This section surveys ML methods applicable to AML, their mathematical formulations, strengths, and limitations.

4.1 Statistical and rule-based baselines

Rule systems define Boolean conditions (e.g., amount > X AND country = Y) or thresholds on engineered features. They are interpretable and simple to audit but have low precision and slow adaptation to new typologies. Statistical baselines (e.g., z-score on log amounts) provide simple anomaly flags.

4.2 Supervised learning

Supervised binary classification trains models $\hat{y} = h_\theta(x) \in [0,1]$ to predict the probability of suspiciousness. Common model families:

- **Logistic regression / generalized linear models (GLMs):** interpretable, fast; can integrate with monotonic constraints for fairness.
- **Tree ensembles (Random Forests, XGBoost, LightGBM):** widely used for tabular data, handle heterogenous features and interactions.
- **Deep neural networks:** feedforward MLPs and architectures for sequences/temporal aggregation; require more data and careful calibration.

Loss functions and class imbalance: use weighted cross-entropy, focal loss, or resampling. Calibration of predicted probabilities is important for downstream prioritization (Platt scaling, isotonic regression).

Mathematical formulation (standard logistic): Given labeled dataset {(xi,yi)}i=1N\{(x_i, y_i)\}_{i=1}^N{(xi,yi)}i=1N with yi∈{0,1}y_i \in \{0,1\}yi∈{0,1}, minimize negative log-likelihood:

L(θ)=−∑i=1N[yilogσ(θ⊤xi)+(1−yi)log(1−σ(θ⊤xi))]+λ‖θ‖22.\mathcal{L}(\theta) = -\sum_{i=1}^N \big[y_i \log \sigma(\theta^\top x_i) + (1-y_i)\log(1-\sigma(\theta^\top x_i))\big] + \lambda \|\theta\|_2^2.L(θ)=−i=1∑N[yilogσ(θ⊤xi)+(1−yi)log(1−σ(θ⊤xi))]+λ‖θ‖22.

4.3 Unsupervised and semi-supervised anomaly detection

When labeled data are scarce or unreliable, learning the distribution of normal behavior p(x)p(x)p(x) and flagging low-probability observations is an approach.

Methods include:

- **Density estimation:** Gaussian mixture models, kernel density estimation.
- **One-class SVMs:** learn boundary of normal class.
- **Autoencoders / Variational Autoencoders (VAE):** reconstruct input; high reconstruction error signals anomalies.
- **Isolation Forests:** randomly partition features and use path length as anomaly measure.
- **Deep generative models (GANs) for anomaly detection:** adversarial training to learn normal data manifold.

Challenge: adversarial adaptation and evolving normal behavior; require continuous retraining and concept drift detection.

4.4 Graph and relational learning

Money-laundering is inherently relational. Graph representations G=(V,E,A)G = (V, E, A)G=(V,E,A) where nodes VVV are accounts or entities and edges EEE are transactions, enable powerful structural detection.

Techniques:

- **Feature extraction on graphs:** compute centrality, motif counts, community features as inputs to classifiers.
- **Graph neural networks (GNNs):** message-passing frameworks that learn node embeddings capturing structural and attribute information (e.g., GraphSAGE, GAT). GNNs support both node-level (flag an account) and edge-level (flag a transaction) predictions.
- **Graph anomaly detection:** models that detect anomalous subgraphs or unusual edge patterns.

- **Link prediction:** estimate the likelihood of an edge becoming suspicious or connecting to high-risk nodes.

Mathematical sketch (GNN message passing):
For node vvv, at layer l+1l+1l+1:

hv(l+1)=UPDATE(hv(l),AGG({hu(l):u∈N(v)})),h_v^{(l+1)} = \mathsf{UPDATE}\Big(h_v^{(l)}, \mathsf{AGG}\big(\{h_u^{(l)}: u \in \mathcal{N}(v)\}\big)\Big),hv(l+1)=UPDATE(hv(l),AGG({hu(l):u∈N(v)})),

with initial hv(0)=xvh_v^{(0)} = x_vhv(0)=xv.

GNNs can capture propagation patterns (e.g., sudden emergence of a hub receiving funds from many nodes).

4.5 Temporal and sequence models

Temporal modeling captures behavioral evolution and scheduling patterns:

- **Recurrent neural networks (RNNs), LSTMs, GRUs:** model sequences of transactions for a given account.
- **Temporal point processes (e.g., Hawkes processes):** model self-exciting behavior (bursts) in transaction arrivals; useful for modeling cascading transfers.
- **Temporal GNNs / dynamic graph embeddings:** incorporate timing into graph representations.

4.6 Hybrid systems and ensembles

Combining models yields stronger performance and operational flexibility:

- **Rule + ML hybrid:** rules capture regulatory constraints while ML filters improve precision and reduce false positives.
- **Ensembles across model families:** combine supervised classifier scores with anomaly detectors and graph scores via stacking or weighted blending.
- **Human-in-the-loop models:** ML suggests alerts; investigators provide feedback that is used to retrain and calibrate models (active learning).

5. Explainability and Regulatory Requirements

Regulatory compliance demands that decisions leading to SARs be auditable and explainable. For any ML model used in AML:

- **Feature-level explanations:** which features triggered the alert (e.g., “account sent three rapid outbound transfers to high-risk jurisdiction within 24 hours”)? Techniques: SHAP, LIME, counterfactual explanations.

- **Model provenance:** versioned models, training data snapshots, and system logs.
- **Confidence and uncertainty quantification:** present probability scores with calibrated uncertainty intervals to inform investigators.
- **Template generation for SARs:** extract evidence snippets (transactions, counterparties) and provide an investigator with a coherent narrative.

Explainability methods must be validated for stability and fidelity; poor explanations can mislead investigators and regulators.

6. Evaluation Metrics, Validation, and Backtesting

Evaluating AML systems requires careful design to reflect class imbalance and operational utility.

6.1 Offline evaluation metrics

- **Precision @ K:** proportion of true positives among top-K alarms aligns with investigator capacity.
- **Recall / detection rate:** fraction of known suspicious cases detected.
- **Area Under Precision-Recall Curve (AUPRC):** more informative than ROC AUC under severe imbalance.
- **Calibration metrics:** Brier score and reliability diagrams for probability outputs.
- **Cost-sensitive metrics:** weighted loss reflecting investigation cost and cost of missed detections (false negatives).

6.2 Backtesting and temporal validation

Use rolling origin evaluation with time windows to avoid leakage. Ensure models trained on data up to time ttt are tested on t+1,…t+1,\dotst+1,…; this captures concept drift and shifting typologies.

6.3 Operational evaluation

- **Alert reduction ratio:** how many alerts can be eliminated while preserving detection rate?
- **Time-to-detection:** latency from suspicious activity to alert.
- **Investigator productivity:** SARs filed per investigator per month; investigation throughput.
- **SAR quality metrics:** proportion of SARs accepted by regulator (where such feedback exists).

6.4 Controlled trials and pilots

Before enterprise rollout, run A/B tests or shadow mode pilots where ML alerts are scored but not acted upon, with investigator labeling. Use these trials to estimate real operational impacts and avoid adverse legal exposure.

7. System Design and Production Architecture

An operational AI-for-AML solution requires robust data engineering, ML lifecycle management, and human workflows.

7.1 Ingest and streaming layer

- Real-time ingestion of transaction streams with entity resolution and enrichment (sanctions check, geolocation).
- Use stream processing platforms (Kafka, Flink) for low latency.

7.2 Feature store and aggregation

- Time window aggregations (rolling counts, amounts) computed in streaming and materialized in feature store for both online inference and offline training.
- Maintain audits of feature provenance.

7.3 Model training and deployment

- Offline training pipelines: batched retraining cadence (daily/weekly) with retraining triggers on drift detection.
- Online serving: low-latency scoring for real-time alerts; batch scoring for periodic risk lists.
- Model governance: registries, CI/CD for models, automated validation tests (backtesting, fairness tests).

7.4 Case management integration

- Integrate ML alerts into case management systems (investigator UI), supporting triage, evidence display, and SAR drafting with prefilled fields.
- Investigator feedback loops to capture labels (false positive/true positive) and reingest into training pipeline.

7.5 Auditability and compliance

- Immutable logs of model inputs, outputs, model versions, and investigator actions for regulatory audits.
- Data retention and deletion policies per legal requirements.

8. Privacy, Security, and Ethical Considerations

AML systems process sensitive financial and personal data; privacy protection is essential.

- **Data minimization and encryption:** only store necessary attributes; encrypt in transit and at rest.
- **Access controls and role-based permissions** for investigator and model operations.
- **Differential privacy** techniques may be applied in federated learning contexts to share insights across institutions without sharing raw data.
- **Fairness and non-discrimination:** monitor for disparate impacts on protected groups (race, nationality) e.g., over-flagging certain demographic segments due to proxy features.
- **Adversarial risks:** actors may probe and manipulate detection systems; adopt adversarial training and anomaly robustness testing.

9. Adversarial and Evolutionary Threats

Criminals adapt. AML models must be robust to evasion:

- **Evasion strategies:** transaction splitting (smurfing), use of front companies, mixing services, rapid account churn.
- **Countermeasures:** detection of structuring patterns, graph motif surveillance, cross-product correlations, and signals from external intelligence (law enforcement, shared industry data).

Collaborative industry initiatives (information sharing) can improve detection of cross-institution laundering networks while respecting privacy and competitive concerns.

10. Case Studies and Example Workflows

10.1 Synthetic case: structuring detection with hybrid model

- **Feature set:** sliding window counts, amount variance, destination country risk, new payee flag, graph distance to sanctioned nodes.
- **Model:** ensemble of autoencoder (anomaly score), Random Forest (supervised), and GNN node embedding (network score).
- **Decision logic:** weighted combination yields final risk score; if above threshold, create an investigator case with top-3 contributing explanations (feature attributions) and key evidence (transactions, counterparties).

10.2 SAR automation pilot

- ML identifies high-probability cases; investigator reviews and uses prefilled SAR template with suggested narrative and supporting transaction list. Investigator edits and files SAR; outcome (filed/declined) stored as label for retraining.

11. Implementation Roadmap and Organizational Change

Adopting AI in AML is not purely technical; organizational readiness is key:

1. **Baseline assessment:** inventory of data, rules, investigator capacity, and compliance requirements.
2. **Pilot stage:** shadow mode pipelines, small-scale pilots focusing on specific product lines (e.g., wire transfers).
3. **Governance and policy:** assemble cross-functional team (compliance, legal, data science, security). Define KPIs and escalation procedures.
4. **Scaling:** roll out to additional products and channels; integrate with case management.
5. **Continuous improvement:** ongoing retraining, concept drift monitoring, collaboration with law enforcement and industry consortia for intelligence sharing.

Change management: training investigators on using ML outputs; transparency to compliance officers and regulators.

12. Research Challenges and Future Directions

Key areas for future research:

- **Label scarcity and transfer learning:** methods for few-shot learning and transfer across jurisdictions or product types.
- **Federated AML:** privacy-preserving cross-institution models to detect networked criminal activity without sharing raw data.
- **Causal inference in AML:** disentangling innocent correlated behavior from intentional laundering.
- **Explainable graph models:** improving interpretability of GNNs for regulator-grade explanations.
- **Benchmarks and public datasets:** creation of de-identified, realistic AML datasets for reproducible research (while preserving privacy).

13. Conclusion

AI offers significant opportunities to improve AML effectiveness by reducing false positives, surfacing complex networked patterns, and enabling prioritization of investigator attention. Careful engineering, rigorous evaluation, strong model governance, and attention to legal and ethical constraints are essential. Hybrid systems that combine interpretable rules, supervised learning, anomaly detection,

and graph models integrated with human expertise present the most practical path forward for financial institutions seeking to modernize AML while meeting regulatory obligations.

References

1. Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. *Statistical Science, 17*(3), 235–255.
2. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys, 41*(3), 1–58.
3. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794).
4. Fatunmbi, T. O. (2022). Quantum-accelerated intelligence in eCommerce: The role of AI, machine learning, and blockchain for scalable, secure digital trade. *International Journal of Artificial Intelligence & Machine Learning, 1*(1), 136–151. https://doi.org/10.34218/IJAIML_01_01_014
5. Fatunmbi, T. O. (2025). Quantum computing and artificial intelligence: Toward a new computational paradigm. *World Journal of Advanced Research and Reviews, 27*(1), 687–695. https://doi.org/10.30574/wjarr.2025.27.1.2498
6. Fawcett, T., & Provost, F. (1997). Adaptive fraud detection. *Data Mining and Knowledge Discovery, 1*(3), 291–316.
7. Fujimoto, S., Meger, D., & Precup, D. (2019). Off-policy deep reinforcement learning without exploration. In *Proceedings of ICML.*
8. Financial Action Task Force (FATF). (2013). *International standards on combating money laundering and the financing of terrorism & proliferation.* (Consolidated FATF recommendations).
9. Hawkes, A. G. (1971). Spectra of some self-exciting and mutually exciting point processes. *Biometrika, 58*(1), 83–90.
10. Hoffmann, H., & Mueller, D. (2019). Explainable AI for regulatory compliance: Towards transparency in financial decisioning. *Journal of Financial Regulation and Compliance, 27*(3), 403–421.
11. Kearns, M., Neel, S., & Roth, A. (2018). Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *Proceedings of the 35th International Conference on Machine Learning (ICML).*
12. Lewis, E., Mohler, G., Brantingham, P. J., & Bertozzi, A. L. (2012). Self-exciting point process models of civilian deaths in Iraq. *IEEE Transactions on Computational Social Systems, 1*(1), 1–11.

13. Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems, 30*, 4765–4774.

14. Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems, 50*(3), 559–569.

15. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144).

16. Rossi, R. A., & Ahmed, N. K. (2015). The network data repository with interactive graph analytics and visualization. AAAI, data resource referenced for network research.

17. Schmidt, N. M., & Haux, R. (2019). Suspicious activity detection in transaction networks: A survey of challenges and methods. *IEEE Transactions on Emerging Topics in Computational Intelligence, 3*(4), 248–260.

18. Wang, T., Wang, S., & Zhao, J. (2020). Graph neural network and transaction monitoring: Detecting anomalous activity in payment networks. *Journal of Financial Crime, 27*(4), 1123–1139.

19. Zhou, Y., & Kearns, M. (2015). Machine learning for risk and fraud detection in financial services: Methods and issues. *Journal of Risk Management, 8*(2), 75–88.