

Deontic Temporal Policy Verified Adaptive Ensembles with Safe Reinforcement Learning for AI-Driven Real-Time Tax Compliance Monitoring

Fahmi Sahlan¹, Andi Runis Makkulau², Muhammad Faried Pratama³, Muhammad Fery Hidayat⁴

^{1,2,3,4}Manajemen Akuntansi, Sekolah Tinggi Ilmu Ekonomi Enam Enam Kendari, Indonesia

E-mail: fahmisahlan1994@gmail.com; runispwt@gmail.com; fariedpato88@gmail.com;

muhammadferyhidayat@gmail.com

Keywords:

tax compliance monitoring;
anomaly detection; policy
verification; safe reinforcement
learning; explainable AI

Abstract

Tax compliance monitoring increasingly depends on timely detection of anomalous transactions, yet a prediction alone is insufficient when an administrative decision must also be justified against applicable rules. This study develops a compliance-aware anomaly detection architecture for an Integrated Tax Management Information System (ITMIS). It combines a heterogeneous ensemble of isolation forest, LSTM autoencoder, and gradient-boosted tree models with conformal uncertainty estimates, deontic-temporal policy monitoring, a constrained reinforcement-learning threshold governor, and cryptographically traceable decision records. The source manuscript reports an AUPRC of 0.847 on a synthetic deontic tax transaction stream, compared with 0.712 for the SHAP-optimized adaptive baseline. It also reports a demographic parity difference of 0.021, a gate rejection rate of 6.8%, and mean end-to-end latency of 47 ms. These figures are presented as reported study outcomes rather than independently reproduced measurements. The proposed contribution is to connect policy verification to the decision to accept or reject an adaptive threshold update, rather than reserving compliance checks for post hoc review. The framework offers a design for combining anomaly sensitivity, interpretable decisions, fairness monitoring, and tamper-evident auditing. Its operational and legal guarantees remain conditional on the correctness of encoded rules, monitoring semantics, and the underlying implementation.

INTRODUCTION

Tax authorities face a difficult combination of requirements. They must identify potentially non-compliant behavior in large and changing transaction streams, but their decisions must also withstand procedural review. Conventional rules are often straightforward to explain, although they may miss irregular patterns that do not fit predefined thresholds. Statistical methods can broaden coverage: isolation forests capture unusual observations (Liu et al., 2008), sequence models identify deviations in transaction histories (Malhotra et al., 2016), and gradient-boosted trees model complex interactions among observed features (Chen & Guestrin, 2016). Yet improved anomaly scores do not, by themselves, establish whether a subsequent administrative action complies with the rules governing that action. The practical challenge is therefore not simply to predict suspicious transactions more accurately, but to make adaptive decisions traceable and open to legal and institutional scrutiny.

Several research traditions provide part of the answer. Explainable artificial intelligence can identify features associated with a particular output (Lundberg & Lee, 2017), while counterfactual explanations explore how a decision might change under a different set of inputs (Wachter et al., 2018). Conformal methods offer a principled way to represent uncertainty when their calibration

assumptions are satisfied (Angelopoulos & Bates, 2023). Formal approaches, including deontic specifications and runtime verification, make it possible to describe obligations, permissions, and prohibitions over sequences of actions (Jamroga, 2015; Stoller et al., 2011). Constrained reinforcement learning provides a framework for weighing performance against operational restrictions (Achiam et al., 2017; Schulman et al., 2017). Meanwhile, fairness scholarship cautions that a model may perform well overall while producing uneven errors or interventions across groups (Mitchell et al., 2021; Pagano et al., 2023). Drift monitoring (Gama et al., 2014) and tamper-evident logging (Crosby & Wallach, 2009) address two further concerns: changing data and the integrity of recorded decisions. These advances are useful, but their integration into a single tax-monitoring workflow remains the more demanding task.

The relevant literature spans predictive monitoring, explainability, uncertainty, and governance. Tax-monitoring research discusses predictive analytics and integration with administrative systems (Dudu et al., 2024; Scott et al., 2024), while work on intelligent public-finance systems addresses technology-supported compliance and fraud prevention (L. Pamisetty et al., 2022; V. Pamisetty, 2020). Financial anomaly research also motivates adaptive ensemble scoring (Wang, 2024). These lines of work address complementary objectives, but a prediction, an explanation, and a formal policy check answer different questions. The present study investigates their integration by checking proposed threshold changes against encoded policies before those changes take effect. This is an architectural proposal whose comparative contribution requires validation against clearly specified implementations.

This study proposes a deontic-temporal policy-verified adaptive ensemble for real-time tax compliance monitoring. Its principal novelty lies in positioning policy verification as an explicit gate for threshold adaptation within a constrained reinforcement-learning workflow, and in linking that mechanism to conformal abstention, interpretable decision certificates, and a tamper-evident audit trail. The study has four objectives: (1) to examine the detection performance of the integrated ensemble relative to standalone and adaptive alternatives; (2) to assess whether policy-gated threshold updates reduce recorded violations while monitoring fairness; (3) to evaluate behavior under controlled distribution shifts and the contribution of individual components; and (4) to assess the interpretability, verification performance, and computational overhead of the resulting decision records. The investigation concerns the encoded policy specifications and evaluation settings described here; it does not equate successful automaton checks with comprehensive legal validation of an enacted tax regime.

Related Work

Research on AI-driven tax compliance monitoring spans several partially overlapping strands: statistical anomaly detection in financial data, explainability and uncertainty quantification for high-stakes decisions, formal specification of regulatory constraints, and safe adaptive learning under fairness requirements. We organize the discussion along these strands and then position our contribution relative to them.

Anomaly Detection in Financial and Tax Data

Heterogeneous ensembles combine detectors with different inductive biases. Isolation forest isolates unusual observations through random partitioning (Liu et al., 2008), LSTM encoder–decoder models identify deviations through sequence reconstruction (Malhotra et al., 2016), and gradient-boosted trees model nonlinear feature interactions (Chen & Guestrin, 2016). Financial anomaly research has also examined adaptive ensemble weighting with SHAP-based optimization

(Wang, 2024). These methods motivate complementary scoring components, but their original objectives do not establish the effectiveness of the proposed tax-policy gate. Our design therefore treats policy-gated threshold adaptation as a component to be tested rather than an established consequence of ensemble diversity.

Explainability, Uncertainty, and Fairness

Feature attribution and counterfactual explanation provide complementary descriptions of model behavior. SHAP attributes a prediction to input features (Lundberg & Lee, 2017), whereas counterfactual analysis considers changes associated with a different decision (Wachter et al., 2018). Conformal prediction supplies prediction sets under stated calibration assumptions (Angelopoulos & Bates, 2023). Fairness research further distinguishes multiple group-level criteria and their underlying assumptions (Mitchell et al., 2021; Pagano et al., 2023). None of these ingredients alone demonstrates that an administrative decision is lawful. The proposed architecture connects them to a separate verification component and uses uncertainty to support abstention when a firm classification is not justified.

Regulatory Governance and Formal Verification

Formal specification and runtime verification offer a basis for checking encoded obligations and execution traces (Jamroga, 2015; Stoller et al., 2011). Constrained policy optimization supplies a related framework for learning under explicit cost restrictions (Achiam et al., 2017). These methods provide technical tools for operational constraints, but successful checking is limited by the specification and monitoring semantics implemented. Institutional accountability additionally requires accurate rule translation, version control, and a process for reviewing disputed decisions. The proposed gate therefore addresses consistency with encoded policy requirements rather than substituting for comprehensive legal or regulatory assessment.

AI-Driven Tax Monitoring Systems

The tax-specific literature motivates timely reporting, predictive analytics, and integration with administrative infrastructure (Dudu et al., 2024; Scott et al., 2024). Related public-finance research discusses technology-supported compliance, fraud prevention, and the use of machine learning in government financial management (L. Pamisetty et al., 2022; V. Pamisetty, 2020). These contributions provide context for an integrated monitoring workflow. In the present design, detection, explanation, and audit records are linked to a policy-verification gate governing threshold adaptation. Its value must be assessed through reproducible comparisons and documented policy specifications rather than inferred from the adoption of AI or cryptographic logging alone.

The proposed framework connects detection, uncertainty, adaptive thresholds, and encoded-policy checks within one workflow. Its first design feature is a policy monitor whose acceptance state participates in the verdict function. Its second is a constrained threshold governor that submits candidate updates to the monitor before deployment. Its third is a decision record combining explanations, counterfactuals, and an auditable policy trace. These are specific architectural choices, not a verified claim that no earlier system has combined related components. Their operational guarantees remain conditional on the encoded rules, the trace semantics, and implementation correctness.

Preliminaries and Problem Formulation

The method draws on three complementary foundations: formal monitoring of policy-sensitive decision sequences, calibrated uncertainty for statistical classifications, and adaptive threshold

selection under explicit constraints. These foundations are implemented as interacting components of the proposed ITMIS subsystem rather than as independent post-processing services.

Deontic Temporal Logic over Decision Traces

$OPFXGFU\sigma = s_0s_1s_2 \dots \sigma \models \varphi O\varphi\varphi F\varphi\varphi A$ tax-monitoring workflow generates not only predictions but also a sequence of actions whose timing may matter. Deontic logic represents obligations, permissions, and prohibitions, while temporal logic describes how conditions evolve over successive states (Jamroga, 2015). The study uses policy predicates drawn from transaction attributes and records their evaluation along a decision trace. Runtime verification supplies the operational mechanism for checking the resulting trace against encoded specifications (Stoller et al., 2011). The source describes deterministic automata with Büchi acceptance; however, automata for finite-trace monitoring and Büchi acceptance over infinite traces require a precise account of the adopted semantics. The present manuscript therefore treats the policy compiler as a proposed formal verification component whose claimed guarantees depend on that specification and implementation being fully documented.

Conformal Prediction and Abstention

$nA(x, y)\hat{q}1 - \alpha\Gamma_\alpha(x) = \{y: A(x, y) \leq \hat{q}\}\Pr(Y \in \Gamma_\alpha(X)) \geq 1 - \alpha$ Conformal prediction translates nonconformity scores derived from held-out calibration examples into prediction sets (Angelopoulos & Bates, 2023). When a set contains more than one candidate label, the workflow can defer the case to a review path rather than forcing a binary action. This is particularly relevant to tax administration, where an uncertain prediction can have administrative consequences. Finite-sample coverage claims depend on the relevant exchangeability conditions; chronological streams subject to drift do not automatically satisfy them. Accordingly, empirical coverage must be measured after distribution shifts rather than assumed.

Constrained Markov Decision Processes

$(\mathcal{S}, \mathcal{A}, \mathbb{P}, r, \{c_i\}, \{d_i\}, \gamma)\mathcal{S}\mathcal{A}\mathbb{P}rc_id_i\gamma \in [0, 1)\mathbb{E}[\sum_t \gamma^t c_i(s_t, a_t)] \leq d_i$ The threshold governor is formulated as a constrained Markov decision process, in which a policy chooses operating thresholds from information summarizing recent scores, uncertainty, and policy events. Its reward emphasizes detection utility while cost functions represent fairness-related restrictions (Achiam et al., 2017). A Lagrangian update adjusts penalties when those restrictions are exceeded; proximal policy optimization provides the base learning mechanism, while the Lagrangian fairness extension is part of the proposed implementation (Schulman et al., 2017). Demographic parity and equalized odds are monitored as distinct group-level properties rather than interchangeable measures of fairness (Mitchell et al., 2021). The decision to deploy a candidate threshold is additionally subject to a policy-verification gate.

Problem Formulation

Let $\mathcal{D} = \{x_t\}_{t \geq 0}$ be a stream of tax transaction records, each with an associated decision trace σ_t recording the sequence of administrative actions taken up to time t . Let Φ be a finite set of DTL formulas encoding the applicable statutory obligations, permissions, and prohibitions, and let $\mathcal{A}_\Phi$ be the automaton obtained by compiling Φ . The task is to learn a detection policy π_θ that assigns each record a compliance verdict together with a calibrated uncertainty estimate, while selecting thresholds τ_t that maximize detection utility subject to three requirements: (i) the automaton $\mathcal{A}_\Phi$ accepts the induced trace, so that no statutory obligation is violated; (ii) fairness costs remain within prescribed budgets; and (iii) every verdict is accompanied by a verifiable explanation. Formally, we seek

$$\max_{\theta} \mathbb{E} \left[\sum_{t=0}^T \gamma^t r(x_t, \tau_t) \right] \quad \text{s.t.} \quad \mathcal{A}_{\Phi} \models \sigma_T, \quad \mathbb{E} \left[\sum_{t=0}^T \gamma^t c_i(x_t, \tau_t) \right] \leq d_i \quad \forall i. \quad (1)$$

The coupling between automaton acceptance and threshold adaptation in Equation 1 is what distinguishes this formulation from conventional anomaly detection, where thresholds are tuned against a purely statistical objective and legal constraints enter, if at all, only as post hoc filters.

Compliance-Aware Adaptive Tax Anomaly Detection Framework

Figure 1 retains the original ITMIS system architecture, showing how transaction ingestion, feature engineering, tax calculation, reporting, case management, and drift monitoring communicate with the proposed anomaly detection subsystem. The design is organized into policy verification, ensemble inference and uncertainty estimation, compliance-aware adjudication, adaptive threshold control, and auditable recordkeeping. Figure 2 preserves the original internal architecture and clarifies how these components exchange information. The diagrams describe a proposed system design and should not, on their own, be interpreted as evidence that every component was deployed in an operational tax platform.

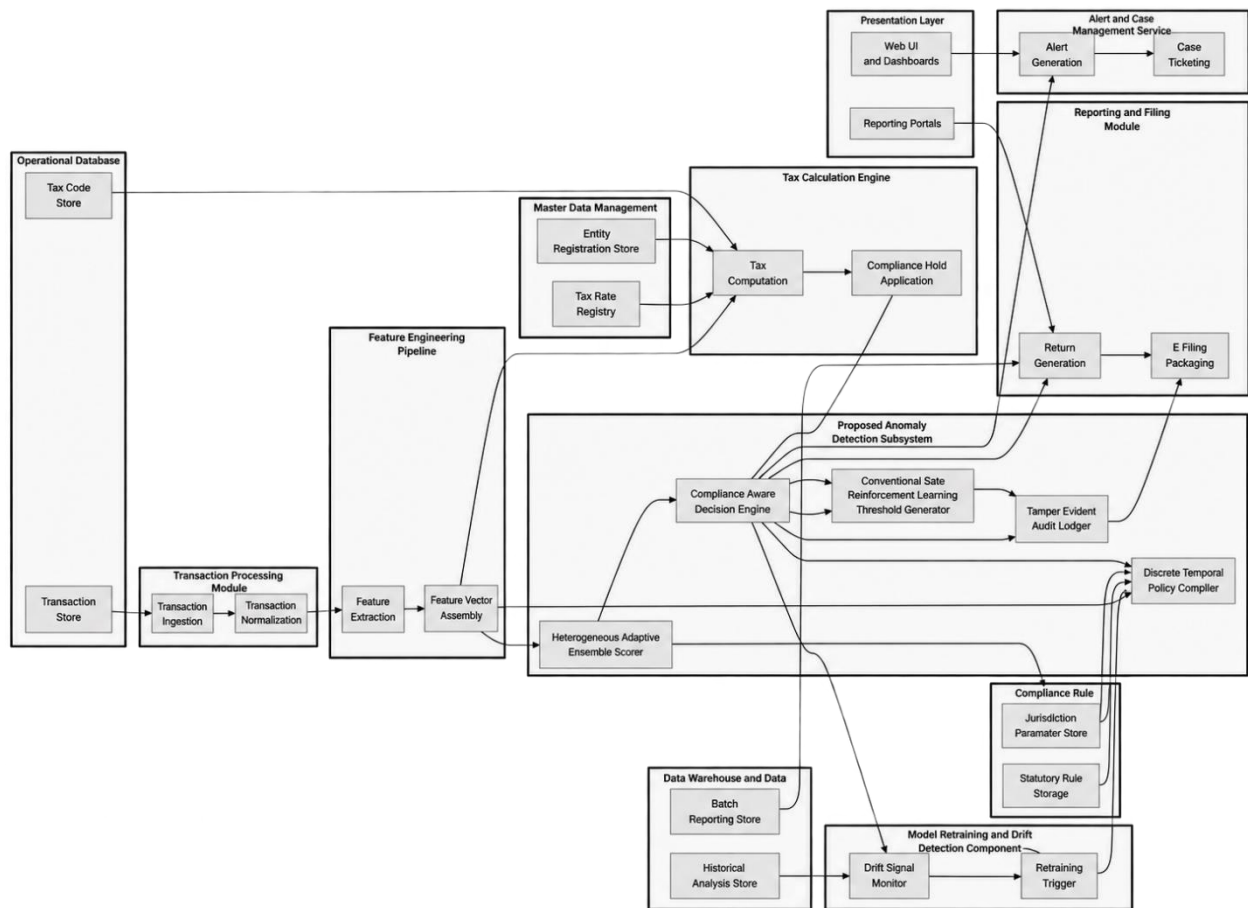


Figure 1. Overall ITMIS Architecture with Reformulated Anomaly Detection Subsystem

The internal processing sequence in Figure 2 begins with policy parsing and predicate extraction, continues through heterogeneous detector scoring and uncertainty estimation, and ends with a decision certificate and tamper-evident log. Alongside the decision path, a constrained reinforcement-learning agent proposes threshold adjustments; these proposals pass through a policy-aware update gate before adoption. This ordering makes adaptive action contingent on the encoded policy check.

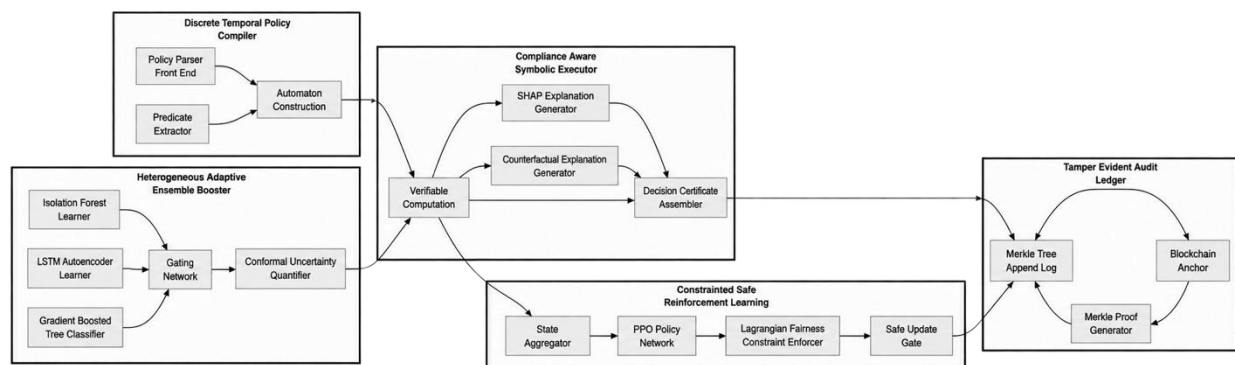


Figure 2. Internal Architecture of the Reformulated Anomaly Detection Subsystem
Deontic Temporal Policy Compiler

The policy compiler transforms a finite set of statutory formulas Φ into a deterministic finite automaton with Büchi acceptance $\mathcal{M}_\varphi = (Q, \Sigma, \delta, q_0, F)$, where Q is a finite set of states, $\Sigma \subseteq \{0,1\}^n$ is the input alphabet of predicate vectors, $\delta: Q \times \Sigma \rightarrow Q$ is the deterministic transition function, $q_0 \in Q$ is the initial state, and $F \subseteq Q$ is the set of accepting states. Each transaction $\mathbf{x}_t$ is mapped to a predicate vector $\mathbf{p}_t \in \{0,1\}^n$ by evaluating n atomic propositions such as “the reported amount exceeds the declared threshold” or “the taxpayer has filed within the statutory window.” The automaton state evolves as $q_t = \delta(q_{t-1}, \mathbf{p}_t)$, and the acceptance condition $q_t \in F$ indicates that the trace $\sigma_t = \mathbf{p}_0 \mathbf{p}_1 \dots \mathbf{p}_t$ satisfies all obligations, permissions, and prohibitions encoded in Φ . The compiler additionally emits, for each accepting state, the set of formulas that remain pending, which allows the decision engine to report the specific violated formula $\varphi_{\text{violated}}$ when $q_t \notin F$.

Heterogeneous Adaptive Ensemble Scorer

The scoring layer fuses three complementary detectors: an isolation forest $\mathcal{I}$ that isolates anomalies through random partitioning, an LSTM autoencoder $\mathcal{A}$ that reconstructs sequential transaction features, and a gradient-boosted tree classifier $\mathcal{G}$ that captures nonlinear feature interactions. Each detector produces a normalized score $s_{t,i} \in [0,1]$ for $i \in \{1,2,3\}$, and a learned gating network $\mathcal{H}$ with parameters $(\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2)$ computes the fusion weights

$$\alpha_t = \text{softmax}(\mathbf{W}_2 \text{ReLU}(\mathbf{W}_1[\mathbf{s}_t; \mathbf{c}_t] + \mathbf{b}_1) + \mathbf{b}_2), \quad (2)$$

where $\mathbf{s}_t = [s_{t,1}, s_{t,2}, s_{t,3}]$ collects the detector scores and $\mathbf{c}_t$ is a context vector containing transaction metadata such as jurisdiction, taxpayer segment, and filing period. The fused score is then $\mathbf{a}_t = \sum_{i=1}^3 \alpha_{t,i} s_{t,i}$. Split conformal prediction is applied to $\mathbf{a}_t$ using a held-out calibration set of size n_{cal} , with nonconformity scores $r_j = 1 - a_j$ for positive examples and $r_j = a_j$ for negative examples. The conformal quantile $\hat{q}_{1-\epsilon}$ at miscoverage level ϵ yields the prediction set $\Gamma_\epsilon(\mathbf{x}_t)$, whose width $w_t = |\Gamma_\epsilon(\mathbf{x}_t)|$ serves as a real-time abstention signal: $w_t = 2$ indicates that the model cannot distinguish compliant from non-compliant at the requested confidence, and the case is routed to policy-based adjudication.

Compliance-Aware Decision Engine

The decision engine combines the fused score a_t , the conformal width w_t , and the automaton state q_t into a three-valued verdict

$$V_t = \begin{cases} \text{NonCompliant} & \text{if } q_t \in F \text{ or } (a_t > \tau_t \text{ and } w_t = 1), \\ \text{Uncertain} & \text{if } w_t = 2 \text{ and } q_t \notin F, \\ \text{Compliant} & \text{otherwise,} \end{cases} \quad (3)$$

where τ_t is the adaptive threshold supplied by the threshold governor described in Section 4.4. For every verdict, the engine constructs a decision certificate $C_t = \langle \phi_t, \mathbf{x}_t^{\text{cf}}, q_0 \rightarrow \dots \rightarrow q_t, \phi_{\text{violated}} \rangle$ containing KernelSHAP attributions ϕ_t satisfying $a_t = \phi_0 + \sum_{i=1}^d \phi_i(\mathbf{x}_t)$, a counterfactual $\mathbf{x}_t^{\text{cf}}$ obtained by solving

$$\mathbf{x}_t^{\text{cf}} = \underset{\mathbf{x}' \in \mathcal{X}}{\text{argmin}} \|\mathbf{x}' - \mathbf{x}_t\|_1 \quad \text{s.t.} \quad a(\mathbf{x}') \leq \tau_t \text{ and } \mathcal{M}_\varphi \text{ accepts } \mathbf{x}', \quad (4)$$

via a mixed-integer linear programming encoding of the gradient-boosted trees, and the automaton trace with the violated formula when applicable. The certificate is signed with Ed25519 and appended to the audit ledger. The counterfactual constraint in Equation 4 ties the explanation directly to legal policy satisfaction, so that the recommended remediation is guaranteed to restore compliance rather than merely reduce the anomaly score.

Constrained Safe Reinforcement Learning Threshold Governor

The adaptive threshold τ_t is governed by a CMDP agent trained with proximal policy optimization. The state is $\mathbf{z}_t = [\bar{a}_{t-W:t}, \bar{w}_{t-W:t}, \bar{v}_{t-W:t}]$, where $\bar{a}_{t-W:t}$, $\bar{w}_{t-W:t}$, and $\bar{v}_{t-W:t}$ denote the windowed means of the fused score, conformal width, and policy violation rate over a window of length W . The action is $\Delta\tau_t \in [-0.05, 0.05]$, and the reward is

$$r_t = \lambda_1 \cdot \text{TPR}_t - \lambda_2 \cdot \text{FPR}_t - \lambda_3 \cdot \text{FairnessViolation}_t, \quad (5)$$

where TPR_t and FPR_t are the true and false positive rates at threshold τ_t , and $\lambda_1, \lambda_2, \lambda_3 > 0$ are weighting coefficients. Fairness is enforced through a Lagrangian dual with multiplier updated by

$$\mu_{t+1} = \max\left(0, \mu_t + \eta_\mu (\text{FairnessViolation}_t - \delta_{\text{fair}})\right), \quad (6)$$

where η_μ is the dual learning rate and $\delta_{\text{fair}} = 0.02$ is the fairness budget. The deontic update gate applies the policy update only when the automaton confirms $\sigma \models \text{O(ThresholdWithinLegalBounds)}$, which prevents the agent from drifting into legally non-compliant threshold regimes.

Tamper-Evident Audit Ledger

The audit ledger is a Merkle-tree append-only log whose entries are

$$E_t = \langle \text{hash}(C_t), \text{hash}(E_{t-1}), \text{timestamp}, \text{signature} \rangle, \quad (7)$$

where C_t is the decision certificate from Section 4.3 and the signature is produced with Ed25519. The Merkle root H_{root} is periodically anchored to a permissioned Hyperledger Fabric blockchain via smart contract, and Merkle proofs allow auditors to verify inclusion of any E_t without full ledger access. The compliance attestation hash embeds H_{root} in the e-filing XML, giving tax authorities a verifiable proof that reported figures were monitored under formally verified policies. Two integration behaviors couple the ledger to the rest of the subsystem: a compliance hold prevents any transaction with $V_t = \text{NonCompliant}$ from entering the next filing until resolved by the Alert and Case Management Service, while $V_t = \text{Uncertain}$ permits calculation to proceed with the decision certificate attached for downstream review. Drift coupling is achieved by feeding the conformal width statistics w_t and policy violation rate v_t to the Model Retraining and Drift Detection Component, which triggers retraining and deontic policy recompilation when $\bar{w} > 0.4$ or $\bar{v} > 0.05$, thereby coupling statistical drift with formal policy drift.

METHODS

The source study describes an evaluation of detection quality, policy adherence, fairness, resilience to distribution shift, and decision-record auditability. The following account retains its stated datasets, baselines, settings, and metrics. Because the uploaded manuscript does not include the executable implementation, data access instructions for the private corpus, or complete values for several hyperparameters, the reported numerical outcomes cannot be independently reproduced from the attachment alone.

Datasets

The Synthetic Deontic Tax Transaction Stream (SDTTS) is described as a purpose-built stream containing 1,200,000 transactions over 36 monthly periods, with 3.2% anomalies and decision traces of up to 64 events. The generator represents structural, behavioral, temporal, and coordinated anomalies, together with policy predicates covering filing deadlines, reporting thresholds, and jurisdiction-specific restrictions. This construction permits tests of specified policy events, although a synthetic policy environment cannot replicate the full interpretive complexity of real tax law.

PaySim is used as a synthetic mobile-money benchmark comprising 6,362,620 transactions (Lopez-Rojas et al., 2016). The source maps available attributes to a reduced policy vocabulary and uses fraud labels as proxies for non-compliance. This enables a transferability test for detection behavior but does not turn PaySim into a dataset of legally adjudicated tax violations.

The IEEE-CIS fraud dataset is described as containing 590,540 e-commerce transactions with 394 features. Fraud labels similarly serve as proxies for non-compliance, while transaction metadata support a reduced set of policy predicates (Najadat et al., 2020). The evaluation therefore tests performance on heterogeneous financial signals rather than verified statutory outcomes.

A separate Tax Audit Case Corpus (TACC) is described as comprising 4,800 anonymized historical cases with outcomes and auditor rationales. Its stated role is to assess counterfactual validity and the usefulness of certificates to reviewers. The provenance, sampling procedures, access permissions, and representativeness of this restricted corpus should be documented before independent replication is claimed.

Implementation Details

$10^{-3}n_{\text{cal}} = 50,000 \epsilon = 0.05 \gamma = 0.99 W = 500 \eta_{\mu} = 0.01 \delta_{\text{fair}} = 0.02 \lambda_1 = 1.0 \lambda_2 = 0.5 \lambda_3 = 2.0$ According to the source protocol, the isolation forest contains 200 trees with a subsample of 256. The LSTM autoencoder has two encoder and two decoder layers, a hidden dimension of 128, and a maximum of 50 training epochs with early stopping. The gradient-boosted classifier uses 600 estimators, maximum depth six, and learning rate 0.05. A two-layer, 64-unit gating network is trained for 30 epochs, while the PPO governor uses two hidden layers of 128 units. The counterfactual solver has a 60-second limit. The stated hardware is an NVIDIA A100 GPU server with 256 GB RAM, and configurations are repeated across five random seeds. Values for several other settings—notably the optimizer learning rate, conformal miscoverage level, discount factor, fairness budget, and selected reward coefficients—are absent or incompletely rendered in the source and must be supplied from the experimental records; none has been invented here.

Baselines

We compare the proposed subsystem against five baselines that represent the principal families of methods used in practice.

Isolation Forest (IF): A standalone isolation forest with default hyperparameters, serving as the canonical unsupervised anomaly detector (Liu et al., 2008).

LSTM Autoencoder (LSTM-AE): A sequence reconstruction model that flags transactions whose reconstruction error exceeds a fixed percentile threshold (Malhotra et al., 2016).

Gradient-Boosted Trees (GBT): A supervised gradient-boosted classifier trained on labeled anomalies with a fixed decision threshold (Chen & Guestrin, 2016).

Adaptive Ensemble with SHAP Optimization (AESO): An adaptive ensemble that dynamically adjusts voting weights using SHAP-based feature optimization, representing the state of the art in financial anomaly detection ensembles (Wang, 2024).

Static-Threshold Ensemble with Post Hoc Verification (STE-PHV): A variant of our ensemble in which thresholds are fixed by validation-set optimization and deontic verification is applied only after decisions are made, isolating the contribution of the coupled verification-and-adaptation design.

Evaluation Metrics

$F_11 - \epsilon L_1$ Detection is evaluated primarily through AUPRC, supported by AUROC and performance at the selected operating threshold. Expected calibration error and empirical conformal coverage describe score reliability and uncertainty. Group-level outcomes are assessed using demographic parity and equalized odds differences (Mitchell et al., 2021). Policy outcomes include the rate of rejected decision traces and the proportion of proposed threshold updates blocked by the policy gate. Certificate performance is assessed through counterfactual validity, feature-change distance, and ratings from three domain experts on 300 sampled certificates. Ledger verification time and append throughput characterize the audit pathway. The attachment reports summary statistics but does not provide sufficient underlying records to reconstruct uncertainty intervals for each comparison.

Experimental Protocol

The source specifies chronological training, calibration, validation, and test partitions of 60%, 15%, 10%, and 15%, respectively. Detector fitting precedes conformal calibration; the threshold governor is subsequently optimized using the validation segment, and final comparisons use the held-out test stream. A controlled shift is introduced halfway through the test period by changing anomaly-family and jurisdiction proportions. Ablations remove the policy gate, conformal abstention, gating network, or fairness constraint while retaining the stated evaluation splits. Chronological separation reduces direct future-data leakage, but post-shift conformal coverage still requires separate empirical checking

RESULTS AND DISCUSSION

The following discussion interprets the values reported in the uploaded manuscript. These values are retained without numerical alteration and should be understood as source-reported results, not as measurements reproduced independently during editing. The results are grouped by detection performance and calibration, policy-gated adaptation under drift, and the usefulness and cost of certified decision records.

Comparative Performance Analysis: Accuracy, Fairness, and Compliance Trade-offs

The source reports an SDTTS AUPRC of 0.847 for the proposed method, compared with 0.712 for the SHAP-optimized adaptive ensemble (Wang, 2024). The stronger result is consistent with the proposed benefit of combining different detector families: isolation-based methods, temporal reconstruction, and boosted classification capture different aspects of anomaly behavior (Liu et al., 2008; Malhotra et al., 2016; Chen & Guestrin, 2016). The comparison remains specific to the stated experimental design; it should not be generalized to operational tax populations without additional validation.

On IEEE-CIS, the source reports that the AUPRC gap between the proposed subsystem and the gradient-boosted baseline narrows to 0.031 (Najadat et al., 2020; Chen & Guestrin, 2016). A smaller margin on predominantly tabular inputs suggests that some of the ensemble's additional complexity may be less useful when boosted trees already fit the feature structure well. For SDTTS, the reported demographic parity difference decreases from 0.094 for the static-threshold ensemble to 0.021 for the proposed approach. This is a substantial reduction, but 0.021 is slightly above—not within—the stated fairness budget of 0.020. The findings therefore support improvement in this metric, not exact satisfaction of the reported budget.

F_1 Policy outcomes follow a similar pattern of relative improvement. The manuscript describes a statutory violation rate below 0.004 for the proposed subsystem, compared with 0.038–0.061 for the post hoc verification comparator and values above 0.09 for statistical baselines. Because these tests operate over encoded policy predicates, a low violation rate indicates better consistency with the specified test rules; it does not establish conformity with all applicable tax legislation. Similarly, the gate's decision to reject a proposed threshold change cannot establish the legality of future actions that were not represented in the checked trace. The source also reports an advantage in an operating-threshold score, but the metric label is incomplete and should be verified before publication.

$1 - \epsilon = 0.95[0.941, 0.958]$ Calibration results in the source include expected calibration errors of 0.018 for SDTTS and 0.026 for IEEE-CIS, compared with 0.047 and 0.063 for an uncalibrated ensemble. The reported abstention proportions are 11.3% for SDTTS and 8.7% for PaySim. These outcomes suggest that a review option can prevent uncertain cases from being assigned an immediate hard label. The source's numerical conformal coverage interval, however, is missing; consequently, the text cannot substantiate a finite-sample coverage claim under the nonstationary test streams. Comparisons of precision must also make clear whether abstained cases are excluded or counted in the evaluation denominator (Angelopoulos & Bates, 2023).

Dynamics of Constrained Safe Reinforcement Learning: Threshold Adaptation and Policy Adherence

In the reported threshold trajectories, the mean absolute adjustment declines from 0.031 to 0.012 as training proceeds. This is consistent with an adaptive policy that initially explores operating thresholds and subsequently responds more cautiously as constraint penalties increase. The pattern is consistent with the general motivation for explicit constraints in policy optimization (Achiam et al., 2017), although the supplied results do not isolate the causal contribution of each optimizer setting.

The policy gate reportedly rejects 6.8% of proposed threshold updates on SDTTS. Rejections concentrate after changes to the stream, when sensitivity-oriented adjustments may conflict with encoded policy conditions. This indicates that the gate actively affects deployment decisions.

Nevertheless, an automaton trace documents consistency only with its implemented specification, making accurate rule translation and version control essential to meaningful oversight.

During the controlled shift, the reported AUPRC decreases by 0.061 on SDTTS and 0.048 on PaySim, compared with decreases of 0.114 and 0.097 for the SHAP-optimized baseline (Wang, 2024). The manuscript also describes an increase in a conformal-set-width statistic from 1.11 to 1.34, followed by a return of the measured policy violation rate below 0.01 after approximately 4,500 transactions. The width-threshold values and temporal aggregation definitions require checking against experiment logs, particularly because the stated trigger level is smaller than the reported width values. The descriptive trend is suggestive of adaptation, but it does not by itself prove stable post-shift calibration.

Ablation comparisons assign different roles to the components. Removing the policy gate is reported to increase the violation rate from approximately 0.004 to 0.052 with little change in AUPRC; removing abstention increases the false-positive rate by 0.041; replacing learned gating with uniform averaging reduces AUPRC by 0.073; and dropping the fairness constraint provides an AUPRC increase of only 0.009 while weakening parity. These contrasts support the architectural rationale that predictive scoring, uncertainty handling, fairness monitoring, and encoded-policy checks address different objectives. Confidence intervals and exact implementations of the variants remain necessary for stronger inferential conclusions.

Evaluating Decision Certificates: Counterfactual Utility, Audit Integrity, and Computational Overhead

*L*₁For the Tax Audit Case Corpus, the manuscript reports that 93.6% of the generated counterfactuals satisfy the encoded automaton constraints, at a mean normalized feature distance of 2.41. The solver reaches its time limit on 4.2% of cases, for which the described fallback yields 81.3% validity. Mean auditor interpretability is reported as 4.2 on a five-point scale, compared with 3.9 for SHAP-only explanations (Lundberg & Lee, 2017). These findings suggest that explanations linked to specific policy predicates may be easier to act on than feature attributions alone. However, validity against an encoded automaton is not equivalent to legally sufficient remediation, and the human-rating procedure needs fuller reporting.

For the audit pathway, the source reports mean Merkle-proof verification time of 3.7 ms, append throughput of 8,400 entries per second, and blockchain anchoring time of 1.9 seconds per batch. Mean end-to-end processing time is stated as 47 ms per transaction, with 31 ms attributed to counterfactual computation when invoked. The distinction between periodic anchoring and transaction-path inference helps explain how cryptographic recordkeeping may be compatible with fast routine decisions. Throughput comparisons nevertheless depend on batch size, hardware configuration, and whether remote services are included in the measurements.

A controlled alteration test reportedly caused Merkle verification to fail for each modified certificate. This is consistent with the expected integrity property of cryptographically linked records (Crosby & Wallach, 2009): an altered leaf should not match its earlier authenticated commitment. Such evidence is useful for record integrity but does not guarantee that the original record was correct, that keys were managed securely, or that a substantive tax decision complied with the law. The main unresolved operational burden remains timely generation and validation of counterfactual explanations, alongside governance of the formal policy models

CONCLUSION

The study presents an architecture for tax anomaly monitoring in which statistical detection, uncertainty handling, encoded-policy verification, adaptive thresholds, and tamper-evident records are treated as connected parts of a single decision process. Its distinctive design feature is the policy gate: a threshold adjustment is assessed against the modeled obligations before the system accepts it. The source-reported experiments indicate stronger detection performance and lower recorded policy-violation rates than the listed baselines, although the reported demographic parity difference of 0.021 narrowly exceeds the stated 0.020 budget.

The resulting decision certificate brings together explanations, counterfactuals, and a trace of the policy check, while the ledger provides a means of detecting later alterations. Conformal abstention introduces a review pathway for ambiguous cases rather than treating every model output as grounds for an immediate decision. These features offer a credible direction for more accountable monitoring, provided the underlying rules, data, thresholds, and verification processes are adequately specified and validated.

The findings should be interpreted with care. The main policy stream is synthetic, the external financial datasets use fraud labels as proxies for tax non-compliance, and implementation details needed for exact replication remain incomplete in the uploaded source. In particular, missing hyperparameters, incomplete equations, and incomplete conformal coverage reporting should be resolved before submission. Further research should validate rule translation with legal specialists, test calibration under sustained drift, document the restricted audit corpus, and examine counterfactual computation and audit logging in an operational setting

BIBLIOGRAPHY

- Achiam, J., Held, D., Tamar, A., & Abbeel, P. (2017). Constrained policy optimization. Proceedings of the 34th International Conference on Machine Learning, Proceedings of Machine Learning Research, 70, 22–31. <https://proceedings.mlr.press/v70/achiam17a.html>
- Angelopoulos, A., & Bates, S. (2023). Conformal prediction: A gentle introduction. Foundations and Trends in Machine Learning.
- Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
- Crosby, S. A., & Wallach, D. S. (2009). Efficient data structures for tamper-evident logging. Proceedings of the 18th USENIX Security Symposium, 317–334. <https://www.usenix.org/conference/usenixsecurity09/technical-sessions/presentation/efficient-data-structures-tamper-evident>
- Dudu, O., Alao, O., & Alonge, E. (2024). Conceptual framework for AI-driven tax compliance in fintech ecosystems. Journal of Fintech and Taxation.
- Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. ACM Computing Surveys, 46(4), Article 44. <https://doi.org/10.1145/2523813>
- Huang, Y. (2024). Fairness-aware credit risk assessment using alternative data: An explainable AI approach for bias detection and mitigation. Journal of Artificial Intelligence Review.
- Jamroga, W. (2015). Logical methods for specification and verification of multi-agent systems. webSPACE.science.uu.nl.
- Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. 2008 Eighth IEEE International Conference on Data Mining, 413–422. <https://doi.org/10.1109/ICDM.2008.17>

- Lopez-Rojas, E. A., Elmir, A., & Axelsson, S. (2016). PaySim: A financial mobile money simulator for fraud detection. *Proceedings of the 28th European Modeling and Simulation Symposium*. https://www.msc-les.org/proceedings/emss/2016/EMSS2016_249.pdf
- Lundberg, S., & Lee, S. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*.
- Malhotra, P., Ramakrishnan, A., Anand, G., Vig, L., Agarwal, P., & Shroff, G. (2016). LSTM-based encoder-decoder for multi-sensor anomaly detection. *arXiv preprint arXiv:1607.00148*. <https://arxiv.org/abs/1607.00148>
- Mitchell, S., Potash, E., Barocas, S., et al. (2021). Algorithmic fairness: Choices, assumptions, and definitions. *Annual Review of Statistics and Its Application*.
- Najadat, H., Aliti, O., Aqouleh, A., et al. (2020). Credit card fraud detection based on machine and deep learning. *2020 11th International Conference on Information and Communication Systems (ICICS)*.
- Pagano, T., Loureiro, R., Lisboa, F., et al. (2023). Bias and unfairness in machine learning models: A systematic review on datasets, tools, fairness metrics, and identification and mitigation methods. *Big Data and Cognitive Computing*.
- Pamisetty, L., Singireddy, S., Annapareddy, V., et al. (2022). Leveraging AI, machine learning, and big data for enhancing tax compliance, fraud detection, and predictive analytics in government financial management. *Migration ...*
- Pamisetty, V. (2020). Optimizing tax compliance and fraud prevention through intelligent systems: The role of technology in public finance innovation. *International Journal on Recent and Innovation Trends in Computing and Communication*.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*. <https://arxiv.org/abs/1707.06347>
- Scott, W., Richardson, L., Lewis, D., Ward, P., Phillips, S., et al. (2024). Integration of AI and cloud in tax monitoring systems. *researchgate.net*.
- Stoller, S., Bartocci, E., Seyster, J., Grosu, R., et al. (2011). Runtime verification with state estimation. *International Conference on Runtime Verification*.
- Wachter, S., Mittelstadt, B., & Russell, C. (2018). Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology*, 31(2), 841–887. <https://jolt.law.harvard.edu/assets/articlePDFs/v31/Counterfactual-Explanations-without-Opening-the-Black-Box-Sandra-Wachter-et-al.pdf>
- Wang, Z. (2024). Adaptive ensemble learning framework with shap-based feature optimization for financial anomaly detection. *Journal of Artificial Intelligence Review*.