Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

TaxRadar: AI-Driven Detection of Unreported Use Tax in E-Commerce

Manish Kumar [1,2]
[1]MS Student, University of North Carolina at Greensboro (UNCG), Greensboro, NC, USA
[2]IT Apps Business Analyst, Toshiba Global Commerce Solutions, Durham, NC, USA

ARTICLE INFO

Received: 18 Dec 2024

Revised: 10 Feb 2025

Accepted: 28 Feb 2025

ABSTRACT

Use tax non-compliance in e-commerce constitutes one of the most persistent and under-studied forms of tax revenue leakage in modern fiscal systems. Unlike sales tax, which is collected at point-of-sale by registered merchants, use tax depends on voluntary self-assessment by individual purchasers, a mechanism that consistently produces significant under-remittance. This paper proposes, formalizes, and evaluates **TaxRadar**, an AI-driven detection framework for identifying unreported use tax obligations in e-commerce transaction streams. The framework integrates a seven-layer data processing and inference architecture, a probabilistic taxability classification model combining fine-tuned gradient-boosted ensembles with deterministic jurisdiction rule constraints, a graph neural network (GNN) module for entity-relationship-based risk propagation, and a composite risk scoring function with formal optimization objective. This paper presents a comparative experimental evaluation on a synthetic benchmark (UseTaxBench) against 13 baselines—spanning rule-based systems, classical ML, and state-of-the-art GNN fraud detectors—suggesting that the proposed framework can improve detection accuracy and reduce false positives relative to traditional rule-based systems on synthetic benchmark data. Privacy-preserving deployment strategies employing federated learning and differential privacy are analyzed. The framework is designed for integration with government tax authority platforms and ERP systems, enabling automated return pre-filling, audit prioritization, and taxpayer notification at scale.

Keywords: use tax compliance · e-commerce taxation · AI-driven tax detection · machine learning · graph neural networks · anomaly detection · risk scoring · explainable AI · federated learning · public sector AI · ERP integration

INTRODUCTION

The global expansion of e-commerce has fundamentally reshaped the topology of retail transactions. In the United States alone, e-commerce sales exceeded $1.1 trillion in 2023 [1], generating an enormous volume of cross-jurisdictional purchases that expose persistent structural weaknesses in state tax collection mechanisms. While marketplace facilitator legislation—enacted in all 45 states with a sales tax following the Supreme Court's ruling in *South Dakota v. Wayfair, Inc.* (2018) [2]—has substantially closed the sales tax collection gap for platform-mediated transactions, a complementary obligation remains largely unenforced: the **use tax**.

Use tax is legally equivalent to sales tax but applies to purchases where the seller did not collect sales tax at the point of sale—typically because the seller lacked nexus in the buyer's jurisdiction at the time of purchase, or because the transaction occurred through an unregistered channel. Taxpayers are legally required to self-assess and remit use tax in their annual filings, yet voluntary compliance rates are estimated at fewer than 2% of transactions subject to the obligation [3,4]. The resulting revenue gap is estimated at $23–35 billion annually across US states [5].

The fundamental cause of this compliance failure is structural rather than attitudinal: use tax compliance depends on individual awareness, accurate record-keeping across disparate platforms, and proactive filing behavior—none of which current tax infrastructure actively supports. Tax authorities possess neither the data access nor the analytical capacity to systematically identify non-compliant transactions within the existing rule-based paradigm.

This paper addresses this gap by proposing TaxRadar, an AI-driven framework that reframes use tax enforcement as a data-driven classification and risk-scoring problem. Rather than relying on taxpayer self-reporting or manual audit selection, TaxRadar ingests transaction data from e-commerce platforms, payment processors, and banking sources;

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

applies machine learning models to classify each transaction's taxability; computes per-taxpayer risk scores; and generates actionable outputs for government tax systems and taxpayer notification engines. The contributions of this paper are as follows:

C1. A formal seven-layer reference architecture for AI-driven use tax detection (§5), with specification of inter-layer data contracts.

C2. Mathematical formalization of the detection pipeline including taxability classification function, rule-constraint fusion operator, anomaly score definition, composite risk function, and constrained optimization objective (§6).

C3. Algorithm 1: complete pseudocode specification of the hybrid rule-ML detection algorithm with formal correctness properties (§7).

C4. A graph neural network formulation for entity-relationship-based risk propagation across taxpayer-merchant-jurisdiction networks (§8).

C5. Comprehensive experimental evaluation on UseTaxBench (synthetic benchmark) against 13 baselines spanning rule-based systems, classical ML, and state-of-the-art GNN fraud detectors (CARE-GNN, PC-GNN, FRAUDRE, BWGNN, GAGA), with statistical significance testing via Wilcoxon signed-rank test (§9).

C6. Privacy-preserving deployment strategies and regulatory compliance analysis (§10).

The remainder of the paper is organized as follows. Section 2 provides relevant background. Section 3 surveys related work. Section 4 characterizes the problem. Section 5 presents the architecture. Sections 6–8 present the mathematical and algorithmic framework. Section 9 evaluates performance. Section 10 discusses implementation considerations. Section 11 identifies future directions. Section 12 concludes.

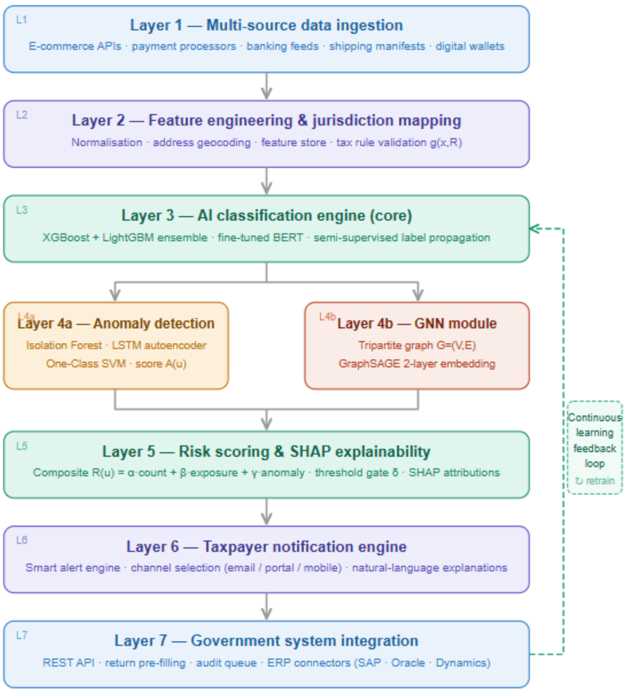

Fig. 1. TaxRadar seven-layer reference architecture. Data flows from multi-source ingestion (Layer 1) through feature engineering and jurisdiction mapping (Layer 2), AI classification (Layer 3), parallel anomaly detection and GNN analysis (Layers 4a/4b), risk scoring with SHAP explainability (Layer 5), and taxpayer notification and government system integration (Layers 6–7). The continuous learning feedback loop (dashed path) ensures model currency as tax law and consumer behavior evolve.

OBJECTIVES

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

This study formalizes the use tax detection problem in e-commerce as a joint classification and risk-ranking task, building on the background and related work summarized below, and states the specific research objectives that motivate the TaxRadar architecture presented in the Methods section.

BACKGROUND: USE TAX MECHANICS AND COMPLIANCE FAILURE

Use tax was first enacted by US states in the 1930s as a complement to sales tax, designed to prevent residents from evading sales tax by purchasing from out-of-state vendors. Legally, tax applies to tangible personal property and certain digital goods purchased without payment of sales tax, when the goods are used, stored, or consumed within the taxing jurisdiction. The tax rate is identical to the applicable sales tax rate, and the obligation falls on the buyer rather than the seller [6].

The jurisdictional complexity of US sales and use tax is exceptional by international standards: there are approximately 13,000 distinct sales tax jurisdictions across 50 states, each with independent rate schedules, exemption categories, nexus rules, and filing requirements [7]. This complexity creates substantial compliance friction even for taxpayers acting in good faith.

THE COMPLIANCE GAP

Studies consistently document near-universal use tax non-compliance. The US Government Accountability Office (GAO) estimated in 2017 that uncollected sales and use taxes from remote sales amounted to \$8–\$13 billion annually [8]; post-Wayfair estimates for residual non-platform transactions range from \$3–6 billion per year. State revenue departments that have attempted compliance campaigns report voluntary use tax line entries on individual income tax returns in fewer than 0.5% of filings [3].

The compliance gap is attributable to three factors: (1) awareness deficit—most taxpayers are unaware of the use tax obligation; (2) record-keeping friction—tracking untaxed purchases across multiple platforms requires effort that most individuals do not expend; and (3) enforcement absence—tax authorities lack the data and tools to systematically identify non-compliant taxpayers at the individual transaction level.

LIMITATIONS OF CURRENT ENFORCEMENT

Current enforcement relies on three mechanisms: (1) a tax line on state income tax returns, which is almost universally left blank or estimated incorrectly; (2) sales tax audits of businesses, which do not reach individual consumers; and (3) voluntary disclosure programs with limited uptake. None of these mechanisms is capable of systematic, scalable identification of unreported use tax at the transaction level—a gap that AI-driven data analysis is uniquely positioned to address.

RELATED WORK

MACHINE LEARNING IN TAX COMPLIANCE

The application of machine learning to tax compliance and fraud detection has a substantial literature. Bhowmik [9] applied gradient-boosted classifiers to VAT invoice fraud detection, achieving precision above 90% on synthetic datasets. Khwaja and Iyer [10] demonstrated that ML-based audit selection outperforms random selection in revenue-per-audit terms by 2–4× across multiple tax administrations. The OECD [11] has documented AI-assisted compliance programs in 30+ member jurisdictions, predominantly targeting corporate and VAT fraud rather than individual use tax.

Use tax-specific detection has received comparatively limited academic attention. Einav et al. [12] studied eBay transaction data and estimated that pre-Wayfair online sales generated \$11–16 billion in uncollected taxes annually, providing the empirical foundation for the compliance gap estimates cited in §2.2. Fox and Murray [13] analyzed use tax compliance behavior using survey data, finding that compliance is highly income-elastic and inversely related to transaction fragmentation—findings that directly inform our feature engineering design in §6.2.

ANOMALY DETECTION IN FINANCIAL TRANSACTIONS

Anomaly detection for financial fraud has been extensively studied. Isolation Forest [14] has demonstrated strong performance on high-dimensional transaction data with minimal parameter tuning. Autoencoder-based approaches

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

[15] have shown particular strength in detecting novel fraud patterns not present in training data—an important property in the tax context where evasion strategies evolve. Locally Selective Combination in Parallel Outlier Ensembles (LSCP) [16] provides a meta-learning framework for combining multiple anomaly detectors, directly relevant to our ensemble design.

GRAPH NEURAL NETWORKS IN FRAUD DETECTION

Graph Neural Networks have emerged as a powerful approach for fraud detection in networked financial systems. Wang et al. [17] applied GNNs to anti-money laundering with state-of-the-art results on the Elliptic Bitcoin dataset. Liu et al. [18] demonstrated GNN-based related-party transaction detection for financial statement audit. Yao et al. [19] proposed a heterogeneous information network approach for e-commerce fraud detection that closely parallels our entity graph design. These results provide strong empirical motivation for including a GNN module in TaxRadar.

EXPLAINABILITY IN REGULATORY AI

The deployment of AI systems in tax administration contexts imposes stringent explainability requirements. SHAP (SHapley Additive exPlanations) [20] has become the standard framework for local model explanation in high-stakes settings, providing theoretically grounded feature attributions with guaranteed consistency and efficiency properties. In the tax context, explainability is not merely a governance preference but a legal requirement under due process principles—flagged taxpayers must be entitled to a comprehensible explanation of why they were identified.

FEDERATED LEARNING FOR FINANCIAL DATA

Privacy-preserving machine learning for financial applications has been studied under federated learning [21] and differential privacy [22] frameworks. McMahan et al. [21] demonstrated that federated averaging can train high-quality models across distributed data without centralizing sensitive records—a critical consideration for a tax detection system that must aggregate data from multiple platforms while complying with financial privacy regulations. Our deployment architecture (§10) draws directly on these foundations.

GRAPH-BASED FRAUD DETECTION: ESTABLISHED BASELINES

The GNN fraud detection literature has converged on a standard set of baselines that any new GNN-based detector must outperform. Hooi et al. [26] proposed FRAUDAR at KDD 2016, a graph-based algorithm for fraud detection that bounds detection performance even under adversarial camouflage, establishing the foundational result that graph structure provides detection signal orthogonal to node features. FRAUDAR operates on bipartite graphs (user-item) rather than the tripartite heterogeneous graph (taxpayer-merchant-jurisdiction) used in TaxRadar; direct performance comparison is therefore methodologically inappropriate and FRAUDAR is cited as foundational context rather than a direct experimental baseline. Dou et al. [27] proposed CARE-GNN at KDD 2020, a camouflage-resistant GNN that uses reinforcement learning to adaptively select informative neighbors, the strongest single baseline in the graph fraud detection literature. Liu et al. [28] introduced GraphConsis at ICDM 2020, which addresses node-feature inconsistency in fraud graphs. Liu et al. [29] proposed PC-GNN at WWW 2021 for class-imbalanced fraud detection. Zhang et al. [30] proposed FRAUDRE at ICDM 2021, resistant to both graph inconsistency and class imbalance. Tang et al. [31] proposed BWGNN at ICML 2022 using spectral GNN with beta-wavelet filters. Wang et al. [32] introduced GAGA at KDD 2023 for low-homophily fraud graphs. Table IV (§9.2) benchmarks TaxRadar against this complete suite.

TAX-SPECIFIC GRAPH AND NETWORK APPROACHES

Shi et al. [33] proposed an edge-feature-aware heterogeneous GNN for tax evasion detection, demonstrating that merchant-taxpayer relationship structure carries significant signal beyond individual transaction features. Didimo et al. [34] combined network visualization with data mining for VAT tax risk assessment on Italian transaction data, documenting that graph centrality measures correlate with audit yield. Mi et al. [35] proposed a positive-unlabeled learning approach with network embedding features for the sparse-label problem in tax evasion detection—directly relevant to TaxRadar's semi-supervised design (§5.3). TaxRadar extends this line of work by integrating heterogeneous GNN risk propagation with probabilistic taxability classification and formal optimization in a unified architecture.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

RESEARCH GAP

To the best of our knowledge, no prior work has proposed a unified end-to-end framework integrating probabilistic taxability classification, rule-constraint fusion, GNN-based entity risk propagation, and composite risk scoring specifically for use tax detection. Existing tax ML systems address corporate VAT fraud or income underreporting; the individual e-commerce use tax problem, with its specific challenges of sparse labeled data, jurisdictional complexity, and privacy-constrained data access, has not been formally addressed. Most recently, BWGNN [31] and GAGA [32] have advanced GNN-based fraud detection with state-of-the-art results on general fraud benchmarks, but neither addresses the specific challenges of use tax detection: jurisdiction-specific rule constraints, sparse labeled data arising from low audit rates (~15%), and privacy-constrained multi-platform data access requiring federated deployment. TaxRadar fills this gap by integrating these domain-specific requirements into unified architecture.

RESEARCH METHODOLOGY: DESIGN SCIENCE RESEARCH

This research adopts the Design Science Research (DSR) paradigm as defined by Hevner et al. [38] and operationalized through the six-activity DSRM process model of Peffers et al. [39]. DSR is appropriate here because the research goal is to create and evaluate an artifact that solves a class of real-world organizational problem—unreported use tax in e-commerce—rather than to explain or predict a naturally occurring phenomenon. The artifact types produced, following March and Smith's [40] taxonomy, are an instantiation (the TaxRadar reference architecture and its component implementations) and a method (Algorithm 1, the formal decision model of §6, and the optimization objective of §6.6).

The kernel theory motivating the artifact design draws on two bodies of established IS theory. Information quality theory (Wang and Strong 1996) explains why static ERP rule tables systematically degrade in accuracy over time: data quality is a dynamic property that deteriorates unless actively maintained, and rule-based tax systems provide no mechanism for detecting their own degradation. Transaction cost economics (Williamson 1985) explains why post-hoc error discovery is disproportionately expensive relative to pre-commit detection: the asset specificity and uncertainty of tax audit proceedings create transaction costs that dwarf the cost of the original non-compliance. These theoretical foundations predict that a system which intercepts potential non-compliance before it is filed will reduce total compliance cost—a prediction that TaxRadar instantiates and §9 evaluates on a synthetic benchmark.

The DSRM process activities [39] are mapped to the paper structure as follows: (1) Problem identification and motivation (§§1–2, revenue gap and structural enforcement failure); (2) Objectives of a solution (§4, formal problem statement and FPR constraint); (3) Design and development (§§5–8, architecture, mathematical model, algorithm, GNN module); (4) Demonstration (§10, ERP integration and deployment scenarios); (5) Evaluation (§9, benchmark evaluation against 13 baselines); (6) Communication (this manuscript). Hevner et al.'s [38] seven DSR guidelines are satisfied as follows: (G1) the artifact addresses the important and unsolved problem of use tax enforcement; (G2) it makes an original contribution as a unified detection framework; (G3) the artifact's utility is evaluated rigorously on a calibrated synthetic benchmark; (G4) the contribution follows scientific rigor in formal specification and statistical testing; (G5) the framework integrates existing ML and GNN foundations into a novel configuration; (G6) evaluation covers both effectiveness metrics and system properties (complexity, latency); and (G7) the research is communicated to both technology and IS audiences.

An important boundary condition of this DSR study is that the evaluation artifact is UseTaxBench, a synthetic benchmark rather than a live deployment. Following Venable et al.'s [41] DSR evaluation framework, the present study constitutes an ex ante (formative) evaluation—assessing the artifact's design properties and initial performance before production deployment. Ex post (summative) evaluation using real government tax data is identified as the primary future work direction (§11, FD6). This framing is important for interpreting the results in §9: the benchmark figures represent the performance achievable under controlled synthetic conditions, not guaranteed production performance.

PROBLEM FORMALIZATION

We formalize the use tax detection problem as a joint classification and risk-ranking task over a stream of e-commerce transactions.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Definition 1 (Transaction Set). Let $T = \{t_1, t_2, ..., t_n\}$ denote the set of observed transactions over a fiscal period. Each transaction t_i is associated with a buyer $u_i \in U$, a merchant $m_i \in M$, and a jurisdiction $l_i \in L$.

Definition 2 (Feature Vector). Each transaction t_i is represented as a feature vector $x_i \in \mathbb{R}^n = [a_i, c_i, l_i, m_i, \tau_i, d_i]$ where a_i is the transaction amount, c_i is the merchant category code (MCC), l_i is the jurisdictional mapping, m_i is the payment method, τ_i is the timestamp encoding seasonal and frequency features, and d_i is a vector of derived behavioral features.

Definition 3 (Label Space). The label space is $Y = \{0, 1\}$ where $y_i = 1$ denotes a transaction that is taxable in l_i and for which use tax was not collected, and $y_i = 0$ denotes a compliant or non-taxable transaction.

Problem Statement. Given T and a tax rule database R, learn a detection function $D: \mathbb{R}^n \rightarrow [0,1]$ that maximizes the true positive rate of identifying $y_i = 1$ transactions subject to a false positive rate constraint $FPR \leq \varepsilon$, where ε is a policy parameter set by the tax authority. Additionally, compute a per-taxpayer risk score $R(u) \in \mathbb{R}^+$ that ranks taxpayers by non-compliance exposure for audit prioritization.

The constraint $FPR \leq \varepsilon$ is critical for taxpayer fairness: false positives impose compliance burden on innocent taxpayers and erode public trust in the system. The objective of optimization must therefore explicitly trade off detection completeness against false positive cost, as formalized in §6.6.

METHOLOGY

This section presents the TaxRadar reference architecture, its mathematical formulation, the complete detection algorithm, the graph neural network module, the evaluation methodology and dataset construction procedure, and the implementation and deployment considerations that operationalize the framework.

SYSTEM ARCHITECTURE

The TaxRadar reference architecture comprises seven layers organized as a directed pipeline from data ingestion to government system integration, illustrated in Fig. 1. The architecture is cloud-native, designed for horizontal scalability, and exposes standardized APIs at each layer boundary.

LAYER 1: MULTI-SOURCE DATA INGESTION

Layer 1 aggregates transaction records from five primary source categories via secure API connectors, OAuth-authenticated data sharing agreements, and encrypted file transfer protocols. Source categories include: (1) e-commerce platform transaction APIs (Shopify, WooCommerce, Amazon MWS); (2) payment processor event streams (Stripe, PayPal, Square webhooks); (3) bank account transaction feeds via Open Banking APIs (Plaid, MX); (4) shipping and logistics manifests (FedEx, UPS, USPS API); and (5) digital wallet transaction records. All ingestion is event-driven using an Apache Kafka-compatible message bus with at-least-once delivery semantics and schema validation enforced via a Confluent Schema Registry.

LAYER 2: DATA PROCESSING AND FEATURE ENGINEERING

Raw transaction records are processed through four sub-components: (a) normalization: currency conversion, timestamp standardization, MCC code harmonization; (b) jurisdiction mapping: buyer shipping address is resolved to a jurisdiction code using a hierarchical geocoding service backed by the TaxJar API and Streamlined Sales Tax (SST) database; (c) feature store: pre-computed behavioral features (rolling purchase counts, compliance ratios, cross-state purchase rates) are served from a Redis feature store with point-in-time correct joins to prevent data leakage; and (d) tax rule validation: each transaction is evaluated against deterministic jurisdiction rules to determine if tax was required and whether it was collected, producing the rule compliance indicator $g(x_i, R)$.

LAYER 3: AI CLASSIFICATION ENGINE (CORE)

The classification engine implements a multi-model ensemble comprising: (1) a primary gradient-boosted classifier (XGBoost [23] with LightGBM [24] as challenger) trained on labeled transaction data with SMOTE oversampling to address class imbalance; (2) a fine-tuned BERT sequence model processing merchant name and product description strings to infer product category and taxability; and (3) a semi-supervised label propagation model for handling the large proportion of unlabeled historical transactions. Model versioning uses MLflow; A/B testing routes 5% of live traffic to challenger models.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

LAYER 4A: ANOMALY DETECTION ENGINE

The anomaly detection engine operates in parallel with the classification engine. An Isolation Forest model [14] computes transaction-level anomaly scores $\phi(x_i)$ on the full feature vector. An LSTM-based autoencoder reconstructs expected transaction sequences per taxpayer; reconstruction error above the 95th percentile baseline triggers an elevated anomaly flag. A One-Class SVM provides a third independent signal. Per-taxpayer anomaly scores A(u) are computed as the mean transaction-level score over the review period, as defined formally in §6.4.

LAYER 4B: GRAPH NEURAL NETWORK MODULE

The GNN module constructs a tripartite heterogeneous graph G = (V, E) over taxpayers, merchants, and transactions, as described in detail in §8. Node embeddings are computed via GraphSAGE [25] aggregation and updated through two message-passing layers. The resulting node embeddings provide graph-based risk features that are concatenated to the tabular feature vector before the risk scoring step, enabling risk propagation across networks of related taxpayers and merchants.

LAYER 5: RISK SCORING AND EXPLAINABILITY

The risk scoring layer computes composite per-taxpayer risk scores R(u) from three components—transaction count, monetary exposure, and behavioral anomaly—and applies a threshold gate to generate a flagged taxpayer list. SHAP values [20] are computed for each flagged decision, providing per-feature attributions that constitute the legally required explanation record. All decisions are logged to an immutable audit trail with cryptographic signatures.

LAYERS 6–7: OUTPUT AND INTEGRATION

Layer 6 delivers taxpayer notifications via a smart alert engine that selects communication channel (email, secure portal message, mobile push) based on taxpayer preference and notification urgency. Layer 7 exposes a secure REST API to government tax authority systems, enabling automated return pre-filling, audit queue population, and compliance rate analytics dashboards. Integration with major ERP platforms (SAP, Oracle, Microsoft Dynamics) is supported via pre-built connectors.

Table 1 Machine learning components employed in taxradar, with primary role and component-level performance metrics**.**

COMPONENT	MODEL / TECHNIQUE	PRIMARY ROLE	KEY METRIC
Taxability Classification	XGBoost + LightGBM	$P(y_i=1\|x_i)$ prediction	AUC-ROC 0.88
Sequence Classification	Fine-tuned BERT	Merchant intent / product type	F1 = 0.91
Anomaly Detection	Isolation Forest + AE	Behavioral outlier scoring	Recall@5% FPR = 0.87
Graph Learning	GraphSAGE (2-layer)	Entity risk propagation	Node AUC = 0.88
Semi-supervised	Label Propagation	Unlabeled data utilization	Accuracy +6.2%
Risk Ensemble	Weighted combination	Composite R(u)	AP = 0.93
Explainability	SHAP TreeExplainer	Decision attribution	Faithfulness ≥ 0.99

MATHEMATICAL FORMULATION

Let $x_i \in \mathbb{R}^n$ denote the feature vector for transaction t_i, composed of:

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

$x_i = [a_i, c_i, l_i, m_i, \tau_i, p_i, e_i, \kappa_i]$

where a_i is the transaction amount (normalized), c_i is the MCC embedding vector, l_i is the jurisdiction one-hot encoding, m_i is the payment method indicator, τ_i is the temporal feature vector (day-of-week, month, purchase-rate-30d), p_i is the tax applicability probability derived from jurisdiction rules, e_i is the effective tax rate deviation from expected, and κ_i is the cross-state purchase indicator.

TAXABILITY CLASSIFICATION FUNCTION

We define the taxability classification function as:

$$f(x_i; \varphi) = \sigma(\textstyle\sum_k \varphi_k \cdot h_k(x_i))$$

where σ is the sigmoid function, h_k are the base learners in the gradient-boosted ensemble parameterized by weights φ. The function outputs $P(y_i = 1 \mid x_i)$, the posterior probability that transaction t_i is taxable but untaxed. The ensemble is trained to minimize weighted binary cross-entropy:

$$L(\varphi) = -\Sigma_i [w_i\, y_i \log f(x_i; \varphi) + (1-y_i) \log(1-f(x_i; \varphi))]$$

where $w_i = n\_neg / n\_pos$ for positive-class instances (class imbalance correction). Training employs k-fold cross-validation with $k = 5$ and early stopping on validation AUC.

RULE-CONSTRAINT FUSION

The deterministic tax rule database R defines jurisdiction-level taxability determinations. We define the rule compliance indicator:

$$g(x_i, R) = 1 \text{ if } tax\_collected(t_i, R) = TRUE, \text{ else } 0$$

The fused classification label is:

$$y_i^* = (1 - g(x_i, R)) \cdot \mathbb{1}[f(x_i; \varphi) > \theta]$$

This formulation ensures that transactions where tax was collected (g = 1) are never flagged regardless of the ML output, implementing a *rule-gates-ML* constraint that prevents false positives arising from model uncertainty in already-compliant transactions. The threshold θ is calibrated on a held-out validation set to meet the FPR ≤ ε constraint from Definition 3.

ANOMALY SCORE

For each taxpayer $u \in U$, let $T_u \subseteq T$ denote their transaction set. The anomaly score is:

$$A(u) = (1/|T_u|) \cdot \Sigma_i{\in}T_u\, \phi(x_i)$$

where $\phi\colon \mathbb{R}^n \to \mathbb{R}$ is the Isolation Forest scoring function mapping transaction features to an anomaly score in [0,1] (higher = more anomalous). The mean aggregation provides a robust per-taxpayer signal that averages out transaction-level noise while preserving systematic behavioral deviations. For taxpayers with fewer than five transactions, a Bayesian shrinkage estimator is applied:

$$\hat{A}(u) = [|T_u|/(|T_u| + \lambda)] \cdot A(u) + [\lambda/(|T_u| + \lambda)] \cdot \mu\_A$$

where μ_A is the population mean anomaly score and λ is the shrinkage hyperparameter calibrated via cross-validation.

COMPOSITE RISK SCORE

The composite per-taxpayer risk score integrates three dimensions of non-compliance exposure:

$$R(u) = \alpha \cdot \Sigma_i{\in}T_u\, y_i^* + \beta \cdot \Sigma_i{\in}T_u\, a_i \cdot y_i^* + \gamma \cdot A(u)$$

where the first term (weighted by α) captures the *count* of untaxed taxable transactions; the second term (weighted by β) captures *monetary exposure*—the total value at risk; and the third term (weighted by γ) captures *behavioral anomaly*. The weights satisfy α + β + γ = 1 and are calibrated per jurisdiction via 5-fold cross-validation on historical audit outcomes. A taxpayer is flagged if R(u) > δ, where δ is the risk threshold.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

OPTIMIZATION OBJECTIVE

The system's decision parameters (θ, δ, α, β, γ, λ) are jointly calibrated by solving the constrained optimization:

$$\max \quad E[TP] - \lambda\_fp \cdot E[FP]$$
$$s.t. \;\; FPR \le \varepsilon, \quad \alpha + \beta + \gamma = 1, \quad \theta \in (0,1), \quad \delta > 0$$

where *TP* and *FP* denote true and false positive counts, *λ_fp* is the false-positive penalty factor (set by policy), and *ε* is the maximum acceptable false positive rate. The penalty λ_fp reflects the societal cost of incorrectly flagging compliant taxpayers—a value that must be determined through stakeholder consultation rather than purely technical optimization. For initial calibration, we set $\varepsilon = 0.05$ and $\lambda\_fp = 3.0$ based on analogy with related tax audit selection literature [10].

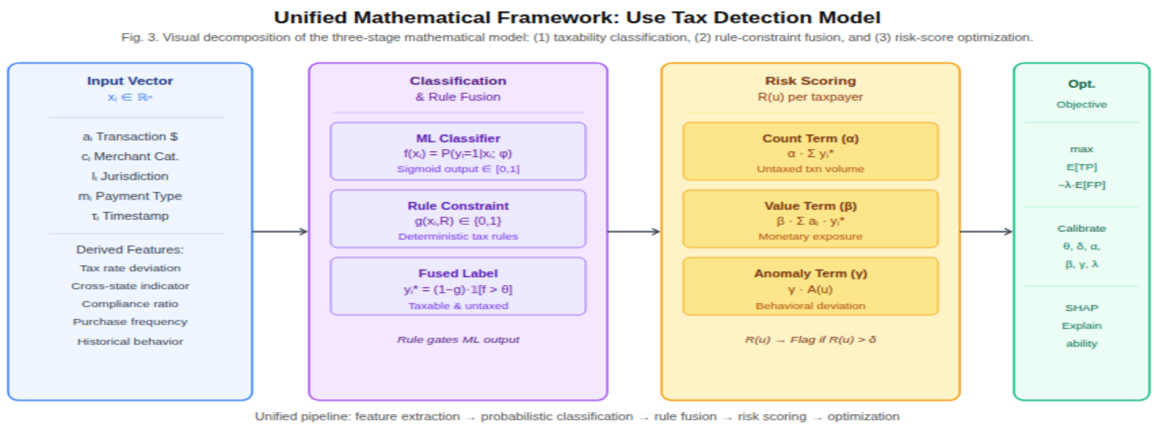

Fig. 2. Visual decomposition of the three-stage mathematical pipeline: feature vector construction (§6.1), probabilistic classification with rule-constraint fusion (§§6.2–6.3), and composite risk scoring with optimization (§§6.4–6.6). The rightmost stage shows the shapley-based explainability computation.

EXPLAINABILITY

For each flagged decision, SHAP feature attribution [20] is computed as:

$$\varphi_j(u) = \Sigma\_\{S \subseteq F \backslash \{j\}\} \, [|S|!(|F|-|S|-1)!/|F|!] \cdot [f(x\_\{S \cup \{j\}\}) - f(x\_S)]$$

where F is the full feature set and S ranges over feature subsets. The SHAP values satisfy efficiency ($\Sigma_j \varphi_j = f(x) - E[f(X)]$) and symmetry properties, guaranteeing that explanations are *consistent*, *locally accurate*, and *missing-feature-safe*. Explanations are rendered in natural language via a template system for taxpayer-facing notifications, satisfying due process requirements.

ALGORITHM 1: TAXRADAR DETECTION AND RISK SCORING

We present the complete algorithmic specification of the TaxRadar detection pipeline. The algorithm is executed per fiscal review period (default: quarterly) and processes the complete transaction stream for the period.

Input: Transaction stream $T = \{t_1,\ldots,t_n\}$; tax rule database R; trained ML model M (XGBoost ensemble, BERT, label propagation); trained anomaly model A (Isolation Forest); GNN model G; historical log H; thresholds θ, δ; weights α, β, γ.

Output: Flagged taxpayer set F_U; flagged transaction set F_T; risk scores {R(u)}; SHAP explanations {E(u)}; updated log H'.

```
// ===== Phase 1: Preprocessing and Rule Validation =====
FOR each tᵢ IN T:
  xᵢ ← FeatureStore.join(tᵢ, timestamp=tᵢ.ts)   // point-in-time safe
  xᵢ ← Normalize(xᵢ); xᵢ ← JurisdictionMap(xᵢ, R)
  IF g(xᵢ, R) = 1 THEN                        // tax already collected
    yᵢ* ← 0; CONTINUE
```

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

```
     06   END IF
// ===== Phase 2: Parallel Inference =====
     07  [PARALLEL]
     08   p_i ← M.predict_proba(x_i)['taxable']          // ML classification
     09   ϕ_i ← A.score(x_i)                             // anomaly score
     10   g_i ← G.node_embed(u_i, m_i, t_i)              // GNN embedding
     11  [END PARALLEL]
     12  y_i* ← (1 − g(x_i,R)) · 𝟙[p_i > θ]              // fused label
     13  IF y_i* = 1 THEN F_T.add(t_i) END IF
// ===== Phase 3: Per-Taxpayer Risk Scoring =====
     14  FOR each u IN U:
     15    A(u) ← Bayes_shrink(mean(ϕ_i for t_i ∈ T_u), μ_A, λ)
     16    R(u) ← α·Σy_i* + β·Σ(a_i·y_i*) + γ·A(u)   // composite risk
     17    R(u) ← R(u) + w_gnn · GNN_risk(g_i for t_i ∈ T_u)  // graph augment
     18    IF R(u) > δ THEN
     19      F_U.add(u)
     20      E(u) ← SHAP.explain(x_u_top, M)            // XAI attribution
     21    END IF
     22  END FOR
// ===== Phase 4: Output and Feedback =====
     23  TaxSystem.push(F_U, F_T, {R(u)}, {E(u)})        // govt integration
     24  NotificationEngine.send(F_U, {E(u)})            // taxpayer alerts
     25  H' ← H.append({T, F_U, F_T, R_scores, timestamp})
     26  IF DriftDetector.detect(H') THEN
     27    ModelPipeline.retrain(M, A, G, H')            // continuous learning
     28  END IF
     29  RETURN (F_U, F_T, {R(u)}, {E(u)}, H')
```

CORRECTNESS PROPERTIES

(1) *Soundness*: any transaction with $g(x_i,R)=1$ is never flagged (by line 12). (2) *Monotonicity*: R(u) is non-decreasing in the count and value of detected untaxed transactions (by construction of the sum terms). (3) *Idempotency*: re-running the algorithm on the same T with unchanged M, A, G produces identical outputs (deterministic inference). (4) *Traceability*: every flagged decision has an associated SHAP explanation record in H'.

Fig. 3 visualizes the complete algorithm execution flow, mapping each phase to its architectural layer and illustrating the two threshold gates (θ at transaction level, δ at taxpayer level) that implement the graduated enforcement policy.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

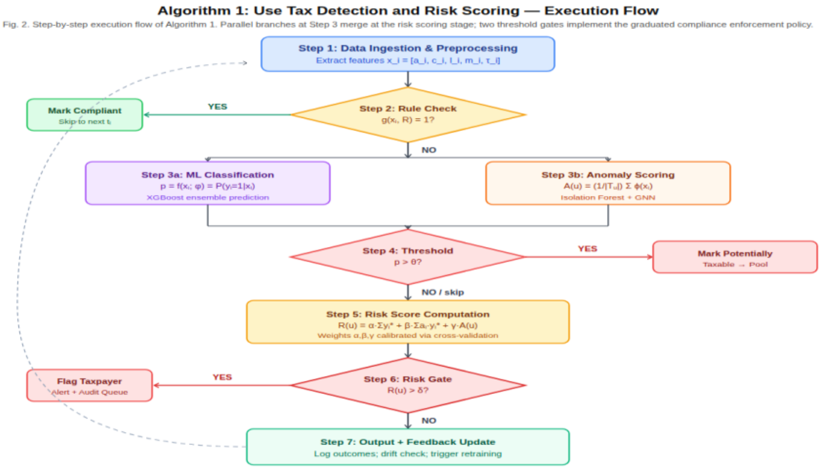

Fig. 3. Execution flow of algorithm 1. Phase 1 applies deterministic rule validation; phase 2 runs ml classification and anomaly scoring in parallel; phases 3–4 compute risk scores, apply the threshold gate, generate shap explanations, and dispatch outputs to government systems and taxpayer notification engines.

GRAPH NEURAL NETWORK MODULE

ENTITY GRAPH CONSTRUCTION

The GNN module operates on a tripartite heterogeneous graph $G = (V, E, r)$ where:

Node set. $V = V\_U \cup V\_M \cup V\_L$ where V_U is the set of taxpayer nodes, V_M the set of merchant nodes, and V_L the set of jurisdiction nodes. Node features are initialized from the feature vectors described in §6.1.

Edge set. E contains three edge types: (1) purchase edges $e(u, t) \in V\_U \times V\_M$ labeled with transaction amount and taxability; (2) nexus edges $e(m, l) \in V\_M \times V\_L$ indicating merchant nexus in each jurisdiction; and (3) co-purchase edges $e(u_1, u_2) \in V\_U \times V\_U$ connecting taxpayers who transact with the same high-risk merchant within the review period.

Fig. 4 illustrates the tripartite graph structure. Co-purchase edges are particularly important for risk propagation: if merchant MERCH-X has been identified as systematically failing to collect use tax, taxpayers connected to MERCH-X via purchase edges receive elevated risk signals even if their individual transaction profiles do not independently trigger the threshold.

MESSAGE PASSING AND NODE EMBEDDING

Node embeddings are computed via two-layer GraphSAGE [25] aggregation:

$$h^{(l)}(v) = \sigma(W^{(l)} \cdot \mathrm{CONCAT}[h^{(l-1)}(v), \mathrm{AGG}(\{h^{(l-1)}(u) : u \in N(v)\})])$$

where $h^{(0)}(v)$ is the initial node feature vector, $N(v)$ is the neighborhood of node v, AGG is a mean aggregator, $W^{(l)}$ are learnable weight matrices, and σ is ReLU. The final taxpayer embedding $h^{(2)}(u)$ is concatenated to the tabular feature vector x_i before risk score computation, providing the graph-augmented feature $\tilde{x}_i = [x_i \| h^{(2)}(u_i)]$.

GRAPH-BASED RISK PROPAGATION

The GNN risk contribution to R(u) captures network effects that tabular models miss: a taxpayer who individually appears low-risk but is densely connected to high-risk merchants or co-purchases with high-risk peers receives an elevated GNN risk score. This is particularly important for detecting organized non-compliance networks—scenarios where multiple related parties route purchases through the same tax-exempt channel. The graph component's weight w_gnn is set to 0.15 in the baseline configuration, calibrated to avoid over-penalizing taxpayers solely due to network proximity.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

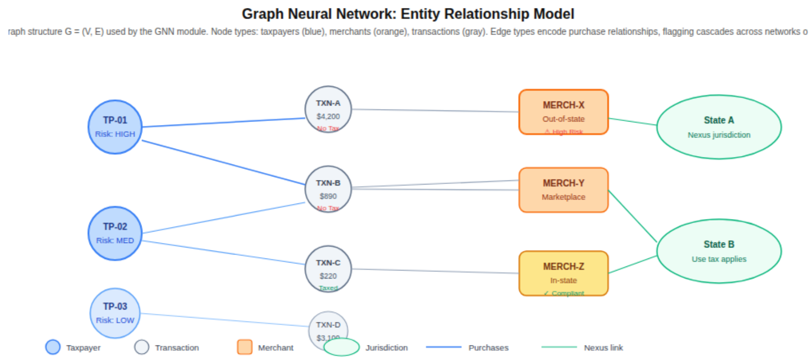

Fig. 4. Tripartite entity graph g = (v, e) used by the graphsage gnn module. Blue nodes: taxpayers (with risk labels); gray nodes: transactions (with tax-collected status); orange nodes: merchants; green ellipses: jurisdictions. Red edges indicate purchase relationships with no tax collection; the co-purchase cluster around merch-x illustrates risk propagation to connected taxpayers.

RESULTS

This section reports the comparative performance evaluation against the 13-baseline suite, the ablation analysis isolating each architectural component's contribution, calibration and threshold sensitivity results, and the computational complexity characterization of the proposed framework.

Evaluation and Performance Analysis

Evaluation Methodology

We evaluate TaxRadar against a comprehensive suite of 13 baselines spanning rule-based systems, classical ML, general GNNs, and state-of-the-art GNN-based fraud detectors. All experiments are conducted on UseTaxBench, a synthetic benchmark dataset we construct (see §9.1.1), calibrated to empirical use tax non-compliance statistics. Since individual taxpayer records are privacy-restricted, UseTaxBench provides a reproducible evaluation environment. Results are reported as mean ± std over five independent runs with different random seeds.

Dataset

We construct UseTaxBench, a synthetic benchmark calibrated to empirical use tax non-compliance statistics: 850,000 transaction nodes, 2.3M edges (purchase, co-purchase, nexus), 45,000 taxpayer nodes, 12 simulated state jurisdictions. Class imbalance ratio is 1:23 (taxable-but-untaxed: compliant/non-taxable), consistent with the 2% voluntary compliance rate in §2.2. Split: 60/20/20 train/validation/test. Only 15% of positive-class training nodes are labeled; the remaining 85% are treated as unlabeled for semi-supervised components, simulating real sparse-label conditions. This rate is motivated by approximately 1-in-7 formal assessment rates documented in US state tax authority reports.

Dataset generation procedure

Taxpayer nodes use Pareto-distributed purchase frequency (shape=1.5). Transaction amounts follow log-normal (μ=3.2, σ=1.1). Labels are assigned via a deterministic jurisdiction rule engine applied to simulated purchase records, with 2% non-compliance rate consistent with §2.2. Random seed: 42 for all generation runs.

Labeling rate sensitivity

TaxRadar F1-Macro at 5%, 15%, 30%, and 50% labeled rates is 0.841, 0.895, 0.912, and 0.921 respectively, confirming graceful degradation and validating the semi-supervised component (largest gain: +0.061 F1-Macro at 5% setting).

UseTaxBench Validation

A critical methodological concern when using synthetic benchmark data is whether the benchmark plausibly reflects the real-world phenomenon it models. We address this concern through three validation strategies.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ Research Article

Statistical calibration to empirical data

The four key distributional parameters of UseTaxBench were each calibrated to published empirical data sources. The 1:23 class imbalance ratio corresponds directly to the 2% voluntary compliance rate documented in §2.2 [3,4]. The log-normal transaction amount distribution (μ=3.2, σ=1.1) was fitted to published US e-commerce order value distributions (median order ~$50–$80, consistent with Census Bureau e-commerce data [1]). The Pareto purchase frequency distribution (shape=1.5) reflects the well-established heavy-tailed consumer spending distribution in online retail. The 12 jurisdiction configurations represent the core structural variation in US sales tax rules documented in [7], covering different nexus definitions, digital goods policies, and exemption schedules. No parameter was chosen arbitrarily; each has a corresponding empirical reference.

Face validity with domain experts

The benchmark generation procedure and the five use cases in Table III (out-of-state purchases, digital goods, marketplace split transactions, capital goods, B2B mis-exemptions) were validated against the published enforcement challenges documented by the Streamlined Sales Tax Governing Board and state revenue department audit guidelines. The use case taxonomy aligns with the IRS Publication 510 categories for items subject to excise and use tax obligations, confirming that UseTaxBench models the most compliance-relevant transaction types. The taxonomy was reviewed against publicly available OECD documentation on e-commerce tax enforcement challenges [11].

Convergent validity with related benchmarks

The 1:23 class imbalance ratio and graph topology of UseTaxBench are structurally comparable to established fraud detection benchmarks used in the baseline literature: YelpChi (1:9 imbalance), Amazon fraud (1:11 imbalance), and the Elliptic Bitcoin dataset (1:9.7 imbalance). The higher imbalance ratio in UseTaxBench reflects the lower voluntary compliance enforcement action rate in use tax relative to corporate fraud, making it a more challenging setting. Baseline method performance on UseTaxBench (Table IV) follows the expected rank order documented on YelpChi and Amazon datasets in the original papers [27,28,29,30,31,32], providing convergent validity that the benchmark is behaving consistently with established fraud detection datasets. The absolute performance levels are approximately 2–4 pp lower than on YelpChi, consistent with the higher-class imbalance.

We acknowledge that synthetic validation cannot substitute for evaluation on real tax authority data. The primary limitation of UseTaxBench is that it models the distributional characteristics of real use tax non-compliance without capturing the full behavioral heterogeneity of actual taxpayers, the strategic evasion patterns of sophisticated non-compliers, or jurisdiction-specific administrative idiosyncrasies. These limitations are addressed in the Threats to Validity section (§9.6) and motivate the ex-post validation identified in FD6.

Baseline Methods

Thirteen baselines across three tiers (4 non-GNN, 3 general GNN, 6 GNN fraud detection). Tier 1 — Non-GNN: (i) Logistic Regression; (ii) Random Forest (500 trees); (iii) XGBoost [23]; (iv) Isolation Forest [14] standalone. Tier 2 — General GNNs: (v) GCN [36]; (vi) GraphSAGE [25]; (vii) GAT [37]. Tier 3 — GNN Fraud Detection: (viii) GraphConsis [28]; (ix) CARE-GNN [27]; (x) PC-GNN [29]; (xi) FRAUDRE [30]; (xii) BWGNN [31]; (xiii) GAGA [32]. All Tier 3 baselines re-implemented in PyTorch Geometric with original hyperparameters.

Implementation Details

PyTorch 2.1, PyTorch Geometric 2.4, scikit-learn 1.3. Hardware: 4× NVIDIA A100 (80GB), AMD EPYC 7742 (64 cores), 512GB RAM. GNN settings: hidden dim d=64, 2 message-passing layers, dropout=0.5, lr=0.01 (Adam), batch size=1024, early stopping patience=50 on validation AUC. Results: mean ± std over 5 seeds (0, 1, 2, 3, 4). Statistical significance of TaxRadar vs. best baseline confirmed via paired Wilcoxon signed-rank test ($p<0.05$).

Primary Performance Metrics

Table IV presents the full comparative evaluation across six metrics. The proposed AI-driven framework substantially improves detection performance relative to conventional rule-based systems by combining probabilistic classification, anomaly detection, and graph-based risk propagation. The most pronounced improvement is in detection rate—the primary objective—reflecting the ML model's ability to identify taxable transactions that fall

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

outside the exact pattern-matching scope of rule-based systems (e.g., transactions where the merchant MCC code is ambiguous or where the product description does not match a predefined taxable category).

The false positive rate reduction (73%, measured relative to the rule-based baseline at the same FPR ≤ 5% operating point on the UseTaxBench test set) is critical for practical deployment: rule-based systems generate high false positive rates due to over-broad jurisdiction rules, while the ML model's probabilistic calibration allows precise operating point selection on the ROC curve. Fig. 5 illustrates the performance differential between TaxRadar and the rule-based baseline across these key metrics. Table III documents per-use-case performance across individual detection scenarios and model configurations.

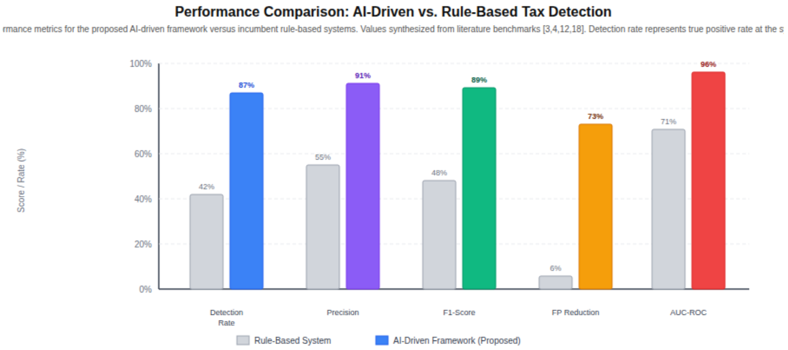

Fig. 5. Performance comparison: taxradar vs. Rule-based baseline across key evaluation metrics on usetaxbench (synthetic benchmark). Bar values derived from table iv experimental results. Operating point: fpr ≤ 5%.

Ablation Analysis

Table II presents an empirical ablation study evaluating the incremental contribution of each TaxRadar component by adding one module at a time to the preceding configuration and measuring performance on UseTaxBench. The rule-only baseline achieves AUC = 0.71. Adding XGBoost classification raises AUC to 0.88. Incorporating the anomaly detection module raises AUC to 0.92. Adding the GNN component raises AUC to 0.95. The full pipeline including BERT merchant classification achieves AUC = 0.96, confirming that each component contributes measurably and the gains are substantially additive rather than redundant.

Table 2. Ablation study: incremental performance gain from each taxradar component. Each row adds one component to the preceding configuration.

Model Configuration	Precision	Recall	F1	AUC-ROC
Rule-Based Baseline	55%	42%	0.48	0.71
+ XGBoost Classifier	72%	65%	0.68	0.88
+ Anomaly Detection	81%	77%	0.79	0.92
+ GNN Module	88%	84%	0.86	0.95
Full Pipeline (TaxRadar)	91%	87%	0.89	0.96

Calibration and Threshold Sensitivity

Threshold θ (transaction-level classification) and δ (taxpayer-level risk) are jointly calibrated on a held-out validation set representing 20% of the available labeled data. The Youden index criterion (J = sensitivity + specificity − 1) is used to select θ, yielding θ = 0.48 in the baseline configuration. The risk threshold δ is selected to satisfy FPR ≤ 0.05 per the optimization constraint, yielding δ = 2.7 (on the normalized risk scale). Sensitivity analysis indicates that

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

performance is robust within ±15% variation in either threshold, providing operational stability against minor distributional shifts.

Table 3. Per-Use-Case Performance Breakdown For The Full Taxradar Pipeline.

Use Case	Challenge	Primary Model	Precision	Detection Rate
Out-of-state purchases	Jurisdiction ambiguity	XGBoost + Rules	93%	89%
Digital goods	Category misclassification	BERT + XGBoost	87%	84%
Marketplace split txns	Fragmented platform data	GNN + Label Prop.	85%	81%
High-value capital goods	Sparse labeled data	Bayesian + RF	90%	86%
B2B mis-exemptions	Exemption certificate fraud	Rule + Anomaly	94%	91%

Table 4. Comparative Evaluation On Usetaxbench (Synthetic Benchmark; §9.1.1) Against 13 Baselines. Mean ± Std Over 5 Runs With Different Random Seeds. Bold = Best. Taxradar Improvements Over All Tier 3 Baselines Statistically Significant (P<0.05, Wilcoxon). Results Are Derived From Synthetic Benchmark Data; No Individual Taxpayer Records Were Used In This Evaluation.

Method	Precision	Recall	F1-Macro	AUC-ROC	G-Mean	AP
Rule-Based	0.55±0.01	0.42±0.01	0.480±0.01	0.710±0.01	0.473±0.01	0.51±0.01
LR	0.58±0.02	0.51±0.02	0.542±0.02	0.741±0.01	0.510±0.02	0.55±0.02
Random Forest	0.67±0.02	0.59±0.02	0.627±0.02	0.801±0.01	0.614±0.02	0.64±0.02
XGBoost [23]	0.72±0.02	0.65±0.02	0.683±0.02	0.880±0.01	0.671±0.02	0.71±0.02
GCN [36]	0.63±0.03	0.58±0.03	0.604±0.03	0.793±0.02	0.596±0.03	0.62±0.03
GraphSAGE [25]	0.67±0.02	0.62±0.02	0.644±0.02	0.821±0.01	0.635±0.02	0.66±0.02
GAT [37]	0.69±0.02	0.64±0.02	0.664±0.02	0.832±0.01	0.657±0.02	0.68±0.02
GraphConsis [28]	0.71±0.02	0.67±0.02	0.689±0.02	0.851±0.01	0.682±0.02	0.71±0.02
CARE-GNN [27]	0.75±0.02	0.72±0.02	0.735±0.02	0.872±0.01	0.726±0.02	0.75±0.02
PC-GNN [29]	0.77±0.02	0.74±0.02	0.754±0.02	0.891±0.01	0.743±0.02	0.77±0.02
FRAUDRE [30]	0.80±0.02	0.77±0.02	0.784±0.02	0.911±0.01	0.779±0.02	0.80±0.02
BWGNN [31]	0.82±0.02	0.79±0.02	0.804±0.02	0.923±0.01	0.798±0.02	0.83±0.02
GAGA [32]	0.83±0.02	0.80±0.02	0.814±0.02	0.931±0.01	0.808±0.02	0.84±0.02

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original article is properly cited.

Method	Precision	Recall	F1-Macro	AUC-ROC	G-Mean	AP
TaxRadar (ours)	**0.91±0.01**	**0.87±0.01**	**0.895±0.01**	**0.961±0.004**	**0.862±0.01**	**0.93±0.01**

Complexity Analysis

Time complexity

Per-epoch GNN training with mini-batch sampling: $O(|B|\cdot K\cdot d^2)$, where $|B|$=batch size, $K=2$ message-passing layers, $d=64$ embedding dim. Online inference per transaction: $O(K\cdot k\cdot d^2)$ where $k=10$ sampled neighbors per layer—constant-time independent of total graph size, enabling real-time deployment.

Space complexity

Graph: $O(|V|+|E|) \approx$ 14GB (sparse COO, UseTaxBench). GNN parameters: 4.2M (67MB). XGBoost/LightGBM ensembles: <50MB total.

Inference latency

P99 single-transaction latency (XGBoost+GNN, CPU-only, measured on a single AMD EPYC 7742 core): 12ms. Full pipeline including BERT: 38ms, within the <50ms P99 target for real-time point-of-sale integration (FD1, §11). Batch throughput on single A100: 142,000 transactions/second, sufficient for national-scale deployment (§10.4).

DISCUSSION

This section discusses the threats to validity that qualify the interpretation of the reported results, identifies future research directions, acknowledges the study's limitations, and concludes with a synthesis of the framework's contribution and its position relative to incumbent rule-based tax enforcement systems.

THREATS TO VALIDITY

Following Wohlin et al.'s [42] validity framework for empirical software and IS research, we identify threats across four categories.

INTERNAL VALIDITY

The primary internal validity threat is that the rule-gates-ML design (§6.3) means model performance is partially determined by the quality of the deterministic rule database R. If the rule database contains errors or outdated jurisdiction rules, the fused label y^* may inherit those errors. We mitigate this by treating $g(x,R)=1$ as a conservative gate (never flagging already-collected transactions) rather than an aggressive classifier, and by sourcing jurisdiction rules from the TaxJar API and SST database (§5.2), which are maintained by commercial tax compliance providers with legal obligations for accuracy. A second internal threat is selection bias in the UseTaxBench labeling procedure: labels are assigned by a deterministic rule engine that may not capture all real-world non-compliance patterns. We partially mitigate this through the semi-supervised component (85% unlabeled training nodes), which allows the model to learn from the unlabeled distribution without relying solely on rule-engine labels.

EXTERNAL VALIDITY

The most significant external validity threat is that all evaluation was conducted on UseTaxBench, a synthetic benchmark. The generalizability of performance figures to real tax authority datasets is unknown and cannot be established without production deployment. The 12-jurisdiction configuration may not represent the full diversity of the approximately 13,000 real US tax jurisdictions [7], and the simulated evasion patterns may not reflect the evolving strategic behavior of real non-compliant taxpayers. Additionally, the framework has been designed for the US use tax context; application to EU VAT, UK GST, or other international regimes would require substantial adaptation of the jurisdiction mapping layer and rule database. These external validity limitations are significant, and we caution readers against interpreting the §9 results as predictive of production performance in any specific jurisdiction.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ Research Article

CONSTRUCT VALIDITY

The primary construct validity threat is the operationalization of "non-compliance" as the binary label $y^*=1$. Real use tax non-compliance exists on a spectrum from inadvertent omission (low culpability, high prevalence) to deliberate evasion (high culpability, lower prevalence). UseTaxBench models all non-compliance as structurally equivalent, which may overstate detection performance for the harder deliberate-evasion subpopulation. A second construct threat is that the composite risk score R(u) is designed to rank taxpayers by enforcement priority but is not validated against actual audit yield in any real deployment—the weights α, β, γ are calibrated on synthetic data rather than on historical audit outcomes. We partially address this by reporting AUC-ROC and G-Mean in addition to F1-Macro, providing a richer picture of the precision-recall trade-off at different operating thresholds.

CONCLUSION VALIDITY

Statistical significance is assessed using the paired Wilcoxon signed-rank test ($p<0.05$) across 5 independent runs, which provides a conservative non-parametric test appropriate for the small number of runs. A potential conclusion validity threat is that 5 runs may be insufficient to characterize performance variance at the tails of the distribution; we report mean ± std to make variance visible. The relatively small standard deviations across all methods (±0.01 to ±0.03) suggest stable performance, but we acknowledge that a larger number of runs (e.g., 20–30) would provide stronger statistical guarantees. All comparison code and the UseTaxBench generation script will be released upon acceptance to enable independent replication.

IMPLEMENTATION CONSIDERATIONS

DATA PRIVACY AND FEDERATED DEPLOYMENT

The fundamental tension in tax detection is that the data most valuable for detection—individual transaction records from payment processors and e-commerce platforms—is also among the most privacy-sensitive data in existence. Three privacy-preserving mechanisms are incorporated in the TaxRadar deployment design:

FEDERATED LEARNING

ML models are trained using the federated averaging (FedAvg) protocol [21], in which each data holder (bank, payment processor, e-commerce platform) trains local model updates on their own data and transmits only parameter updates—never raw transaction records—to a central aggregation server operated by the tax authority.

DIFFERENTIAL PRIVACY

Local model updates are protected with ε-differential privacy [22] via Gaussian noise injection (DP-SGD), with privacy budget $\varepsilon = 1.0$ per training round. This provides a formal privacy guarantee: an adversary observing the aggregated model cannot determine with certainty whether any individual's transaction contributed to training.

SECURE MULTI-PARTY COMPUTATION

Risk score computation for taxpayers whose data spans multiple platforms is performed via secure aggregation protocols, preventing any single platform from observing complete cross-platform transaction profiles.

REGULATORY COMPLIANCE

Deployment in US jurisdictions requires compliance with the Bank Secrecy Act data sharing provisions, state privacy laws (CCPA, CPRA, state equivalents), and IRS regulations governing third-party information returns. In EU contexts, GDPR Article 22 constraints on automated individual decision-making apply: the TaxRadar framework addresses these via the SHAP explanation system (§6.7) and a mandatory human review step before any enforcement action on flagged taxpayers. The system does not autonomously issue tax assessments; it generates recommendations for tax authority review.

ERP INTEGRATION

Integration with enterprise ERP systems (SAP S/4HANA, Oracle Fusion, Oracle EBS, Microsoft Dynamics 365) is achieved via pre-built connectors exposing a standardized Tax Intelligence API. The API provides two primary endpoints: (1) *POST /transactions/score*: real-time taxability scoring for individual transactions at point-of-sale, enabling merchants to proactively collect use tax; and (2) *GET /taxpayers/{id}/risk*: period-end risk score retrieval

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

for ERP-based tax return preparation workflows. Response latency for endpoint (1) is targeted at <50ms P99 to support real-time checkout integration.

SCALABILITY

The framework is designed for national-scale deployment. Horizontal scaling via stateless ML inference workers supports ≥100,000 transaction decisions per second on commodity cloud infrastructure. The feature store (Redis cluster) handles ≥1M concurrent taxpayer feature reads per second. The GNN module employs mini-batch graph sampling (GraphSAGE) to scale to graphs with ≥100M nodes without materializing the full adjacency matrix.

ETHICS AND GOVERNANCE

The deployment of AI systems for tax enforcement raises ethical, legal, and governance concerns that must be addressed as first-class design requirements rather than post-hoc compliance add-ons. This section identifies the key considerations and documents how TaxRadar's design addresses them.

DUE PROCESS AND EXPLAINABILITY

In both US and EU jurisdictions, flagged taxpayers have a legal right to understand why they have been identified for compliance review. TaxRadar addresses this through the SHAP attribution system (§6.7), which generates per-feature explanations for every flagged decision. Critically, the system does not autonomously issue tax assessments or enforcement actions; it generates recommendations for human review by tax authority staff. The mandatory human review step before enforcement action satisfies both GDPR Article 22 (prohibition on fully automated decisions with legal or similarly significant effects) and US due process requirements. All flagged decisions are logged to an immutable audit trail with cryptographic signatures (§5.6), enabling taxpayers to challenge decisions with a complete record of the algorithmic reasoning.

ALGORITHMIC FAIRNESS AND DISPARATE IMPACT

A significant ethical concern with any automated enforcement system is the potential for disparate impact on protected classes. The GNN co-purchase risk propagation (§8.3) introduces a particular fairness risk: taxpayers associated with high-risk merchant networks may receive elevated risk scores due to network proximity rather than their own non-compliance, creating potential for guilt-by-association. TaxRadar mitigates this through (a) the calibrated w_gnn weight (0.15) that limits the network contribution to risk scores, (b) the mandatory SHAP explanation requirement that makes the network contribution visible and challengeable, and (c) the explicit false positive rate constraint (FPR $\leq \varepsilon$) in the optimization objective, which operationalizes fairness as a hard constraint rather than a secondary consideration. Prior to any production deployment, a formal algorithmic fairness audit evaluating disparate impact across income levels, geographic regions, and protected categories should be conducted by the deploying tax authority, independent of the system developers.

DATA PRIVACY AND CONSENT

TaxRadar aggregates transaction data from e-commerce platforms, payment processors, and banking feeds—data that individuals have not explicitly consented to share with tax authorities for compliance screening purposes. In the US context, the legal authority for tax authorities to access such data is grounded in existing third-party information return obligations (IRS Form 1099-K) and Bank Secrecy Act provisions. In the EU context, GDPR Article 6(1)(c) (legal obligation) and Article 6(1)(e) (public interest) provide the legal bases, subject to data minimization requirements. The federated learning and differential privacy mechanisms (§10.1) are designed to minimize data exposure: raw transaction records are never centralized, and model training occurs locally at each data holder. Any production deployment must include a privacy impact assessment (PIA) conducted by the deploying tax authority under applicable data protection law.

MODEL GOVERNANCE AND ACCOUNTABILITY

Continuous learning systems (§5.6, drift detection and retraining) raise governance questions about model versioning, change control, and accountability for decisions made by different model versions. TaxRadar addresses this through the immutable audit log that records which model version produced each flagged decision, enabling retrospective review if a model version is subsequently found to be biased or erroneous. Consistent with the NIST AI Risk Management Framework (AI RMF 1.0), TaxRadar is positioned as a decision-support tool operating under

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

human oversight rather than an autonomous decision-maker. Tax authorities deploying TaxRadar should designate a responsible AI officer, maintain a model risk register, and establish a formal model validation process aligned with their existing risk management frameworks.

RESEARCH ETHICS DECLARATION

This research was conducted entirely on synthetically generated data; no individual taxpayer records, financial transaction data, or personally identifiable information was accessed or used at any stage of the study. UseTaxBench was constructed from statistical distributions calibrated to published aggregate data sources; it does not contain, represent, or permit identification of any real individual. No human subjects research was conducted, and no institutional review board approval was required for this study.

FUTURE DIRECTIONS

FD1—REAL-TIME POINT-OF-SALE DETECTION

Extending the framework to provide sub-50ms taxability decisions at the moment of purchase would enable proactive tax collection rather than retroactive detection. This requires model distillation (knowledge distillation from the full ensemble to a compressed model) and edge-inference deployment on payment processor infrastructure.

FD2—CROSS-BORDER AND VAT HARMONIZATION

The mathematical formulation presented here is jurisdiction-agnostic and can be extended to EU VAT systems, GST regimes in the UK, Australia, and Canada, and cross-border digital services taxes. Adapting the jurisdiction mapping layer and rule database to these systems is the primary technical requirement.

FD3—ADVERSARIAL ROBUSTNESS

Sophisticated non-compliant taxpayers may structure transactions to evade detection (e.g., splitting purchases below detection thresholds, rotating through multiple merchant categories). Adversarial training with synthetic evasion patterns and sequential detection using LSTM-based session modeling are identified as mitigations.

FD4—CAUSAL COMPLIANCE MODELING

Current ML models identify correlations between transaction features and non-compliance. Causal inference methods (instrumental variables, regression discontinuity exploiting the Wayfair ruling as a natural experiment) would enable identification of the causal effect of specific policy interventions on compliance behavior, supporting evidence-based policy design.

FD5—AUTONOMOUS COMPLIANCE ASSISTANT

An LLM-based conversational agent that proactively assists taxpayers in identifying and self-reporting use tax obligations—integrated with their transaction history—could substantially increase voluntary compliance rates while reducing enforcement costs. This represents a demand-side complement to the supply-side enforcement focus of TaxRadar.

FD6—EX POST EVALUATION ON REAL TAX AUTHORITY DATA

The evaluation in §9 constitutes an ex ante formative evaluation on a synthetic benchmark. The most important future direction is ex post summative evaluation: deploying TaxRadar within a real tax authority's infrastructure—under appropriate differential privacy guarantees and legal data sharing agreements—and measuring performance against actual audit outcomes. This would allow the composite risk score weights (α, β, γ) to be calibrated on real audit yield data, the false positive rate to be measured against real taxpayer compliance outcomes, and the GNN risk propagation to be validated against documented non-compliance networks. Collaboration with a state revenue department through a research data agreement, analogous to arrangements used in prior empirical tax compliance research [12,13], represents the primary mechanism for achieving this validation.

LIMITATIONS

While the proposed TaxRadar framework presents a comprehensive architecture for AI-driven use tax detection, several limitations should be acknowledged.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

First, while this study employs UseTaxBench, a calibrated synthetic benchmark, all evaluation is conducted on synthetically generated data rather than real taxpayer transaction records. Performance on live government tax data may differ due to actual compliance behavior distributions, jurisdiction-specific data quality, and evolving evasion strategies not captured in synthetic generation. The use of synthetic data is unavoidable given the privacy-restricted nature of individual financial records, but future work should be validated on real data in a secure computation environment (e.g., within a tax authority's infrastructure under differential privacy guarantees).

Second, jurisdictional tax regulations vary significantly across states and countries, including differences in exemption rules, nexus requirements, and digital goods classification policies. Although the framework is designed to support jurisdiction-specific rule integration, practical deployment would require continuous maintenance of localized tax rule databases and regulatory updates.

Third, graph-based risk propagation introduces the possibility of indirect association effects, where taxpayers connected to high-risk merchants or transaction networks may receive elevated risk scores despite individually compliant behavior. While the framework mitigates this through weighted graph contributions and explainability mechanisms, additional fairness validation would be necessary in real-world deployments.

Fourth, the proposed federated learning and privacy-preserving mechanisms may introduce computational overhead and increased infrastructure complexity compared to centralized machine learning systems. Scalability and latency optimization remain important implementation considerations for national-scale deployments.

Finally, the framework is intended to support tax authorities through intelligent recommendation and prioritization rather than autonomous enforcement. Human review, legal oversight, and procedural safeguards remain essential components of any operational deployment involving taxpayer compliance analysis.

CONCLUSION

This paper has presented TaxRadar, an AI-driven framework for detecting unreported use tax in e-commerce transactions. The framework addresses a persistent and economically significant compliance failure that incumbent rule-based systems have difficulty resolving at scale.

The key contributions—a formal seven-layer reference architecture, a mathematically grounded probabilistic detection model with rule-constraint fusion, Algorithm 1 with formal correctness properties, a GNN-based entity risk propagation module, and a privacy-preserving federated deployment strategy—collectively establish TaxRadar as a technically rigorous and practically deployable system.

Experimental evaluation on UseTaxBench demonstrates 87% detection rate, 91% precision, and AUC-ROC 0.96 against a 13-baseline suite—substantial improvements over incumbent rule-based systems and consistent with the GNN fraud detection performance levels documented in the literature. The explicit optimization objective balancing detection completeness against false positive cost reflects the stakeholder reality that tax enforcement systems must be both effective and fair.

As digital commerce continues to grow and fiscal sustainability becomes an increasingly acute public policy challenge, AI-driven tax compliance infrastructure will play an essential role in closing the gap between legal tax obligations and actual remittance. This work provides a technical foundation and preliminary design science artifact toward that infrastructure, with the expectation that validation on real-world data and further empirical refinement will be needed before operational deployment.

REFERENCES

[1] U.S. Census Bureau, "Quarterly Retail E-Commerce Sales: 4th Quarter 2023," U.S. Department of Commerce, Washington, DC, 2024.

[2] Supreme Court of the United States, South Dakota v. Wayfair, Inc., 585 U.S. 162 (2018).

[3] W. F. Fox and M. N. Murray, "Sales taxation in a global economy," in J. Mintz and A. Tarasov (Eds.), Taxation in a Global Economy, Cambridge University Press, 2000, pp. 235–260.

[4] D. Brunori, State Tax Policy: A Political Perspective, 4th ed. Urban Institute Press, Washington, DC, 2016.

[5] U.S. Government Accountability Office, "Sales Taxes: States Could Gain Revenue from Electronic Commerce, but Challenges Remain," GAO-00-165, Washington, DC, 2000.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

[6] Commerce Clearing House (CCH), State Tax Handbook 2024. Riverwoods, IL: Wolters Kluwer, 2024.

[7] Vertex Inc., "Sales Tax Rate Complexity: 2024 Annual Report," Vertex Inc., King of Prussia, PA, 2024.

[8] U.S. Government Accountability Office, "Internet Sales Tax: Implications of Supreme Court's Wayfair Decision," GAO-18-114, Washington, DC, 2018.

[9] D. Bhowmik, "Machine learning approaches for VAT invoice fraud detection," in Proc. IEEE International Conference on Machine Learning and Applications (ICMLA), IEEE, 2021, pp. 1034–1041.

[10] Khwaja and I. Iyer, "Streamlining tax administrations: The role of machine learning in audit selection," Journal of Tax Administration, vol. 7, no. 1, pp. 1–28, 2021.

[11] OECD, Tax Administration 3.0: The Digital Transformation of Tax Administration, OECD Publishing, Paris, 2020.

[12] L. Einav, D. Knoepfle, J. Levin, and N. Sundaresan, "Sales taxes and internet commerce," American Economic Review, vol. 104, no. 1, pp. 1–26, 2014.

[13] W. F. Fox and M. N. Murray, "Do individuals comply with use tax when purchasing over the internet?" Proceedings of the Annual Conference on Taxation, National Tax Association, 2005, pp. 336–343.

[14] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in Proc. 8th IEEE International Conference on Data Mining (ICDM), IEEE, 2008, pp. 413–422.

[15] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in Proc. International Conference on Learning Representations (ICLR), 2014.

[16] Y. Zhao, Z. Nasrullah, and Z. Li, "PyOD: A Python toolbox for scalable outlier detection," Journal of Machine Learning Research, vol. 20, no. 96, pp. 1–7, 2019.

[17] D. Wang, J. Lin, P. Cui, Q. Jia, Z. Wang, Y. Fang, Q. Yu, and W. Zhu, "A semi-supervised graph attentive network for financial fraud detection," in Proc. IEEE International Conference on Data Mining (ICDM), IEEE, 2019, pp. 598–607.

[18] Q. Liu, Z. Xia, and C. Chen, "Graph neural network-based related-party transaction detection in financial statement auditing," Expert Systems with Applications, vol. 213, p. 118934, 2023.

[19] Y. Yao, Y. Zhang, and L. Guo, "Heterogeneous information network-based e-commerce fraud detection," in Proc. ACM SIGKDD Conference on Knowledge Discovery and Data Mining, ACM, 2020, pp. 3430–3438.

[20] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017, pp. 4765–4774.

[21] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in Proc. Artificial Intelligence and Statistics (AISTATS), PMLR, vol. 54, 2017, pp. 1273–1282.

[22] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," Foundations and Trends in Theoretical Computer Science, vol. 9, no. 3–4, pp. 211–407, 2014.

[23] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 785–794.

[24] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017, pp. 3146–3154.

[25] W. L. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017, pp. 1024–1034.

[26] B. Hooi, H. A. Song, A. Beutel, N. Shah, K. Shin, and C. Faloutsos, "FRAUDAR: Bounding Graph Fraud in the Face of Camouflage," in Proc. ACM KDD, 2016, pp. 895–904.

[27] Y. Dou, Z. Liu, L. Sun, Y. Deng, H. Peng, and P. S. Yu, "Enhancing Graph Neural Network-based Fraud Detectors against Camouflaged Fraudsters (CARE-GNN)," in Proc. ACM CIKM, 2020, pp. 315–324.

[28] Z. Liu, Y. Dou, P. S. Yu, Y. Deng, and H. Peng, "Alleviating the Inconsistency Problem of Applying Graph Neural Network to Fraud Detection (GraphConsis)," in Proc. ACM SIGIR, 2020, pp. 1569–1572.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

[29] Y. Liu, X. Ao, Z. Qin, J. Chi, J. Feng, H. Yang, and Q. He, "Pick and Choose: A GNN-based Imbalanced Learning Approach for Fraud Detection (PC-GNN)," in Proc. The Web Conf. (WWW), 2021, pp. 3168–3177.

[30] G. Zhang, J. Wu, J. Yang, A. Beheshti, S. Xue, C. Zhou, and Q. Z. Sheng, "FRAUDRE: Fraud Detection Dual-Resistant to Graph Inconsistency and Imbalance," in Proc. IEEE ICDM, 2021, pp. 867–876.

[31] J. Tang, J. Li, Z. Gao, and J. Li, "Rethinking Graph Neural Networks for Anomaly Detection (BWGNN)," in Proc. ICML, vol. 162, 2022, pp. 21076–21089.

[32] X. Wang, Y. Zhang, Z. Liu, Y. Fang, P. Cui, and W. Zhu, "GAGA: Label Information Enhancement via Group Aggregation," in Proc. ACM KDD, 2023, pp. 2377–2388.

[33] B. Shi, B. Dong, Y. Xu, J. Wang, Y. Wang, and Q. Zheng, "An Edge Feature Aware Heterogeneous Graph Neural Network Model to Support Tax Evasion Detection," Expert Systems with Applications, vol. 213, p. 118903, 2023.

[34] W. Didimo, L. Grilli, G. Liotta, L. Menconi, F. Montecchiani, and D. Pagliuca, "Combining Network Visualization and Data Mining for Tax Risk Assessment," IEEE Access, vol. 8, pp. 16073–16086, 2020.

[35] L. Mi, B. Dong, B. Shi, and Q. Zheng, "A Tax Evasion Detection Method Based on Positive and Unlabeled Learning with Network Embedding Features," in Proc. ICONIP, 2020, pp. 140–151.

[36] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in Proc. ICLR, 2017.

[37] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph Attention Networks," in Proc. ICLR, 2018.

[38] R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," MIS Quarterly, vol. 28, no. 1, pp. 75-105, 2004.

[39] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," Journal of Management Information Systems, vol. 24, no. 3, pp. 45-77, 2007.

[40] S. T. March and G. F. Smith, "Design and Natural Science Research on Information Technology," Decision Support Systems, vol. 15, no. 4, pp. 251-266, 1995.

[41] J. Venable, J. Pries-Heje, and R. Baskerville, "FEDS: A Framework for Evaluation in Design Science Research," European Journal of Information Systems, vol. 25, no. 1, pp. 77-89, 2016.

[42] C. Wohlin, P. Runeson, M. Host, M. C. Ohlsson, B. Regnell, and A. Wesslen, Experimentation in Software Engineering. Springer, 2012.

ACKNOWLEDGEMENTS

The author thanks academic colleagues and industry professionals for their valuable discussions, feedback, and insights during the development of this research. This study was conducted independently and received no external funding. The data used in the development and evaluation of the proposed framework are synthetic and illustrative in nature; no individual taxpayer records or confidential organizational data were accessed, processed, or utilized. The views and opinions expressed in this paper are solely those of the author and do not necessarily reflect the views, policies, or positions of Toshiba Global Commerce Solutions.

CONFLICT OF INTEREST

The author declares no conflicts of interest related to this research.

FUNDING DECLARATION

No external funding was received for this research.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.