Leakage-Free One-Year-Ahead Prediction of Corporate Tax Avoidance Proxy Measures in Korea

Wonho Song and Hyungjoon Kim *

Posted Date: 28 February 2026

doi: 10.20944/preprints202602.1863.v1

Keywords: Korea tax avoidance; machine learning; deep learning; transformer; leakage-free one-year-ahead prediction

Article

Leakage-Free One-Year-Ahead Prediction of Corporate Tax Avoidance Proxy Measures in Korea

Wonho Song and Hyungjoon Kim *

Department of Computer Engineering, Changwon National University, 20, Changwondaehak-ro, Uichang-gu, Changwon-si, Gyeongsangnam-do, Republic of Korea

* Correspondence: hyungjoon@changwon.ac.kr

Abstract

Since 2011, the mandatory adoption of Korean International Financial Reporting Standards (K-IFRS) by listed Korean firms has improved the consistency of financial reporting and enhanced comparability across firms and over time This institutional change has made it more feasible to construct long-horizon firm–year panel datasets and apply quantitative predictive analyses. The KoTaP dataset provides standardized firm–year panel data for Korean listed non-financial firms over 2011–2024, and this study empirically evaluates the feasibility of risk screening based on one-year-ahead (t→t+1) forecasting of tax-avoidance proxies (CETR, GETR, TSTA, TSDA) using KoTaP. Specifically, we define an ex-ante setting in which only information observable in year t is used to predict tax-avoidance indicators at t+1. We then propose a leakage-free evaluation protocol that enforces chronological splits and fits all preprocessing steps on the training data only. We further partition input features into raw and derived variables and compare three configurations Raw-only, Derived-only, and Raw+Derived to quantify the contribution of derived feature construction. Finally, we compare three machine-learning models and one deep-learning model under the same evaluation procedure and derive practical implications for model selection and deployment in terms of performance and stability.

Keywords: Korea tax avoidance; machine learning; deep learning; transformer; leakage-free one-year-ahead prediction

1. Introduction

Since 2011, Korea has mandated the adoption of Korean International Financial Reporting Standards (K-IFRS) for listed firms, thereby strengthening the international alignment of financial reporting and improving cross-firm comparability [1–3]. In the 2007 IFRS adoption roadmap, the government and supervisory authorities announced that listed companies would be required to adopt K-IFRS in 2011, and they allowed early (voluntary) adoption from 2009 to facilitate a smooth transition [1,2]. This transition aimed to increase transparency in financial reporting, strengthen investor confidence, and reduce information asymmetry in the context of global capital markets [1–3].

After the introduction of K-IFRS, the standardized reporting framework and disclosure infrastructure have improved comparability across firms and over time, providing a basis for a wide range of applied analyses such as performance evaluation, risk management, and capital-market decision-making [6,8]. Notably, Korea's electronic disclosure system (DART) operates an XBRL-based reporting framework aligned with K-IFRS, improving the accessibility and machine-readability of structured financial information [4,5]. This, in turn, has enabled broader use of disclosure data not only for large-sample empirical research but also for data-driven monitoring and screening as decision-support tools [7]. However, tax-avoidance-related measures can vary substantially depending on variable definitions, assumptions about observability, and how panel

 doi:10.20944/preprints202602.1863.v1

consistency is ensured; thus, practically usable predictive and comparative studies require both a long-horizon research-grade panel and a leakage-free evaluation design [7,9].

To address this research demand, KoTaP (Korean Tax Avoidance Panel) is a public dataset that organizes and standardizes firm–year information for Korean listed non-financial firms (KOSPI and KOSDAQ) over 2011–2024 into a long-horizon panel format [9,10]. Not all firms are observed throughout the full 2011–2024 period; the dataset therefore forms an unbalanced panel in which the number of observed years differs across firms [10]. Centered on tax-avoidance proxies (CETR, GETR, TSTA, TSDA), KoTaP includes multidimensional variables covering firm characteristics, financial performance, stability, growth, and corporate governance. The final sample consists of 12,653 firm–year observations from 1,754 firms (65 variables in total). Moreover, by reconstructing and standardizing raw items extracted from DART disclosures into research-ready panel variables, KoTaP enables reproducible empirical and predictive studies under consistent variable definitions without relying on commercial databases.

Building on KoTaP, this study empirically evaluates the feasibility of prediction-based risk screening for tax-avoidance proxies (CETR, GETR, TSTA, TSDA). To this end, we define an ex-ante scenario that predicts next-year (t+1) tax-avoidance indicators using only information observable in year t, explicitly accounting for the firm–year panel structure. We design a leakage-free evaluation protocol by enforcing chronological splits and adopting fit-on-train-only principle for preprocessing, thereby preventing information leakage. We further partition input features into raw and derived variables and conduct an ablation comparison across three configurations—raw-only, derived-only, and Raw+Derived—to quantify the contribution of feature construction (especially derived-feature design). Finally, under identical conditions, we compare three machine-learning models and one deep-learning model to derive practical implications for model selection and deployment in limited-sample settings, focusing on both performance and robustness.

The main contributions of this study are as follows.

1. **Formulation of an ex-ante t → t+1 prediction problem:** Using KoTaP, we define a one-year-ahead forecasting task for tax-avoidance proxies (CETR, GETR, TSTA, TSDA) and present an application scenario that reflects the firm–year panel structure.
2. **Leakage-free evaluation protocol:** We provide a reproducible evaluation design that enables fair comparisons across methods by enforcing chronological splits and explicitly applying fit-on-train-only principles in preprocessing and model selection to prevent future information from leaking into training.
3. **Quantification of raw vs. derived vs. (Raw+Derived) feature effects (ablation):** By constructing three input configurations—raw, derived, and Raw+Derived—we quantify, by target, the independent predictive contribution of derived variables and their complementary (combined) gains with raw variables.
4. **ML/DL model comparison and selection rationale under limited-sample conditions:** Under a unified protocol, we compare three machine-learning models and one deep-learning model, summarizing performance and robustness differences under practical constraints of Korean financial panel data (sample size, missingness, heterogeneity) and providing deployment-oriented guidance for model choice.
5. **Analysis of feasibility and limitations of prediction-based risk screening:** We compare target-specific prediction difficulty and performance patterns and discuss practical caveats and limitations for real-world use (e.g., availability constraints, persistence differences across indicators).

The remainder of the paper is organized as follows. Section 2 reviews the background and related work. Section 3 describes the prediction task definition, the leakage-free evaluation design, and the machine-learning and deep-learning models used for comparison. Section 4 presents and analyzes the experimental setup and results. Section 5 concludes with implications and future directions.

 doi:10.20944/preprints202602.1863.v1

2. Related Works

2.1. Algorithms for Firm–Year Panel (Temporal/Panel) Data Analysis

This study reviews predictive and analytical methods for long-horizon firm–year panel data, where tabular covariates and temporal (year) information coexist. In this setting, panel- and time-series-based algorithms that jointly account for cross-sectional heterogeneity (across firms) and temporal dependence are essential. Hyndman and Khandakar systematized procedures for automatic identification, estimation, and selection of classical time-series models such as ARIMA and ETS through the forecast package, providing traditional baselines that remain useful even when data are limited [11]. Hyndman et al. formulated exponential smoothing (ETS) models within a state-space framework, enabling consistent likelihood-based estimation and prediction-interval construction as well as automatic model selection [12]. Arellano and Bond developed GMM estimation for dynamic panel models that include lagged dependent variables and proposed diagnostic procedures such as serial-correlation tests and overidentifying-restriction tests; together, these provide an econometric foundation for handling lag dependence and endogeneity in dynamic panels [13]. Salinas et al. (DeepAR) proposed a global probabilistic forecasting approach that jointly learns many related time series, demonstrating that predictive performance can be improved by sharing information across firms even when each individual series is short [14]. Rangapuram et al. introduced the Deep State Space Model, which preserves state-space-model (SSM) structure while conditioning series-specific parameters via deep learning, offering an alternative that combines probabilistic forecasting with structural constraints for data efficiency [15]. Lai et al. (LSTNet) proposed an architecture that combines CNNs and RNNs (with an autoregressive component) to model both short- and long-term patterns in multivariate time series, reporting improved forecasting performance in environments with complex temporal dynamics [16]. Lim et al. (Temporal Fusion Transformer; TFT) presented a multi-horizon forecasting framework that jointly handles static and time-varying covariates, and enhanced interpretability through variable selection and interpretable attention mechanisms [17]. Chronopoulos et al. analyzed the properties of feed-forward neural network (FFN)-based estimation and prediction in panel data and provided theoretical and empirical evidence, summarizing the feasibility and considerations of deep learning in panel contexts [18]. Yang et al. proposed an ML-based panel modeling framework that incorporates panel characteristics such as fixed effects and cross-sectional dependence, providing methodological grounding for bias correction and generalization under panel structures [19]. Since our observations are annual and the time-series length (T) is relatively short, classical time-series models are discussed primarily as firm-specific baselines, whereas deep time-series models are considered mainly as comparison methods from the perspective of global learning via information sharing across firms.

2.2. Algorithms for Tabular Data Analysis

Model choice has a substantial impact on tabular data performance due to nonlinear interactions among variables, heterogeneous scales and distributions, missing values, and the treatment of categorical variables. Traditionally, regularized linear models offer interpretability and stable variable selection, while tree ensembles (random forests and gradient-boosted decision trees) effectively capture complex nonlinearities and interactions and have served as strong baselines across a wide range of tabular tasks. Tibshirani introduced Lasso, establishing a representative baseline for high-dimensional tabular data by inducing coefficient sparsity via L_1 regularization and providing a variable-selection effect that mitigates overfitting [20]. Zou and Hastie proposed Elastic Net, which combines L_1 and L_2 regularization to yield more stable estimation and group-selection behavior than Lasso in settings with many correlated predictors [21]. Breiman introduced Random Forests, showing that a tree ensemble combining bootstrap aggregation and random feature selection can robustly capture nonlinearities and interactions, and it has become a strong general-purpose baseline for tabular data [22]. Friedman proposed the Gradient Boosting Machine (GBM), establishing a general boosting framework that sequentially combines weak learners to minimize a loss function,

 doi:10.20944/preprints202602.1863.v1

which later became the foundation for various GBDT variants such as XGBoost, LightGBM, and CatBoost [23]. Chen and Guestrin developed XGBoost, combining regularization, sparsity-aware handling, and parallel/system optimizations to establish a practical standard for applying GBDT efficiently to large-scale tabular data [24]. Ke et al. introduced histogram-based learning and techniques such as GOSS/EFB in LightGBM, improving training efficiency and strengthening computational scalability and performance on large-scale, high-dimensional tabular datasets [25]. Dorogush et al. proposed CatBoost, emphasizing stable performance on tabular problems with many categorical variables through a target-statistics-based treatment designed to reduce bias [26]. More recently, deep-learning-based tabular architectures such as TabNet, TabTransformer, and FT-Transformer have been proposed, highlighting opportunities for improved performance, interpretability, and representation learning for categorical features [27–29]. Arik and Pfister proposed TabNet, which uses sequential attention to select and focus on important features step-by-step, pursuing both performance and interpretability in deep tabular learning [27]. Huang et al. introduced TabTransformer, using self-attention to learn contextual embeddings of categorical variables, thereby modeling meaning shifts induced by categorical combinations [28]. Gorishniy et al. revisited deep learning for tabular data and organized strong baselines (e.g., ResNet/FT-Transformer families) and training recipes, thereby systematizing key comparison points between tree ensembles and deep-learning approaches [29].

2.3. Application Studies on Prediction/Screening Using Financial and Tax Data

Financial statements and tax-related disclosures are core sources for quantitatively assessing firms' performance and risk, and a substantial body of research has accumulated on prediction-based screening (early warning or signal detection) using such information. In particular, tax avoidance and tax risk are difficult to observe directly and are often inferred via various proxies, motivating the adoption of a wide range of methodologies—from classical econometric models to modern ML/DL approaches. Hanlon and Heitzman provide a comprehensive review of tax research and discuss how challenges in measurement (proxy construction) and identification repeatedly arise across domains including corporate decision-making, the information role of taxes, and asset pricing, with tax avoidance as a prominent example [30]. Frank, Lynch, and Rego present empirical evidence of a positive association between aggressive tax reporting and aggressive financial reporting, suggesting that tax-related signals can interact with financial reporting incentives [31]. Guenther et al. use machine learning to forecast one-year-ahead effective tax rates (ETR) and compare the bias and precision of the predictions, showing that disclosure and accounting items contain substantial information for predicting tax outcomes [32]. Borrotti et al. predict ex-ante the risk of aggressive tax-location choices (e.g., the use of tax havens) by European corporate groups using publicly available accounting information and discuss how such predictions can support preventive enforcement strategies by tax authorities [33].

Rahman et al. build tax-avoidance classification models using Malaysian listed-firm data with logistic regression, decision trees, and random forests, and report that the choice of feature sets—such as industry, governance, and firm characteristics—can affect predictive reliability [34]. Beyond tax avoidance, a related stream of research has developed scoring and screening methods for risks that are similarly hard to observe directly or are confirmed only ex post (e.g., accounting manipulation, fraud, and bankruptcy), typically leveraging financial ratios and derived indicators; this literature is informative for the design of prediction-based risk screening frameworks [35–38]. Beneish proposes the M-score, which combines financial-statement-based indicators to detect the likelihood of earnings manipulation ex-ante, illustrating that derived ratios and composite indices can be effective for screening [35]. Dechow et al. propose an F-score model that predicts the risk of material accounting misstatements (errors/fraud) from financial characteristics, linking the approach to ex-ante risk detection in regulatory and auditing contexts [36]. Altman demonstrates the predictability of bankruptcy (financial distress) through a linear combination of financial ratios (the Z-score), providing a classical starting point for ratio-based risk prediction [37]. Cecchini et al. present

 doi:10.20944/preprints202602.1863.v1

a method that automatically analyzes the MD&A text in 10-K filings to predict "catastrophic events" such as fraud and bankruptcy, highlighting the potential of combining numerical financial data with textual signals [38].

3. Methodology

This section describes (i) data construction and variable design, (ii) an ex-ante (leakage-free) prediction and evaluation strategy, and (iii) benchmark models and implementation details for prediction-based risk screening of tax-avoidance proxies using the KoTaP dataset. Specifically, we consider a firm–year panel setting in which CETR, GETR, TSTA, and TSDA in year $t+1$ are predicted using only information observable in year t. We apply an evaluation protocol that prevents any future information from entering the pipeline, including derived-feature construction and model training/validation. We also distinguish three input configurations Raw-only, Derived-only, and Raw+Derived to quantify how feature design affects predictive performance and robustness.

3.1. Data and Variable Construction

This subsection constructs the analysis sample from the KoTaP (2011–2024) firm–year panel and defines target and input variables, with a focus on feature-set design for one-year-ahead forecasting. The KoTaP dataset consists 12,653 firm–year observations from 1,754 firms over 2011–2024, and it forms an unbalanced panel because the observation window can differ across firms due to incorporation, listing, and delisting events [9,10]. Since the accounting definitions, computational foundations, and preprocessing rules are documented in detail in the KoTaP paper and dataset documentation [9,10], this section focuses on the sample construction rules and feature-set design principles for the one-year-ahead $t \rightarrow t+1$ forecasting task. Preprocessing/scaling and the train/validation/test split rules are described separately in Section 3.2 together with the leakage-free evaluation design.

Our forecasting task is defined as predicting the next-year tax-avoidance proxy $Y_{i,t+1}$ from inputs $X_{i,t}$ observed for firm i at year t. Therefore, the last observed year of each firm has no $t+1$ target and is excluded from training and evaluation. Under an unbalanced panel, we use only consecutive observation intervals where both t and $t+1$ exist for the same firm as prediction pairs.

3.1.1. Target Variables (Tax-Avoidance Proxies)

We predict four corporate tax-avoidance proxies: CETR, GETR, TSTA, and TSDA. Because tax avoidance is difficult to observe directly from outside the firm due to the non-disclosure of taxable income and tax adjustments and because tax burdens are determined by multiple interacting factors (e.g., tax rules, deductions/credits, and deferred taxes) even within the same firm, empirical research commonly uses standardized proxies that can be computed from publicly disclosed financial information as targets [30]. Such proxies aim to capture latent signals of tax avoidance through either (i) relatively low tax burdens relative to earnings (ETR-type measures) or (ii) unusually large gaps between book income and taxable income (book–tax differences; BTD-type measures). Using multiple proxies in parallel is widely adopted to mitigate the limitations of any single indicator [30,31].

CETR (Cash Effective Tax Rate) is a cash-flow-based effective tax rate that scales cash taxes paid by pre-tax income. It provides an intuitive interpretation: conditional on comparable earnings, a lower cash tax outflow may indicate a higher degree of tax avoidance (tax minimization) [39]. GETR (GAAP Effective Tax Rate) is an accrual-based effective tax rate defined as total tax expense divided by pre-tax income; it reflects another dimension of tax burden by incorporating deferred tax effects in addition to cash taxes [30]. Because CETR and GETR summarize tax burdens from the perspectives of cash flows and accounting recognition, respectively, jointly forecasting and evaluating them enables a more multifaceted assessment of tax-related risk signals [30,39]. In this study, all target

 doi:10.20944/preprints202602.1863.v1

values—including CETR and GETR—are taken as computed in KoTaP according to its published definitions and preprocessing rules [9,10].

TSTA and TSDA are additional tax-avoidance proxies provided by KoTaP, designed to complement the ETR-based approach by reflecting signals that are harder to capture using effective tax rates alone—such as those arising from book–tax gaps and shifts in the composition of tax expenses [9,10]. In general, BTD-type measures have been discussed as providing tax-avoidance-related signals via "abnormal gaps" that can be amplified by differences between accounting standards and tax law as well as by tax strategies [30,40]. Importantly, ETR/BTD-based proxies are not directly observed ground-truth labels of tax avoidance but noisy (measurement-error–contaminated) labels constructed from public information [30]. Accordingly, we treat these four indicators as complementary targets and compare (i) predictability itself, (ii) sensitivity to feature configurations (raw/derived) and model choice, and (iii) differences in performance and stability across proxy choices. Moreover, prior ML application studies have reported prediction/screening of effective tax rates or tax-related outcomes from disclosure information [32–34], supporting the practical relevance of our setting from a prediction-based risk screening perspective.

3.1.2. Input Variables and Feature Sets

The KoTaP dataset contains 65 variables, which can be categorized—depending on the analysis purpose—into identifiers/meta variables, raw variables, and derived variables [9,10]. Identifiers/meta variables include firm and year identifiers (e.g., stock, year, name), market and industry classifications (e.g., KOSPI, ind), and incorporation/fiscal information (e.g., fnd_year, fiscal). In this study, these variables are used only for defining panel keys and designing data splits, and they are excluded from the input feature set in the default setting. However, some derived variables computed from meta variables (e.g., AGE derived from fnd_year) may be included as inputs according to KoTaP's definitions [9,10].

Raw variables consist of financial accounts and firm characteristics extracted from public disclosures and financial statements (e.g., asset, sales, liab, tax, ocf, big4, forn, own), while derived variables consist of financial ratios/indices and lag features computed from raw variables (e.g., ROA, LEV, SIZE, LOSS, lag_asset) [9,10]. We use KoTaP's provided construction rules for derived and lag features without modification [9,10], and we ensure that all input features are defined using only information available up to the prediction year t. In no case do we include the prediction target $Y_{i,t+1}$ or any transformed feature that incorporates information from year $t+1$.

KoTaP also provides, in addition to the four tax-avoidance targets (CETR, GETR, TSTA, TSDA), cumulative proxies (e.g., CETR3, GETR5) and adjusted proxies (e.g., A_CETR, A_GETR) [9,10]. We treat these as a separate group of auxiliary tax proxies and explicitly control whether they are included as inputs. For example, cumulative proxies such as $\mathrm{CETR3}_t$ and $\mathrm{GETR5}_t$ are computed by aggregating past periods up to year t according to KoTaP's definitions, and they do not incorporate information from the target year $t+1$ [9,10].

Table 1. Variable taxonomy of KoTaP.

Group	Variables
Identifier / Meta	name, stock, year, KOSPI, fnd_year, fiscal, ind
Raw (directly extracted)	big4, forn, own, c_asset, inv, asset, sales, cogs, dep, tax, rec, ni, ocf, cash, tan, land, cip, intan, liab, c_liab, pti, total, equit
Derived (ratios / indicators)	SIZE, LEV, CUR, GRW, ROA, ROE, CFO, PPE, AGE, INVREC, MB, TQ, LOSS
Derived (lag features)	lag_asset, lag_liab, lag_equit, lag_sales, lag_total, lag1_ni, lag_c_asset, lag_c_liab

 doi:10.20944/preprints202602.1863.v1

Tax-avoidance proxies (targets)	CETR, GETR, TSTA, TSDA
Tax-avoidance proxies (auxiliary)	CETR3, GETR3, CETR5, GETR5, A_CETR, A_GETR, A_CETR3, A_GETR3, A_CETR5, A_GETR5

We define three primary input configurations. Raw-only uses raw variables only, whereas Derived-only uses derived variables only, including financial ratios/indices, growth rates, and lag features. Here, the Derived feature set includes both derived indicators (financial ratios/indices) and lag features. Raw+Derived combines the two sets, allowing us to assess both the complementary information (synergy) that derived variables may provide beyond raw information and the potential redundancy between them. Across all configurations, derived variables are computed solely from information available up to the prediction year t; under no circumstances is the prediction target $Y_{i,t+1}$itself included in the input.

In addition, we conduct extended experiments that explicitly control whether tax-proxy variables are included as input features. Specifically, we consider (i) whether to include lagged target proxies (e.g., $\mathrm{CETR}_{i,t}$, $\mathrm{GETR}_{i,t}$) and (ii) whether to include auxiliary cumulative/adjusted proxies (e.g., $\mathrm{CETR3}_t$, $\mathrm{GETR5}_t$, $\mathrm{A\_CETR}_t$). We then quantify how the use of tax proxies affects predictive performance and stability. All included tax-proxy features are restricted to values observable at year t, ensuring consistency with the ex-ante scenario and the leakage-prevention principle. Table 2 summarizes the resulting feature-set variants by tax-proxy inclusion.

Table 2. Feature-set variants for controlling tax-proxy usage.

Feature set	Raw/Derived setting	Aux tax-proxies used?	Target proxies at time t used?
FS1	Raw-only / Derived-only / Raw+Derived	No	No
FS2	Derived-only / Raw+Derived	Yes	No
FS3	Derived-only / Raw+Derived	No	Yes
FS4	Derived-only / Raw+Derived	Yes	Yes

3.2. Ex-ante Forecasting and Leakage-Free Evaluation

This section presents the prediction task definition and evaluation design under a realistic deployment setting. Specifically, we formulate a one-year-ahead ex-ante forecasting task that predicts next-year tax-avoidance indicators $Y_{i,t+1}$ from covariates $X_{i,t}$ available for firm i at year t. The evaluation follows chronology-preserving data splits and explicitly states leakage-prevention principles to avoid mixing information across training and evaluation, with careful attention to potential duplication and dependence arising from the firm–year panel structure. We use a separate validation split for hyperparameter tuning and model selection to avoid optimistic bias and compare models using multiple metrics that capture both accuracy and robustness.

We first construct training samples from the original firm–year panel in a way that matches the one-year-ahead forecasting setup. For firm i, when observations exist for both years $t$and $t+1$, we define $(X_{i,t}, Y_{i,t+1})$as a single sample, yielding the full dataset $D = \{(X_{i,t}, Y_{i,t+1})\}$. Because KoTaP is an unbalanced panel where the observation window can differ across firms, the number of feasible year-pairs may vary by firm. Rather than excluding firms a priori, we include all observations that

 doi:10.20944/preprints202602.1863.v1

form consecutive year-pairs and use only these for prediction and evaluation. This "year-pair-based construction" directly reflects the ex-ante assumption that $t+1$is predicted using only information observable at time t.

Data splitting is performed to preserve temporal order. We define the training, validation, and test intervals based on the input year t; the corresponding target year is then naturally determined as $t+1$ within each interval. Concretely, the training interval is set to $t \in [2011, 2019]$, which corresponds to predicting targets in $t+1 \in [2012, 2020]$. The validation interval is set to $t = 2021$, predicting the target year $t+1 = 2022$, and the test interval is set to $t = 2023$, predicting the target year $t+1 = 2024$for final evaluation. We further introduce gap years between train–validation and validation–test splits (e.g., $t = 2020$ and $t = 2022$ are not used for evaluation) to mitigate the risk that temporal proximity induces overly optimistic model selection due to dependence across adjacent periods. This design supports a more conservative out-of-time assessment that better reflects generalization to future periods. Table 3 summarizes the specific input years, target years, and their purposes.

Table 3. Time-based data split for ex-ante forecasting (t → t+1).

Split	Input year(s) t	Target year(s) t+1	Purpose	Notes
Train	2011–2019	2012–2020	Model fitting (parameter learning)	Samples are constructed only when both years exist for firm i
Validation	2021	2022	Hyperparameter/epoch selection and model/setting selection (out-of-time)	Out-of-time validation; no parameter updates. Used for hyperparameter/epoch selection
Test	2023	2024	Final evaluation (reported results)	Used once after model selection; no iterative tuning
Gap (unused)	2020, 2022	2021, 2023	—	Buffer years not used in training/validation/testing to reduce temporal leakage/overlap effects

The leakage-free principle is defined along three axes: ex-ante availability of information, preservation of temporal ordering, and separation of fitting and evaluation. First, inputs are strictly limited to variables observable at the prediction year t; under no circumstances is the prediction target $Y_{i,t+1}$ itself used as an input. Second, derived features (e.g., growth rates and lag features) are computed solely from information up to year t, and their construction rules are fixed to prevent any implicit inclusion of $t+1$ information. Third, data transformations and preprocessing pipelines such as scaling/normalization, outlier mitigation, and encoding are fit only on the training period and then applied unchanged to the validation and test periods, ensuring that information from evaluation periods does not influence the training process. Moreover, even in extended experiments that include tax-proxy variables as inputs (e.g., the feature-set variants FS1–FS4 in Section 3.1), all included values are restricted to those observable at year t, and any construction that directly or indirectly references the $t+1$ target is excluded.

Hyperparameter tuning and model selection are conducted in a strictly separated validation framework. Hyperparameters as well as early-stopping criteria and checkpoint/epoch selection are determined based on out-of-time validation performance for $t = 2021 \rightarrow 2022$. Model-parameter learning (gradient updates) is performed only on the training period, while the validation and test periods are used solely for evaluation and selection. The final selected configuration is evaluated

 doi:10.20944/preprints202602.1863.v1

exactly once on the test split ($t = 2023 \rightarrow 2024$) and reported. Because deep learning models can exhibit variability due to random initialization and stochastic training dynamics, we run multiple seeds under the same protocol and report both mean performance and variability (e.g., standard deviation) to compare robustness.

For performance evaluation, we use metrics appropriate for continuous-valued prediction for each target. Specifically, we report mean absolute error (MAE), root mean squared error (RMSE), and the coefficient of determination (R^2) to quantify predictive accuracy, and we compare performance differences across raw/derived input configurations and tax-proxy inclusion under the same split and leakage-free evaluation protocol. Rather than claiming the superiority of a single model, our goal is to systematically analyze under a leakage-free out-of-time setting how input feature design (raw, derived, combined), the extent of tax-proxy usage, and model families (ML vs. DL) affect predictive performance and stability.

3.3. Benchmark Models and Implementation Details

This section introduces three machine-learning models and one deep-learning model used as benchmarks for the KoTaP-based one-year-ahead prediction task and summarizes the implementation and training settings. The benchmark suite includes representative tabular learning algorithms that are widely used in limited-sample settings. All models are trained under the same time-based data splits (Table 3) and the same leakage-free feature protocol (Sections 3.1–3.2); preprocessing is implemented in a model-appropriate manner while preserving the train-only-fit principle. We also specify model-specific training strategies (loss functions, regularization, and early stopping), the hyperparameter search ranges, the number of runs, and reproducibility controls such as fixed random seeds. This improves experimental reproducibility and facilitates transparent interpretation of the results.

3.3.1. Benchmark Models

Our benchmark suite consists of three Gradient Boosting Decision Tree (GBDT) methods frequently reported to deliver strong performance and stability on tabular data and one Transformer-based deep-learning model specialized for tabular inputs (TabTransformer). As machine-learning (ML) baselines, we use XGBoost, LightGBM, and CatBoost. XGBoost is a representative GBDT implementation that provides regularized boosting optimization and a variety of regularization options; it robustly captures nonlinearities and feature interactions and is widely used as a standard baseline for predictive modeling on tabular financial data [24]. LightGBM is a GBDT implementation that combines histogram-based split finding with efficient training strategies, offering both fast training and strong performance, and has become a powerful baseline for large-scale, high-dimensional tabular datasets [25]. CatBoost provides boosting methods designed with categorical-feature handling in mind and is included as a comparator because it can deliver stable performance in settings that include binary/categorical firm characteristics [26].

As the deep-learning (DL) baseline, we use TabTransformer. TabTransformer is a Transformer-based tabular model that contextualizes categorical-feature embeddings via self-attention to learn cross-feature interactions; it serves as a representative baseline to assess the potential benefits of deep-learning approaches relative to GBDT methods [28].

3.3.2. Training Objective and Target Handling

Because we predict continuous-valued targets (tax-avoidance proxies), all models are trained in a regression setting. Since the targets $Y \in \{\mathrm{CETR}, \mathrm{GETR}, \mathrm{TSTA}, \mathrm{TSDA}\}$ may differ in scale and distribution, our default implementation trains a separate single-output model per target ("one target per model"), allowing a direct comparison of how model choice and feature-set variants affect predictive performance for each target. We use standard squared-error objectives (MSE/RMSE) for

 doi:10.20944/preprints202602.1863.v1

both tree/boosting models and neural-network models, and report MAE, RMSE, and R^2 on the out-of-time test split defined in Section 3.2.

3.3.3. Preprocessing and Feature Protocol (Train-Only Fit)

Preprocessing and feature construction follow the leakage-prevention principles in Section 3.2. Specifically, identifier/meta variables (e.g., stock, year, and market/industry indicators) are used only to define panel keys and design data splits and are excluded from model inputs. The three input configurations—Raw-only, Derived-only, and Raw+Derived—and the feature-set variants controlling tax-proxy inclusion (FS1–FS4) follow the definitions in Section 3.1. Transformations that depend on data statistics such as scaling/normalization, outlier mitigation, and missing-value imputation are fit only on the training split and then applied unchanged to the validation and test splits.

Because tree-based GBDT models (XGBoost/LightGBM/CatBoost) are relatively insensitive to feature scaling, we keep the original scale by default, whereas for the DL model (TabTransformer) we standardize continuous variables (e.g., z-score normalization) to improve training stability and convergence. Categorical or binary indicator variables are handled in accordance with model characteristics; in TabTransformer, the categorical token-embedding pathway and the continuous-feature pathway are treated separately during training [28]. All preprocessing parameters e.g., clipping quantiles, standardization rules, and imputation schemes are fixed under a single protocol (fit on Train only) and applied consistently to Validation/Test.

3.3.4. Hyperparameter Tuning and Early Stopping

Hyperparameters are searched by fitting models on the training split and selecting configurations based on validation performance following the procedure in Section 3.2; we then report the final performance exactly once on the test split. For GBDT models, we use early stopping to mitigate overfitting and improve computational efficiency (e.g., stopping training when validation performance does not improve for a specified number of rounds). The main search dimensions include the learning rate, tree complexity (depth/number of leaves), sampling ratios (subsample/feature fraction), and regularization terms (L_1/L_2-type) [24–26]. For TabTransformer, we tune the embedding/hidden dimensions, the number of layers, the number of attention heads, dropout, and optimizer settings (AdamW, learning rate, weight decay), and apply validation-based early stopping [28].

3.3.5. Reproducibility

All experiments are conducted using the same data splits (Table 3) and the same feature-set definitions (Section 3.1). We also specify the underlying software libraries used for each model so that experiments can be reproduced under identical conditions. The hyperparameter search space and training configuration are summarized in Table 4. Early stopping and best-iteration/checkpoint selection are performed based on out-of-time validation performance ($t = 2021 \rightarrow 2022$). The validation split is not used for parameter updates; it is used only for evaluation to select model and training settings.

Table 4. Hyperparameter search space and training configuration.

Model	Objective (Loss)	Key hyperparameters (search space)	Early stopping, Training notes
XGBoost [24]	Regression (squared error)	n_estimators: 500, 1000, 2000 learning_rate: 0.01, 0.05, 0.1 max_depth: 3, 5, 7	Early stopping on out-of-time Validation (t=2021→2022) Use best_iteration for final fit

 doi:10.20944/preprints202602.1863.v1

		min_child_weight: 1, 5, 10 subsample: 0.6, 0.8, 1.0 colsample_bytree: 0.6, 0.8, 1.0 reg_lambda(L2): 0, 1, 10 reg_alpha(L1): 0, 0.1, 1	
LightGBM [25]	Regression (L2)	n_estimators: 1000, 3000, 8000 learning_rate: 0.01, 0.05, 0.1 num_leaves: 31, 63, 127 max_depth: -1, 6, 10 min_data_in_leaf: 20, 50, 100 feature_fraction: 0.7, 0.9, 1.0 bagging_fraction: 0.7, 0.9, 1.0 lambda_l2: 0, 1, 10 lambda_l1: 0, 0.1, 1	Early stopping on out-of-time Validation (t=2021→2022) Use best_iteration
CatBoost [26]	Regression (RMSE)	iterations: 2000, 5000, 8000 learning_rate: 0.01, 0.05, 0.1 depth: 4, 6, 8, 10 l2_leaf_reg: 1, 3, 10 random_strength: 0, 1, 5 bagging_temperature: 0, 1, 5 rsm: 0.7, 0.9, 1.0	Early stopping on out-of-time Validation (t=2021→2022) Categorical/binary features handled by CatBoost mechanism
TabTransformer [28]	Regression (MSE)	n_layers: 2, 4, 6 d_model (embedding/hidden): 64, 128, 256 n_heads: 4, 8 dropout: 0.0, 0.1, 0.2 mlp_hidden: 128, 256, 512 batch_size: 256, 512 learning_rate: 1e-4, 3e-4, 1e-3 weight_decay: 0, 1e-5, 1e-4	AdamW optimizer Early stopping on out-of-time Validation (t=2021→2022) Continuous features standardized (fit on Train only)

4. Experiments

This section empirically evaluates the proposed leakage-free, one-year-ahead forecasting task on the KoTaP (2011–2024) firm–year panel. Specifically, given firm i's covariates $X_{i,t}$ observed in fiscal year t, we predict next-year tax-avoidance proxies $Y_{i,t+1} \in$ {CETR,GETR,TSTA,TSDA} (Section 3). To examine how feature construction and model choice affect predictive performance under limited-sample panel setting, we compare three input configurations—Raw-only, Derived-only, and Raw+Derived—under the same temporal split and evaluation protocol. Model selection (including hyperparameter tuning) is conducted on the temporally separated validation split, and final performance is reported on the held-out test split to reflect realistic deployment where future

 doi:10.20944/preprints202602.1863.v1

outcomes are unavailable at training time. As evaluation metrics, we report RMSE, MAE, and R^2; RMSE/MAE quantify absolute prediction errors, while R^2 is reported for completeness but can be difficult to interpret when the target variance is small.

4.1. Machine Learning Results

FS1 (tax-agnostic) baselines: Table 5 shows that gradient-boosting models provide strong, stable baselines under leakage-free evaluation. Overall, LightGBM with Raw+Derived achieves the lowest RMSE for GETR, TSTA, and TSDA, while CatBoost (Raw+Derived) performs best for CETR. Across targets, Raw+Derived tends to be the most robust input configuration suggesting that derived ratios and lagged indicators complement raw financial statement items. For ETR-type targets (CETR/GETR), out-of-sample R^2can be near zero or negative even when RMSE/MAE are relatively low, indicating that error-based metrics (RMSE/MAE) are more reliable for model comparison in this setting.

Table 5. ML baselines across Raw/Derived/Raw+Derived.

Target	Model	Raw-only			Derived-only			Raw+Derived		
		RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2
CETR	XGBoost	0.234	0.185	-0.114	0.231	0.183	-0.093	0.231	0.183	-0.087
	LightGBM	0.232	0.185	-0.095	0.233	0.185	-0.111	0.231	0.185	-0.094
	CatBoost	0.229	0.181	-0.068	0.230	0.182	-0.078	0.228	0.18	-0.064
GETR	XGBoost	0.134	0.091	-0.02	0.136	0.094	-0.042	0.135	0.094	-0.031
	LightGBM	0.134	0.092	-0.017	0.135	0.093	-0.023	0.134	0.093	-0.014
	CatBoost	0.134	0.091	-0.013	0.136	0.094	-0.042	0.135	0.092	-0.027
TSTA	XGBoost	0.231	0.141	0.282	0.235	0.144	0.258	0.224	0.136	0.327
	LightGBM	0.230	0.141	0.289	0.238	0.145	0.239	0.221	0.134	0.345
	CatBoost	0.240	0.150	0.231	0.241	0.149	0.22	0.229	0.14	0.295
TSDA	XGBoost	0.237	0.140	0.273	0.239	0.142	0.259	0.23	0.133	0.314
	LightGBM	0.228	0.135	0.326	0.238	0.141	0.268	0.223	0.131	0.357
	CatBoost	0.241	0.146	0.185	0.241	0.143	0.191	0.232	0.134	0.252

FS2–FS4 (tax-history augmentation) on a strong ML baseline: To isolate the effect of tax-history signals while controlling for model choice and reducing computational overhead, we evaluate FS2–FS4 using LightGBM under Raw+Derived, which performs strongly in FS1 for three of the four targets. Tables 6–8 summarize the results. Adding auxiliary tax aggregates (FS2, Table 6) yields modest improvements for CETR/GETR and little change for TSTA/TSDA. In contrast, including lagged target proxies (FS3, Table 7) leads to substantial error reductions for TSTA/TSDA and markedly higher R^2, reflecting strong temporal persistence in these proxies when past values are available at prediction time. The combined variant (FS4, Table 8) provides the best or comparable performance overall, indicating that tax-history features can materially strengthen screening accuracy when they are legitimately observable at time t.

Table 6. (FS2): LightGBM + Raw+Derived with auxiliary tax aggregates.

Target	RMSE	MAE	R2	Features	ΔRMSE vs FS1
CETR	0.216	0.160	0.052	54	-0.016

 doi:10.20944/preprints202602.1863.v1

GETR	0.130	0.086	0.053	54	-0.005
TSTA	0.223	0.132	0.332	54	0.002
TSDA	0.222	0.130	0.358	54	0.000

Table 7. (FS3): LightGBM + Raw+Derived with lagged target proxies Y_t.

Target	RMSE	MAE	R2	Features	ΔRMSE vs FS1
CETR	0.216	0.160	0.048	48	-0.015
GETR	0.128	0.084	0.082	48	-0.007
TSTA	0.108	0.065	0.843	48	-0.113
TSDA	0.104	0.060	0.859	48	-0.118

Table 8. (FS4): LightGBM + Raw+Derived with auxiliary + lagged targets.

Target	RMSE	MAE	R2	Features	ΔRMSE vs FS1
CETR	0.214	0.158	0.061	58	-0.017
GETR	0.128	0.084	0.082	58	-0.007
TSTA	0.107	0.065	0.846	58	-0.114
TSDA	0.102	0.060	0.865	58	-0.120

4.2. Deep Learning Results

This subsection reports deep-learning results using TabTransformer, a Transformer-based model for tabular prediction that learns contextualized feature representations via self-attention [28]. We follow the same leakage-free $t \rightarrow t+1$ forecasting protocol and evaluation metrics as in Section 4.1 and first present the FS1 (tax-agnostic) results across the three input configurations (Raw-only, Derived-only, Raw+Derived). We then analyze how augmenting inputs with tax-history signals (FS2–FS4) affects performance under the same model, focusing on the Raw+Derived setting.

FS1 (tax-agnostic) performance and training dynamics: Table 9 summarizes test performance of TabTransformer under FS1 across feature configurations. To complement the final scores, Figure 1 visualizes validation RMSE as a function of training epoch for each target. Overall, TabTransformer exhibits rapid convergence within 10–20 epochs for all targets, after which improvements saturate. For CETR, validation RMSE decreases sharply up to around 20 epochs and then slightly worsens by 30 epochs for some feature sets, suggesting mild overfitting and motivating validation-based checkpoint selection or early stopping (Figure 1). For GETR, the validation curves generally continue to improve marginally as training proceeds, indicating a more stable optimization trajectory. For TSTA/TSDA, Raw-only remains consistently worse, while Raw+Derived (and, to a lesser extent, Derived-only) shows clear gains and steadier convergence, implying that combining raw accounts with derived ratios/lag indicators provides a more informative representation for these targets (Figure 1). Taken together, these trends support using a fixed epoch budget with validation-based checkpoint selection, and they reinforce the feature-configuration effects observed in Table 9.

 doi:10.20944/preprints202602.1863.v1

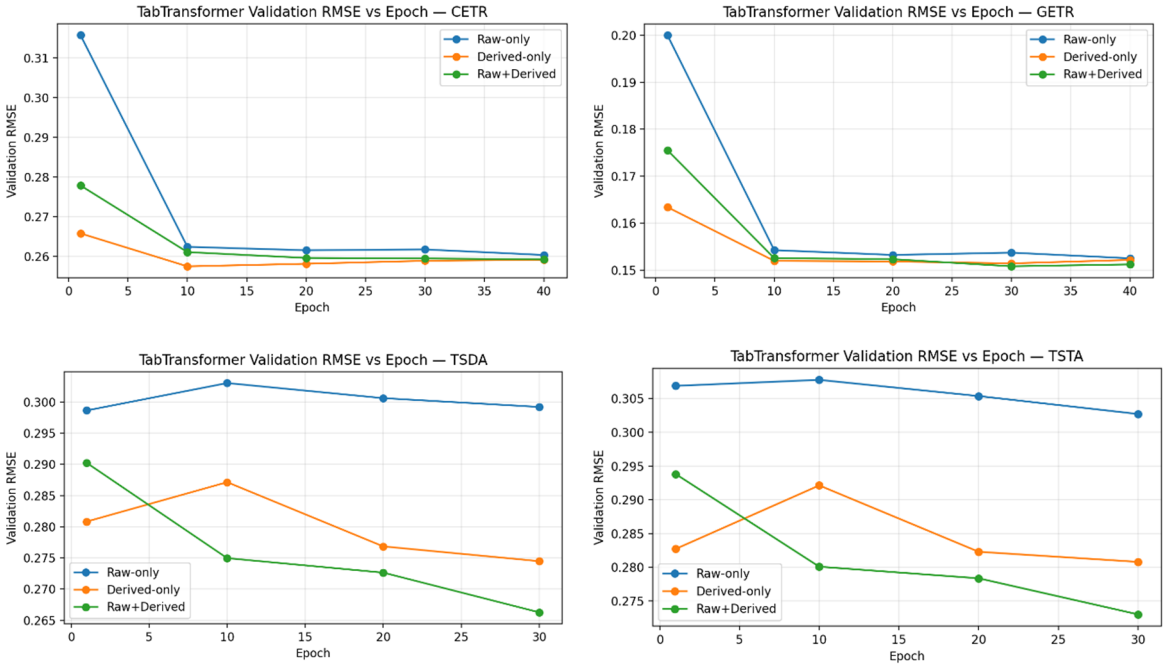

Figure 1. Validation RMSE versus training epoch for TabTransformer under FS1 (tax-agnostic) across three feature configurations (Raw-only, Derived-only, Raw+Derived) and four targets (CETR, GETR, TSTA, TSDA).

Table 9. TabTransformer results under FS1 (tax-agnostic) across feature configurations.

Target	Raw-only			Derived-only			Raw+Derived		
	RMSE	MAE	R2	RMSE	MAE	R2	RMSE	MAE	R2
CETR	0.235	0.184	-0.126	0.232	0.185	-0.095	0.233	0.184	-0.105
GETR	0.139	0.093	-0.088	0.139	0.096	-0.087	0.14	0.096	-0.105
TSTA	0.287	0.165	-0.108	0.272	0.161	0.008	0.265	0.158	0.055
TSDA	0.291	0.165	-0.095	0.274	0.159	0.029	0.267	0.155	0.074

Effect of tax-history augmentation (FS2–FS4): To isolate the value of tax-history signals under a deep tabular learner, we evaluate FS2–FS4 under Raw+Derived and report results in Tables 10–12. Adding auxiliary tax aggregates (FS2, Table 10) yields modest improvements for CETR/GETR, while changes for TSTA/TSDA are limited. In contrast, incorporating lagged target proxies (FS3, Table 11) produces substantial error reductions for TSTA/TSDA and large increases in R^2, reflecting strong temporal persistence when prior-year proxy values are available at prediction time. The combined setting (FS4, Table 12) maintains these large gains and provides the best or comparable performance for CETR/GETR as well. These ablations indicate that tax-history features, particularly lagged proxies, can materially enhance screening accuracy, and they should be interpreted in light of practical observability at time t (e.g., reporting/filing lags and availability constraints).

Table 10. TabTransformer (Raw+Derived) with FS2 (auxiliary tax aggregates).

Target	RMSE	MAE	R2	Features	ΔRMSE vs FS1
CETR	0.219	0.165	0.018	54	-0.012
GETR	0.135	0.09	-0.026	54	-0.005
TSTA	0.273	0.161	-0.004	54	0.004
TSDA	0.275	0.16	0.016	54	0.003

 doi:10.20944/preprints202602.1863.v1

Table 11. TabTransformer (Raw+Derived) with FS3 (lagged target proxies Y_t).

Target	RMSE	MAE	R2	Features	ΔRMSE vs FS1
CETR	0.221	0.165	-0.001	48	-0.009
GETR	0.134	0.088	-0.01	48	-0.006
TSTA	0.109	0.073	0.841	48	-0.161
TSDA	0.106	0.069	0.855	48	-0.167

Table 12. TabTransformer (Raw+Derived) with FS4 (auxiliary + lagged targets).

Target	RMSE	MAE	R2	Features	ΔRMSE vs FS1
CETR	0.220	0.158	0.011	58	-0.011
GETR	0.133	0.087	0.007	58	-0.007
TSTA	0.112	0.074	0.833	58	-0.158
TSDA	0.108	0.07	0.848	58	-0.165

4.3. Discussion

This section discusses the practical feasibility of building a prediction-based tax-avoidance risk-screening system in light of the experimental results and summarizes deployment-oriented recommendations on model and feature-configuration choices. We also highlight interpretive caveats and operational considerations when tax-history variables are included as inputs, as in FS2–FS4.

By formulating a leakage-free $t \rightarrow t+1$ forecasting task and comparing ML/DL models under the same splits and protocol, this study shows that non-trivial predictive signals for next-year tax-avoidance proxies can be extracted even in a firm–year panel setting (Table 5, Table 9). In particular, for indicators such as TSTA/TSDA, whose underlying mechanisms may exhibit persistence, models using Raw+Derived features tend to achieve consistently lower errors and meaningful explanatory power. This suggests that, in practice, a "screening (signal-detection)" scenario—ranking firms by a risk score and selecting a high-risk subset—can be a valid operational approach.

From the perspective of model selection, tree-based gradient boosting serves as a robust baseline in tabular settings with limited samples. Under FS1 (no tax-proxy inputs), LightGBM (Raw+Derived) performs strongly overall on GETR/TSTA/TSDA, while CatBoost yields slightly lower error for CETR (Table 5). Accordingly, for practical deployment, a reasonable strategy is to adopt LightGBM as a default first-stage scorer when a single model is preferred for simplicity, and to operate target-specific best-performing models when certain targets favor a different method (e.g., CETR–CatBoost; others–LightGBM). The deep-learning baseline (TabTransformer) converges relatively quickly (Figure 1) and provides a representative tabular DL comparator; however, in the current setting, boosting methods are more competitive, especially for TSTA/TSDA (Table 5 vs. Table 9). Thus, from a system-building standpoint, prioritizing ML-based models for initial deployment and using deep learning as a secondary comparator or extension (e.g., for pretraining, transfer learning, or interpretability-driven requirements) is a practical choice. Nevertheless, advances in generative models for financial time series (e.g., [41]) suggest that structure-preserving synthetic data augmentation may alleviate data scarcity and improve the practicality of adopting DL-based approaches.

Regarding feature-set design, Raw+Derived emerges as the most stable choice across the Raw-only, Derived-only, and Raw+Derived comparisons (Table 5, Table 9), implying that raw accounts (scale information) and derived indicators (ratios, growth, and lag information) provide complementary signals. Therefore, for implementation, we recommend including both core raw variables such as assets, liabilities, sales, and cash flows and derived features such as profitability/stability/growth ratios and lag features. Meanwhile, the FS2–FS4 experiments disentangle how "tax-history information" changes predictive performance. FS2, which includes

 doi:10.20944/preprints202602.1863.v1

auxiliary aggregated tax proxies, provides modest improvements for CETR/GETR, whereas FS3/FS4, which include lagged values of tax-avoidance proxies (target lags), yield substantial gains, especially for TSTA/TSDA (Table 6–8, Table 10–12). Operationally, this can be interpreted as a two-stage strategy: (1) a general-purpose screening setup (FS1; no tax proxies) that can be applied even when tax-history information is unavailable, and (2) an enhanced history-aware screening setup (FS3/FS4) that strengthens performance when prior-year proxies are legitimately observable.

Because our targets are not the "true" underlying tax-avoidance behavior but proxy measures defined from accounting and disclosure data [30,31], predictive performance should be interpreted as risk signal detection rather than a direct determination of tax avoidance. Moreover, the lagged proxy values used in FS3/FS4 must be available at the prediction time t in real systems (e.g., subject to disclosure or filing delays); if this assumption does not hold, the observed performance gains may not generalize. Finally, to establish the practical utility of a screening system, additional evaluations beyond RMSE/MAE are needed, including ranking-based performance (e.g., top-k accuracy and hit rates for alerts), calibration, robustness under distribution shifts (across years/markets/industries), and explainability (feature-attribution analysis and case-based explanations).

5. Conclusions

This paper investigated leakage-free, one-year-ahead forecasting of corporate tax-avoidance proxies on the KoTaP (2011–2024) firm–year panel. By formulating an ex-ante $t \rightarrow t+1$ prediction setting and evaluating multiple feature configurations and benchmark models under a consistent temporal protocol, we provided an empirical basis for building a practical risk-screening system. Across tax-agnostic inputs (FS1), gradient-boosting methods with Raw+Derived features delivered the most robust performance, while TabTransformer provided a deep tabular baseline with stable training dynamics. Overall, gradient boosting remained more competitive for TSTA/TSDA in our setting. Additional ablations (FS2–FS4) showed that tax-history signals, especially lagged proxy values when legitimately observable, can substantially improve screening accuracy, highlighting the importance of feature availability assumptions in deployment. Future work will extend the evaluation to ranking- and decision-oriented metrics, robustness under distribution shifts, and data-efficient deep learning via domain-aware synthetic data augmentation.

References

1. Financial Services Commission (FSC). Korea Accounting Standards Board Announces Korean Translation of International Financial Reporting Standards. Press Release, 24 December 2007. Available online: https://www.fsc.go.kr/eng/pr010101/21771 (accessed on 22 January 2026).
2. IFRS Foundation. IFRS Adoption and Implementation in Korea, and the Lessons Learned (IFRS Country Report). 2013. Available online: https://www.ifrs.org/content/dam/ifrs/meetings/2013/june/ifrs-advisory-council/ap2b-adoption-and-implementation-in-korea.pdf (accessed on 22 January 2026).
3. Deloitte IAS Plus. Korea—The Korean Experience with IFRS Adoption. 19 March 2013. Available online: https://www.iasplus.com/en/news/2013/03/korea (accessed on 22 January 2026).
4. Financial Supervisory Service (FSS). About DART | How Does DART Work? Available online: https://englishdart.fss.or.kr/about/engAbout1.do (accessed on 22 January 2026).
5. XBRL International. Jurisdictions—XBRL Korea (DART Filing). Available online: https://www.xbrl.org/the-consortium/about/jurisdictions/ (accessed on 22 January 2026).
6. Shin, H.; Oh, H. Mandatory Adoption of IFRS and Earnings Transparency in Korea. J. Appl. Bus. Res. 2017, 33, 1129–1138. https://doi.org/10.19030/jabr.v33i6.10050.
7. Hong, J.Y. Significance and Challenges of Expanding Mandatory XBRL-Based Financial Disclosure for Korean Firms. Capital Market Focus (KCMI) 2023, No. 2023-10. Korea Capital Market Institute, 15 May 2023. Available online: https://www.kcmi.re.kr/publications/pub_detail_view?cno=6119&syear=2023&zcd=002001016&zno=1722 (accessed on 22 January 2026).

 doi:10.20944/preprints202602.1863.v1

8. Jung, D.J.; Hur, J.A.; Jung, A.R. The Precondition of Benefits from IFRS Adoption: Financial Statement Comparability. J. Asian Financ. Econ. Bus. 2020, 7, 255–265. https://doi.org/10.13106/JAFEB.2020.VOL7.NO12.255.
9. Na, H.; Song, W.; Han, S.; Jo, D.; Myung, S.; Kim, H. KoTaP: A Panel Dataset for Corporate Tax Avoidance, Performance, and Governance in Korea. Scientific Data 2026. doi:10.1038/s41597-026-06722-5.
10. Na, H.; Kim, H.; Song, W.; Myung, S.; Han, S.; Jo, D. KoTaP: A Panel Dataset for Corporate Tax Avoidance, Performance, and Governance in Korea (2011–2024). Version v2. Zenodo, 2025. doi:10.5281/zenodo.17149808.
11. Hyndman, R.J.; Khandakar, Y. Automatic Time Series Forecasting: The forecast Package for R. J. Stat. Softw. 2008, 27, 1–22. https://doi.org/10.18637/jss.v027.i03.
12. Hyndman, R.J.; Koehler, A.B.; Snyder, R.D.; Grose, S. A State Space Framework for Automatic Forecasting Using Exponential Smoothing Methods. Int. J. Forecast. 2002, 18, 439–454. https://doi.org/10.1016/S0169-2070(01)00110-8.
13. Arellano, M.; Bond, S. Some Tests of Specification for Panel Data: Monte Carlo Evidence and an Application to Employment Equations. Rev. Econ. Stud. 1991, 58, 277–297. https://doi.org/10.2307/2297968.
14. Salinas, D.; Flunkert, V.; Gasthaus, J.; Januschowski, T. DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks. Int. J. Forecast. 2020, 36, 1181–1191. https://doi.org/10.1016/j.ijforecast.2019.07.001.
15. Rangapuram, S.S.; Seeger, M.; Gasthaus, J.; Stella, L.; Wang, Y.; Januschowski, T. Deep State Space Models for Time Series Forecasting. In Advances in Neural Information Processing Systems; 2018. Available online: https://papers.nips.cc/paper/8004-deep-state-space-models-for-time-series-forecasting (accessed on 22 January 2026).
16. Lai, G.; Chang, W.-C.; Yang, Y.; Liu, H. Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '18), Ann Arbor, MI, USA, 8–12 July 2018; pp. 95–104. https://doi.org/10.1145/3209978.3210006.
17. Lim, B.; Arık, S.Ö.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for Interpretable Multi-Horizon Time Series Forecasting. Int. J. Forecast. 2021, 37, 1748–1764. https://doi.org/10.1016/j.ijforecast.2021.03.012.
18. Chronopoulos, I.; Chrysikou, K.; Kapetanios, G.; Mitchell, J.; Raftapostolos, A. Deep Neural Network Estimation in Panel Data Models. Federal Reserve Bank of Cleveland Working Paper No. 23-15, 2023. Available online: https://arxiv.org/abs/2305.19921 (accessed on 22 January 2026).
19. Yang, B.; Long, W.; Cai, Z. Machine Learning Based Panel Data Models. Working Papers Series in Theoretical and Applied Economics 202402, University of Kansas, 2024 (revised January 2024). Available online: https://kuwpaper.ku.edu/2024Papers/202402.pdf (accessed on 22 January 2026).
20. Tibshirani, R. Regression Shrinkage and Selection via the Lasso. J. R. Stat. Soc. Ser. B 1996, 58, 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.
21. Zou, H.; Hastie, T. Regularization and Variable Selection via the Elastic Net. J. R. Stat. Soc. Ser. B 2005, 67, 301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x.
22. Breiman, L. Random Forests. Mach. Learn. 2001, 45, 5–32. https://doi.org/10.1023/A:1010933404324.
23. Friedman, J.H. Greedy Function Approximation: A Gradient Boosting Machine. Ann. Stat. 2001, 29, 1189–1232. https://doi.org/10.1214/aos/1013203451.
24. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16), San Francisco, CA, USA, 13–17 August 2016; pp. 785–794. https://doi.org/10.1145/2939672.2939785.
25. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In Advances in Neural Information Processing Systems 30 (NeurIPS 2017); 2017; pp. 3146–3154. Available online: https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html (accessed on 22 January 2026).
26. Dorogush, A.V.; Ershov, V.; Gulin, A. CatBoost: Gradient Boosting with Categorical Features Support. arXiv 2018, arXiv:1810.11363. https://doi.org/10.48550/arXiv.1810.11363.

 doi:10.20944/preprints202602.1863.v1

27. Arik, S.Ö.; Pfister, T. TabNet: Attentive Interpretable Tabular Learning. In Proceedings of the AAAI Conference on Artificial Intelligence 2021, 35, 6679–6687. Available online: https://ojs.aaai.org/index.php/AAAI/article/view/16826 (accessed on 22 January 2026).
28. Huang, X.; Khetan, A.; Cvitkovic, M.; Karnin, Z. TabTransformer: Tabular Data Modeling Using Contextual Embeddings. arXiv 2020, arXiv:2012.06678. https://doi.org/10.48550/arXiv.2012.06678.
29. Gorishniy, Y.; Rubachev, I.; Khrulkov, V.; Babenko, A. Revisiting Deep Learning Models for Tabular Data. In Advances in Neural Information Processing Systems 34 (NeurIPS 2021); 2021; pp. 18932–18943. Available online: https://dl.acm.org/doi/10.5555/3540261.3541708 (accessed on 22 January 2026).
30. Hanlon, M.; Heitzman, S. A Review of Tax Research. J. Account. Econ. 2010, 50, 127–178. https://doi.org/10.1016/j.jacceco.2010.09.002.
31. Frank, M.M.; Lynch, L.J.; Rego, S.O. Tax Reporting Aggressiveness and Its Relation to Aggressive Financial Reporting. Account. Rev. 2009, 84, 467–496. https://doi.org/10.2308/accr.2009.84.2.467.
32. Guenther, D.A.; Peterson, K.; Searcy, J.; Williams, B.M. How Useful Are Tax Disclosures in Predicting Effective Tax Rates? A Machine Learning Approach. Account. Rev. 2023, 98, 297–322. https://doi.org/10.2308/TAR-2021-0398.
33. Borrotti, M.; Rabasco, M.; Santoro, A. Using Accounting Information to Predict Aggressive Tax Location Decisions by European Groups. Econ. Syst. 2023, 47, 101090. https://doi.org/10.1016/j.ecosys.2023.101090.
34. Rahman, R.A.; Masrom, S.; Omar, N.; Zakaria, M. An Application of Machine Learning on Corporate Tax Avoidance Detection Model. IAES Int. J. Artif. Intell. 2020, 9, 721–725. https://doi.org/10.11591/ijai.v9.i4.pp721-725.
35. Beneish, M.D. The Detection of Earnings Manipulation. Financ. Anal. J. 1999, 55, 24–36. https://doi.org/10.2469/faj.v55.n5.2296.
36. Dechow, P.M.; Ge, W.; Larson, C.R.; Sloan, R.G. Predicting Material Accounting Misstatements. Contemp. Account. Res. 2011, 28, 17–82. https://doi.org/10.1111/j.1911-3846.2010.01041.x.
37. Altman, E.I. Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy. J. Financ. 1968, 23, 589–609. https://doi.org/10.1111/j.1540-6261.1968.tb00843.x.
38. Cecchini, M.; Aytug, H.; Koehler, G.J.; Pathak, P. Making Words Work: Using Financial Text as a Predictor of Financial Events. Decis. Support Syst. 2010, 50, 164–175. https://doi.org/10.1016/j.dss.2010.07.012.
39. Dyreng, S.D.; Hanlon, M.; Maydew, E.L. Long-Run Corporate Tax Avoidance. Account. Rev. 2008, 83, 61–82. https://doi.org/10.2308/accr.2008.83.1.61.
40. Desai, M.A.; Dharmapala, D. Corporate Tax Avoidance and High-Powered Incentives. J. Financ. Econ. 2006, 79, 145–179. https://doi.org/10.1016/j.jfineco.2005.02.002.
41. Choi, G.; et al. LFTD: Transformer-Enhanced Diffusion Model for Realistic Financial Time-Series Data Generation. Pre-prints.org 2026.