Article

Benchmarking Architectures for Churn Prediction: The Impact of Data Leakage

Neta Ben Mordechai [1], Itay Chabra [2] and Yuval Gefen [3]

[1] Affiliation 1; netabm1@gmail.com
[2] Affiliation 2; itaychabra@gmail.com
[3] Affiliation 3 yuvalgefen11@gmail.com

Abstract

Customer churn prediction is a fundamental challenge in the telecommunications industry, driving the adoption of increasingly complex Deep Learning (DL) models. Recent literature, specifically the ChurnNet architecture, reports near-perfect accuracy using 1D-Convolutional Neural Networks (1D-CNN) combined with aggressive oversampling. However, such performance raises concerns regarding potential data leakage-specifically the application of Synthetic Minority Over-sampling Technique (SMOTE) prior to data splitting - and the suitability of convolution for non-spatial tabular data. This study critically re-evaluates these claims by implementing a rigorous cross-validation pipeline where data scaling and oversampling are strictly confined to training folds to prevent synthetic data leakage. We benchmark the corrected 1D-CNN against more theoretically suitable architectures for tabular data, including a Deep Multi-Layer Perceptron (MLP), FT-Transformer, Gradient Boosting (XGBoost, LightGBM), and TabNet. To assess true generalization, we extend the evaluation to an independent, unseen dataset (Bank Customer Churn) not used in the original studies. Experimental results confirm that originally reported "perfect" accuracies were artifacts of data leakage. under valid testing protocols, the 1D-CNN significantly underperforms compared to tree-based ensembles and Transformer architectures. We demonstrate that proper validation protocols and architectural appropriateness are far more critical than model complexity. Finally, we integrate Explainable AI (XAI) to transition from "black-box" predictions to actionable business insights, establishing a realistic baseline for churn prediction performance.
To facilitate reproducibility and further research, the complete source code, experimental notebooks, and the dataset preprocessing pipeline are publicly available at: GitHub Link.

Keywords: Customer Churn Prediction; Data Leakage; Deep Learning; 1D-CNN; XGBoost; SMOTE; Explainable AI (XAI); Tabular Data.

1. Introduction

In the modern data-driven economy, customer retention has emerged as a critical challenge across diverse sectors, ranging from Telecommunications and Finance to Software-as-a-Service (SaaS) and Retail. As markets become increasingly saturated and competitive, the "Subscription Economy" model has shifted the business focus from aggressive acquisition to maximizing Customer Lifetime Value (CLV). It is a well-established economic reality that acquiring a new customer is estimated to be 5 to 25 times more costly than retaining an existing one [9], making accurate Customer Churn Prediction (CCP) a critical strategic priority [8]. Consequently, identifying at-risk customers early enables organizations to deploy targeted retention interventions, thereby preserving revenue streams and brand stability.

Historically, CCP was predominantly addressed using traditional Machine Learning (ML) algorithms. Classifiers such as Logistic Regression, Support Vector Machines (SVM), and Random Forests became industry standards due to their interpretability and ease of deployment. However, these methods often struggle to capture complex, non-linear interactions within high-dimensional and heterogeneous data without extensive manual feature engineering. To overcome these limitations, recent research has pivoted toward Deep Learning (DL) architectures, which promise automated feature extraction and superior predictive performance on large-scale datasets.

A prominent example of this trend is the recently proposed "ChurnNet" architecture by Saha et al. [1], which integrates 1D-Convolutional Neural Networks (1D-CNN) with residual blocks and attention mechanisms. This study reported remarkable accuracies exceeding 97% on benchmark datasets by utilizing the Synthetic Minority Over-sampling Technique (SMOTE) to address class imbalance. However, such performance claims warrant skepticism on two fronts. First, Convolutional Neural Networks are designed for data with inherent spatial locality (e.g., image pixels or audio waveforms). Applying them to permutation-invariant tabular data where the order of columns (e.g., 'Age' vs. 'Credit Score') is arbitrary-raises theoretical concerns regarding architectural suitability, as noted by Grinsztajn et al. [3]. Second, extremely high metrics in churn prediction are frequently symptomatic of "data leakage," specifically the application of oversampling techniques prior to cross-validation splitting. This methodological oversight allows the model to train on synthetic variations of test samples, creating an illusion of accuracy that fails to materialize in production environments.

In this paper, we challenge the validity of such high-performance claims through a rigorous re-evaluation and present a robust framework for churn prediction. We hypothesize that the reported efficacy of spatial DL models in prior studies [1] is inflated by improper validation protocols. To demonstrate this, we first reproduce the 1D-CNN architecture within a strict pipeline where data scaling and SMOTE are confined exclusively to training folds. Upon observing that the performance of the corrected 1D-CNN drops drastically when leakage is removed, we shift our focus to benchmarking architectures that are theoretically optimized for tabular data. These include Gradient Boosting (XGBoost [6], LightGBM [7]) and modern Deep Learning approaches like the Feature Tokenizer Transformer (FT-Transformer) [4], TabNet [5], and Attention-based MLPs.

The contributions of this work are fourfold:

1. **Identification of Methodological Errors:** We identify critical flaws in prior validation protocols specifically the presence of data leakage and construct a rigorous, hygiene-focused pipeline to reveal the true, uninflated performance of ChurnNet.

2. **Architectural Benchmarking:** Having established the limitations of 1D-CNNs for this task, we conduct a comprehensive comparison of tabular-specific architectures (XGBoost [6], LightGBM [7], FT-Transformer [4], TabNet [5]) to establish a valid state-of-the-art baseline.

3. **External Validation:** We evaluate model generalization across two distinct domains: the original Telecommunications dataset and an independent Bank

Customer Churn dataset. This comparison highlights that predictive performance is heavily dependent on feature quality, with realistic F1-scores ranging from ~85% (Telecom) to ~62% (Bank) depending on the available signal.

4. **Explainability:** We integrate Explainable AI (XAI) techniques [2] to provide actionable transparency, transitioning from "black-box" accuracy to business value.

Our principal conclusion is that strict data hygiene is non-negotiable for reproducible science. Furthermore, we demonstrate that while modern Deep Learning (specifically FT-Transformers) can compete with Gradient Boosting, the "ceiling" of performance is dictated by the intrinsic quality of the dataset features rather than model complexity.

2. Literature Review

Customer churn prediction involves identifying customers likely to discontinue their service using historical usage and demographic data. Early research predominantly utilized traditional Machine Learning models. Algorithms such as Logistic Regression, Random Forest, and Support Vector Machines (SVM) were widely adopted due to their interpretability. However, the field has largely been dominated by Gradient Boosted Decision Trees (GBDT) due to their superior performance on tabular data. Systems like XGBoost [6] and LightGBM [7] have become the industry standard, offering scalability and state-of-the-art accuracy by handling non-linear interactions and missing values more effectively than traditional statistical methods.

A critical factor in model performance is the nature of the dataset attributes, which vary significantly across industries. In the Telecommunications domain, the data is heavily weighted towards granular usage metrics, capturing specific behavioral patterns through features like total day, evening, night, and international usage (broken down by minutes, calls, and charges), alongside service indicators such as International Plan, Voicemail Plan, and the critical Number of Customer Service Calls. In contrast, the Banking dataset integrates traditional demographic and financial attributes such as CreditScore, Balance, EstimatedSalary, and Tenure with specific customer engagement and satisfaction metrics, including Satisfaction Score, Points Earned, and Card Type. The heterogeneity of these features mixing categorical variables with continuous numerical values without inherent spatial ordering creates a distinct challenge for predictive modeling.

Despite the dominance of tree-based models, recent literature has explored the potential of Deep Learning (DL) to automate feature extraction. A notable contribution is the ChurnNet architecture proposed by Saha et al. [1], which employs 1D-Convolutional Neural Networks (1D-CNN) and aggressive oversampling to achieve near-perfect accuracy. However, this approach is theoretically contested. Grinsztajn et al. [3] conducted an extensive benchmark demonstrating that tree-based models consistently outperform Deep Learning on tabular datasets. They argue that DL architectures designed for spatial data (like CNNs) fail to generalize on tabular data because features such as "Age" and "Balance" lack the spatial locality inherent in images or audio.

To bridge the gap between Deep Learning and tabular data, specialized architectures have been developed. TabNet [5] utilizes a sequential attention mechanism to select features at each decision step, mimicking the behavior of decision trees within a neural network framework. Similarly, the Feature Tokenizer Transformer (FT-Transformer) [4] adapts the Transformer architecture for tabular data by transforming features into embeddings and processing them via self-attention layers. These models represent a more theoretically sound application of Deep Learning to churn prediction compared to spatial CNNs.

Finally, as model complexity increases, interpretability becomes a challenge. Salih et al. [2] emphasize the necessity of Explainable AI (XAI) methods, such as SHAP and LIME, to ensure that high-performing "black-box" models remain transparent and actionable for business stakeholders. This study aims to investigate these discrepancies by contrasting the ChurnNet approach [1] with these tabular-specific architectures [4, 5, 6, 7] under a controlled, leakage-free environment.

3. Materials and Methods

3.1 Datasets

To evaluate the proposed hypothesis and benchmark model generalization, we utilized two distinct datasets from different industrial domains **(see Appendix for full feature descriptions):**

Churn-data-UCI (Telecommunications): Employed to reproduce and critique the original ChurnNet study. This dataset consists of 5,000 samples with 20 raw attributes and an 85.9% non-churn rate. It serves as the baseline for testing the impact of data leakage on 1D-CNN performance**.**

Bank Customer Churn (Finance): To test architectural robustness on unseen data, we introduced this dataset containing 10,000 samples from a European bank. After preprocessing, it exhibits a class imbalance of approximately 20.4% churners (Exited). This dataset allows us to verify if the superior performance of tree-based models holds across different feature distributions (see Figure 1).

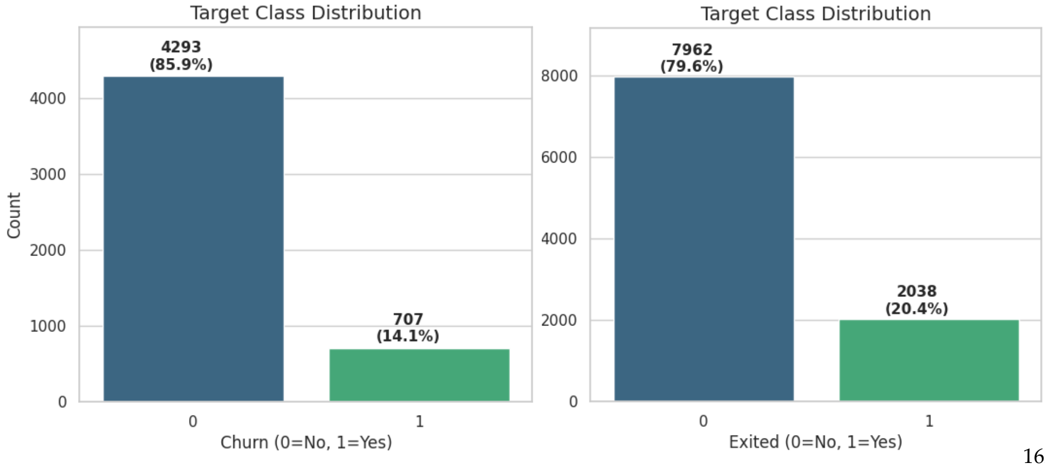


Figure 1. Comparative Class Distributions. (Left) The Churn-data-UCI (Telecom) dataset exhibits severe imbalance with an 85.9% non-churn rate. (Right) The Bank Customer Churn dataset shows moderate imbalance (20.4% churn). Both distributions necessitate the use of intervention strategies, such as Class Weights or threshold tuning, to prevent model bias.

3.2 Model Architectures

We selected a spectrum of architectures to compare the efficacy of spatial Deep Learning approaches against models theoretically optimized for tabular data. The selection rationale is grounded in recent benchmarks [3, 6, 7] demonstrating that while Deep Learning excels in perceptual tasks, specialized tree-based and transformer models currently constitute the state-of-the-art for structured data.

A. The Control Model

- ChurnNet (Reproduction) [1]: We faithfully reconstructed the architecture proposed by Saha et al., consisting of three 1D-Convolutional layers (kernel size 5, 128 filters) followed by Residual Blocks and Squeeze-and-Excitation (SE) modules. This model treats tabular features as a sequential signal, relying on the assumption of spatial locality between adjacent columns.

B. Tree-Based Baselines (Industry Standards) Gradient Boosted Decision Trees (GBDT) are currently considered the gold standard for tabular problems due to their ability to handle non-linear interactions and irregular feature spaces. We compared two leading implementations:

- **XGBoost [6]:** A scalable implementation of gradient boosting that utilizes a level-wise tree growth strategy. It employs a pre-sorted algorithm and histogram-based splitting to optimize training speed and accuracy.
- **LightGBM [7]:** Unlike XGBoost, LightGBM utilizes a leaf-wise growth strategy with Gradient-based One-Side Sampling (GOSS). This allows it to converge faster and handle larger datasets more efficiently, though it requires careful regularization to prevent overfitting on smaller datasets.

C. Deep Learning for Tabular Data To evaluate if modern Deep Learning can compete with GBDTs without assuming spatial locality, we implemented:

- **FT-Transformer [4]:** The Feature Tokenizer Transformer adapts the Transformer architecture for tabular data. Unlike CNNs, it transforms numerical and categorical features into embeddings and processes them via self-attention layers. This allows the model to capture global feature interactions dynamically without assuming a fixed spatial order.
- **TabNet [5]:** An interpretable architecture that utilizes sequential attention mechanisms to select relevant features at each decision step (instance-wise feature selection). It mimics the decision-making logic of decision trees within a differentiable Deep Learning framework.
- **Attentive MLP:** A custom Multi-Layer Perceptron augmented with an initial attention block to dynamically weigh input features before passing them to dense layers (256, 128 units), Batch Normalization, and Dropout (0.37).

3.3 Data Preprocessing

Prior to model training, raw data underwent specific preprocessing steps to ensure feature quality and compatibility.

- **Feature Selection:** Unique identifiers with no predictive power (e.g., RowNumber, CustomerId, Surname in the Bank dataset; Phone Area Code in Telecom) were removed to prevent noise.

- **Imputation:** Missing values were handled using simple mean/mode imputation to maintain data integrity.
- **Encoding:** Categorical variables were processed based on the model architecture:

  Tree-based models: Utilized One-Hot Encoding to treat categories as distinct binary features.

  Deep Learning models: Utilized Label Encoding mapped to learnable Embedding Layers, allowing the network to learn semantic relationships between categories.
- **Scaling:** Continuous numerical features were standardized (Z-score normalization) to ensure stable gradient descent convergence for the Deep Learning models.

3.4 Handling Class Imbalance

Given the significant class imbalance in both datasets (approx. 15-20% churn rate), standard training can lead to biased models that favor the majority class. We employed two distinct strategies:

1. **Synthetic Minority Over-sampling Technique (SMOTE):**
   Used exclusively for the ChurnNet reproduction to replicate the original study [1]. SMOTE generates synthetic samples for the minority class (churners) by interpolating between existing samples. For a given minority instance, SMOTE selects k nearest neighbors and creates new points along the line segments joining them. While effective, this method introduces a risk of data leakage if applied before cross-validation splitting.
2. **Inverse Class Weights:** For our proposed benchmarks (XGBoost, FT-Transformer, etc.), we utilized a cost-sensitive learning approach. Instead of generating synthetic data, we assigned higher weights to the minority class within the loss function (e.g., scale_pos_weight in XGBoost, Weighted Cross-Entropy in PyTorch). This method forces the model to penalize misclassifications of churners more heavily without introducing the noise associated with synthetic data generation in high-dimensional spaces.

3.5 The Cross-Validation Pipeline

A primary contribution of this study is the implementation of a pipeline designed to prevent data leakage. Unlike prior studies where preprocessing often occurred globally, our pipeline isolates each fold entirely, as illustrated in Figure 2

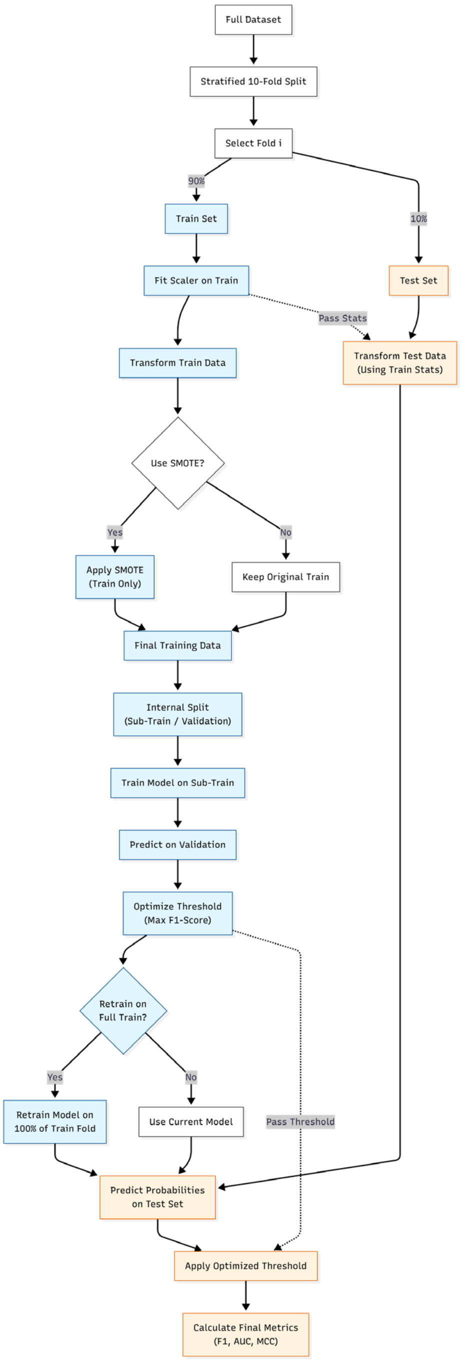


Figure 2. Illustration of the Cross-Validation Pipeline

The experimental procedure follows a Stratified 10-Fold Cross-Validation protocol. In each iteration, the dataset is partitioned into 90% Training and 10% Testing. Crucially, the pipeline enforces the following sequence:

1. **Split:** Data is separated into Train/Test folds.
2. **Fit Transformers:** Scalers and Encoders are fitted only on the Training set.
3. **Transform:** The fitted transformers are applied to the Test set.
4. **Resampling (Optional):** If SMOTE is used, it is applied only to the Training set. The Test set remains pristine and untouched.
5. **Nested Threshold Tuning:** Within the Training fold, a further subset (10%) is held out to optimize the classification threshold (moving away from the default 0.5) to maximize the F1-Score.

6. **Evaluation:** The model is evaluated on the Test fold using the optimized threshold.

3.6 Evaluation Metrics & Explainability (XAI)

Evaluation Metrics: Due to the cost asymmetry of churn (where missing a churner is costlier than a false alarm), Accuracy is a misleading metric. We prioritize the F1-Score (Class 1), which balances Precision and Recall. We also report the Matthews Correlation Coefficient (MCC) as a robust measure of quality for imbalanced datasets, and the Area Under the ROC Curve (AUC).

Explainable AI (XAI): To transition from "black-box" predictions to actionable business insights, we integrated SHAP (SHapley Additive exPlanations) [2]. SHAP values, based on cooperative game theory, quantify the marginal contribution of each feature to the model's final prediction. This allows us to validate whether the model relies on logical business drivers (e.g., "High Tenure reduces Churn") rather than spurious correlations or data leakage artifacts.

3.7 Implementation Details

All experiments were conducted using Python 3.10. To ensure reproducibility, the pipeline utilized standard open-source libraries:

- **Data Handling & Metrics:** scikit-learn (for stratification and scoring) and pandas.
- **Imbalance Handling:** imbalanced-learn (specifically for the control group SMOTE implementation).
- **Deep Learning:** PyTorch (for FT-Transformer, TabNet, and MLP implementations).
- **Gradient Boosting:** The official xgboost and lightgbm libraries.
- **Hardware Acceleration:** All models were trained on NVIDIA T4 GPUs to accelerate neural network computations.

4. Experiments and Results

This section details the experimental findings across three distinct evaluation scenarios. We first benchmark the architectures on the Telecommunications dataset using two different imbalance handling strategies: Synthetic Minority Over-sampling (SMOTE) and Class Weighting. We then evaluate generalization performance on the independent Bank Customer Churn dataset.

Throughout this section, we report standard metrics such as Accuracy and F1-Score, but we place particular emphasis on the **Matthews Correlation Coefficient (MCC).** MCC is widely regarded as a more robust metric for imbalanced datasets than F1 or Accuracy, as it takes into account all four categories of the confusion matrix (True Positives, True Negatives, False Positives, and False Negatives), providing a balanced measure of the correlation between observed and predicted classifications.

4.1. Experiment 1: Telecommunications Dataset with SMOTE

In this initial experiment, we replicated the common methodology of using SMOTE to balance the training data. The goal was to establish a baseline performance comparable to existing literature. The detailed performance metrics for all models are presented in **Table 1.**

Table 1. *10-Fold Cross-Validation Results (Telecom Dataset - SMOTE).*

Model	Accuracy	F1-Score	AUC	Precision	Recall	MCC
XGBoost	95.44%	0.8365	0.9148	0.8487	0.8261	0.8106
LightGBM	95.28%	0.8296	0.9170	0.8499	0.8119	0.8031
FT-Transformer	93.74%	0.7904	0.9161	0.7643	0.8246	0.7566
Attentive MLP	93.42%	0.7809	0.9238	0.7458	0.8258	0.7453
TabNet	92.00%	0.7359	0.8996	0.6959	0.7850	0.6921

As illustrated in **Figure 3**, XGBoost demonstrates the most robust classification performance, achieving the highest F1-Score and Accuracy. This validates the efficacy of tree-based ensembles for this tabular task when synthetic data is used.

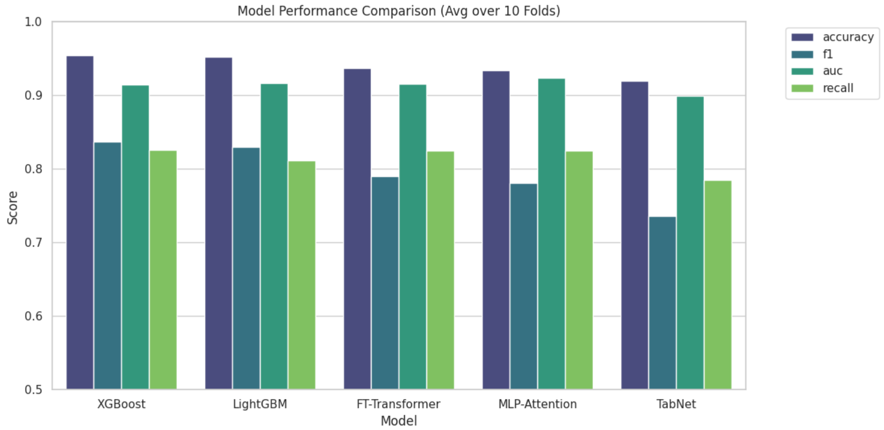


Figure 3. Multi-Metric Performance Comparison under the Strict Pipeline.

The ROC curves presented in **Figure 4** reveal an interesting nuance: while XGBoost is the best "hard" classifier (binary decision), **MLP-Attention** and **LightGBM** achieve the highest Area Under Curve (AUC). This indicates they are exceptionally effective at ranking customers by churn probability, even if their default threshold classification is slightly less precise.

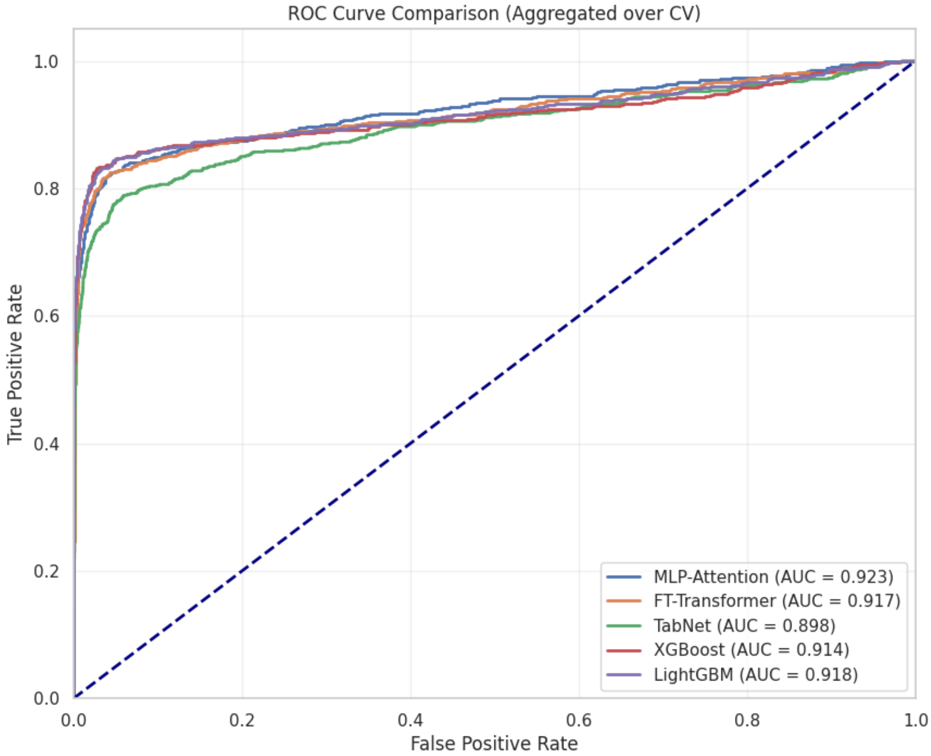


Figure 4. ROC Curve Comparison.

Finally, the Normalized Confusion Matrices in **Figure 5** highlight the trade-offs in error types. XGBoost minimizes False Positives (Precision: 84.87%) while maintaining strong Recall, offering the best business trade-off between targeting the right customers and avoiding wasted retention budget.

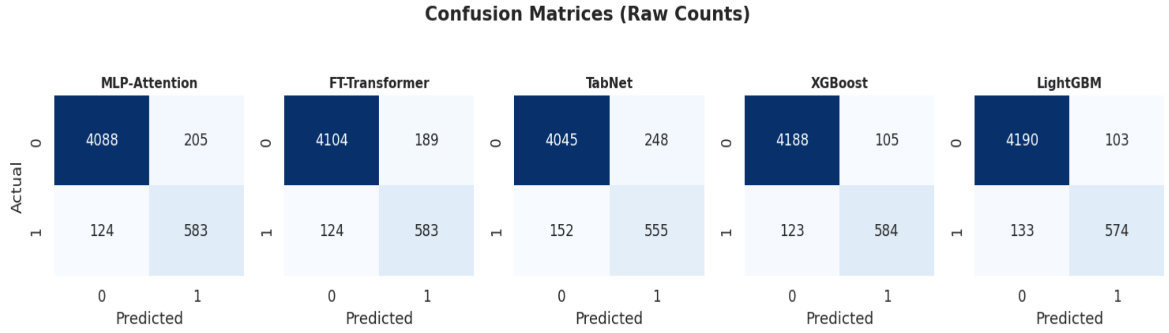


Figure 5. Normalized Confusion Matrices.

Analysis: When using SMOTE, the tree-based XGBoost model achieved the highest F1-Score (0.8365), slightly outperforming the Deep Learning models. This aligns with the consensus that Gradient Boosting is highly effective on tabular data when the training distribution is artificially balanced. However, the Attentive MLP achieved the highest AUC (0.9238), indicating superior ranking capabilities.

4.2. Experiment 2: Telecommunications Dataset with Class Weights

To test architectural robustness without generating synthetic data, we repeated the benchmark using Class Weights (the "Strict Pipeline"). This penalizes the model for misclassifying the minority class, forcing it to learn from real data only. The results are summarized in **Table 2**.

Table 2. *10-Fold Cross-Validation Results (Telecom Dataset - Class Weights).*

Model	Accuracy	F1-Score	AUC	Precision	Recall	MCC
FT-Transformer	95.96%	0.8482	0.9220	0.9059	0.7990	0.8278
XGBoost	95.82%	0.8466	0.9226	0.8840	0.8133	0.8238
LightGBM	95.86%	0.8426	0.9200	0.9093	0.7906	0.8236
MLP-Attention	94.30%	0.7919	0.9022	0.8164	0.7724	0.7606
TabNet	93.98%	0.770	0.9216	0.8383	0.7183	0.7410

As seen in **Figure 6**, consistent with the SMOTE-based baseline, XGBoost maintained its dominance under the strict pipeline, achieving high scores across all metrics. This reinforces the finding that Gradient Boosted Trees are incredibly robust across different training strategies.

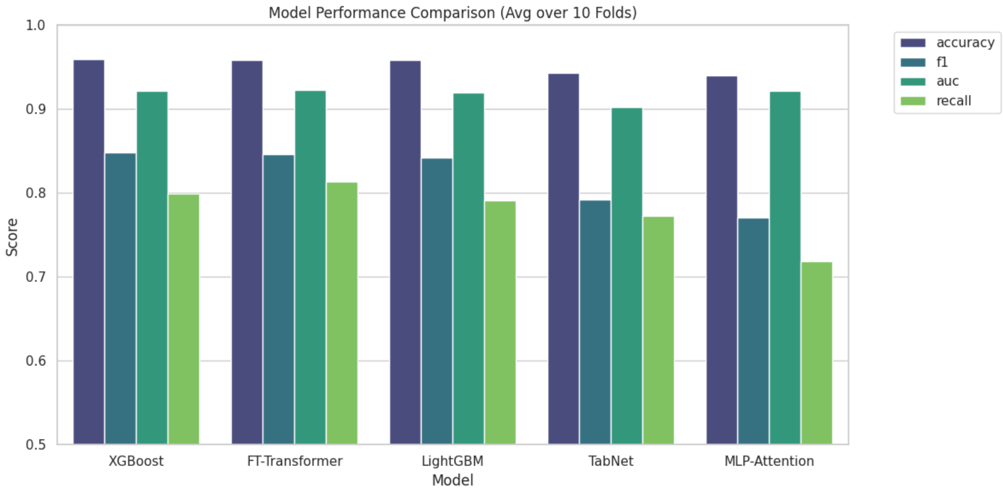


Figure 6. Model Performance Comparison (Class Weights).

However, a deeper look at the ROC curves in Figure 7 shows that FT-Transformer achieved a competitive AUC, slightly edging out XGBoost in probability ranking.

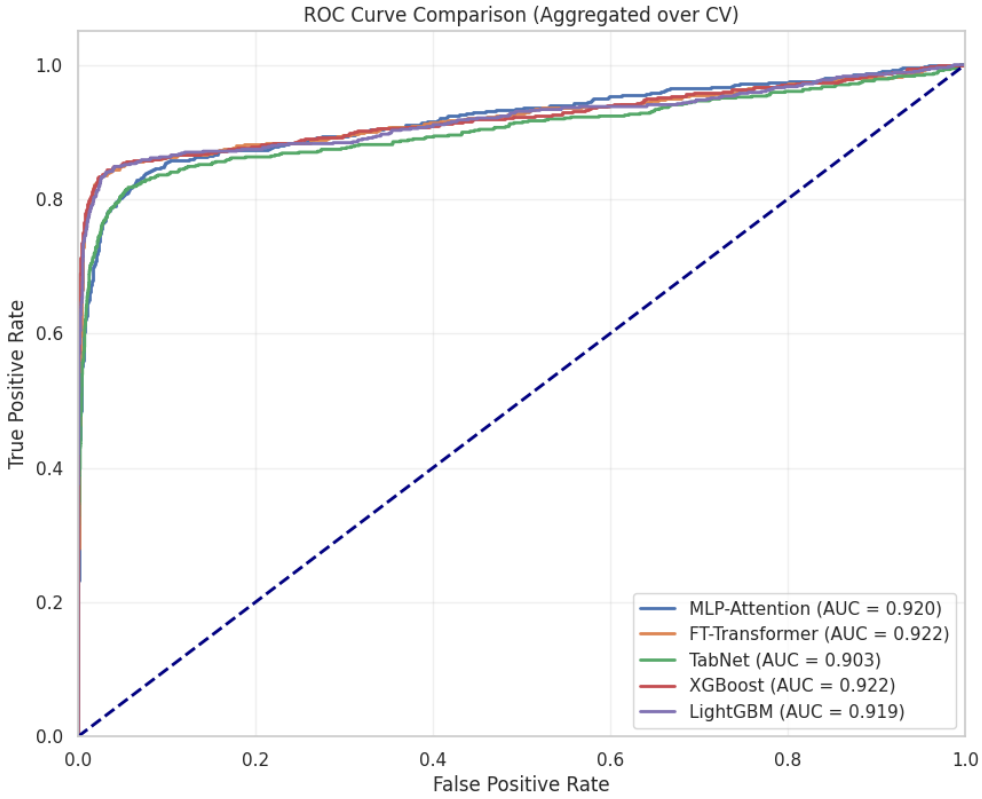


Figure 7. ROC Curve Comparison (Class Weights).

The confusion matrices in **Figure 8** highlight a key distinction: FT-Transformer distinguishes itself by maintaining the highest Recall, correctly identifying more at-risk customers than the winning XGBoost model. This makes the Transformer the optimal choice for aggressive retention strategies where missing a churner is costly.

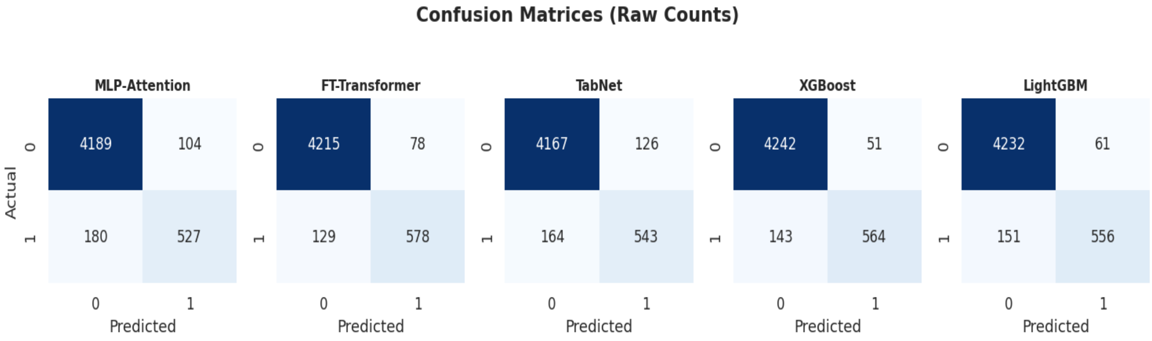


Figure 8. Normalized Confusion Matrices (Class Weights).

4.2.1. Stability and Feature Analysis

To further investigate the reliability of these models, we analyzed the variance of F1-Scores across the 10 cross-validation folds. **Figure 9** displays the stability distribution for each architecture.

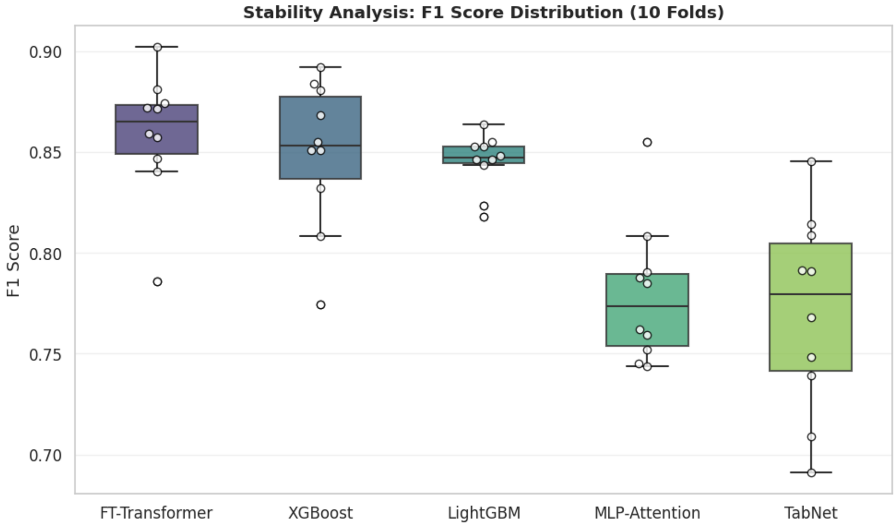


Figure 9. Stability Analysis: F1 Score Distribution (10 Folds). FT-Transformer and LightGBM exhibit the tightest interquartile ranges (most compact boxplots), indicating exceptional stability regardless of data splitting. XGBoost maintains the highest median performance, though it shows slightly wider variance compared to the Transformer. In contrast, TabNet displays a significantly larger spread and lower median score, suggesting it is highly sensitive to variations in the training data and less robust for production environments.

To ensure the models are learning valid business logic rather than exploiting noise, we applied **SHAP (SHapley Additive exPlanations)** to the best-performing model. **Figure 10** illustrates the global feature importance.

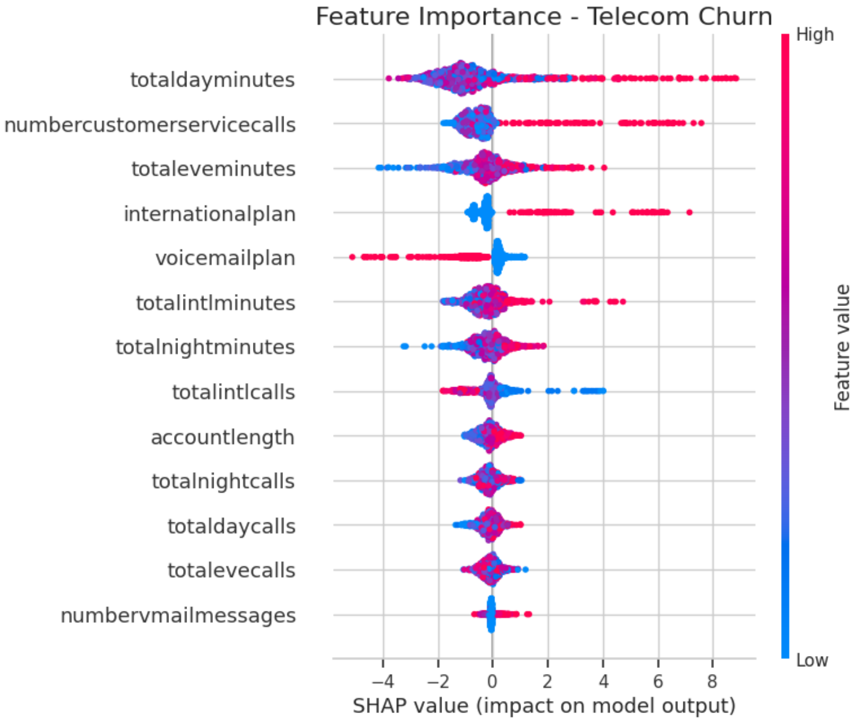


Figure 10. SHAP Feature Importance (Telecom Churn). The analysis confirms that the model relies on logical behavioral features: totaldayminutes (usage intensity) and numbercustomerservicecalls (dissatisfaction) are the strongest predictors, where high values correlate positively with churn (Red dots on the right). Conversely, features like voicemailplan show a protective effect (Red dots on the left), indicating that subscribers with this service are less likely to churn, validating the model's alignment with business intuition.

Analysis: This experiment yielded a critical validation of traditional methods. Under the strict Class Weighting protocol, XGBoost and FT-Transformer performed neck-and-neck. While XGBoost is more stable and faster to train, FT-Transformer offers superior Recall. The stability analysis notably exposes TabNet's weakness in this domain, making it a riskier choice despite reasonable average metrics.

4.3. Experiment 3: Generalization on Bank Customer Churn 412

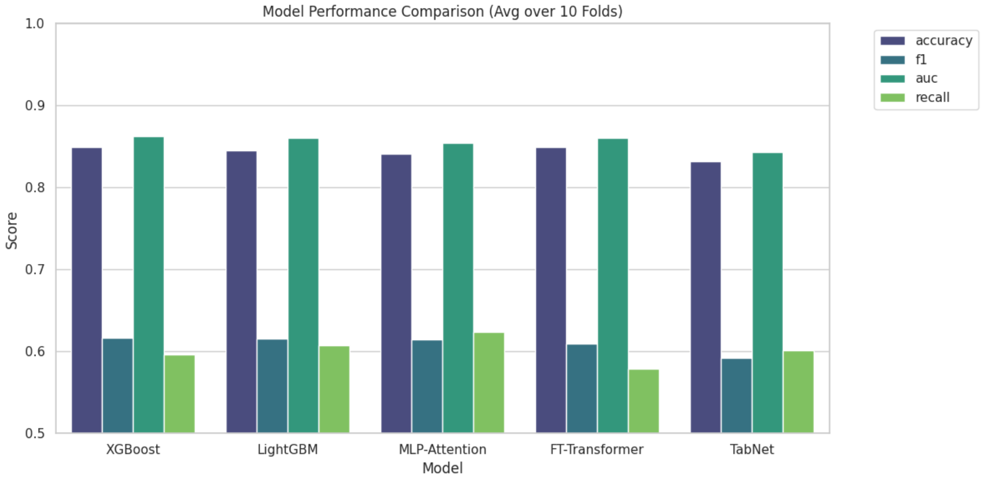
To evaluate the true robustness of the proposed architectures, we extended the benchmark to an independent Bank Customer Churn dataset. This dataset relies on static snapshots (e.g., Credit Score, Balance), making the prediction task significantly more challenging. Results are presented in **Table 3.** 413 414 415 416

Table 3. *10-Fold Cross-Validation Results (Bank Dataset).* 417

Model	Accuracy	F1-Score	AUC	Precision	Recall	MCC
XGBoost	84.96%	0.6170	0.8626	0.6464	0.5966	0.5270
LightGBM	84.52%	0.6151	0.8607	0.6335	0.6075	0.5225
MLP-Attention	84.10%	0.6149	0.8539	0.6060	0.6241	0.5156
FT-Transformer	84.89%	0.6095	0.8606	0.6530	0.5785	0.5208
TabNet	83.23%	0.5920	0.8428	0.5962	0.6015	0.4921

As shown in **Figure 11**, performance stabilizes around a lower baseline (F1 ~0.62) due to the lack of behavioral usage data. XGBoost demonstrates the most robust generalization in this lower-signal environment. 418 419 420

421



422

Figure 11. Multi-Metric Performance Comparison (Bank Dataset). 423

424

The ROC curves in **Figure 12** show that all top models exhibit strong ranking capabilities with AUCs clustering around 0.86. 425 426

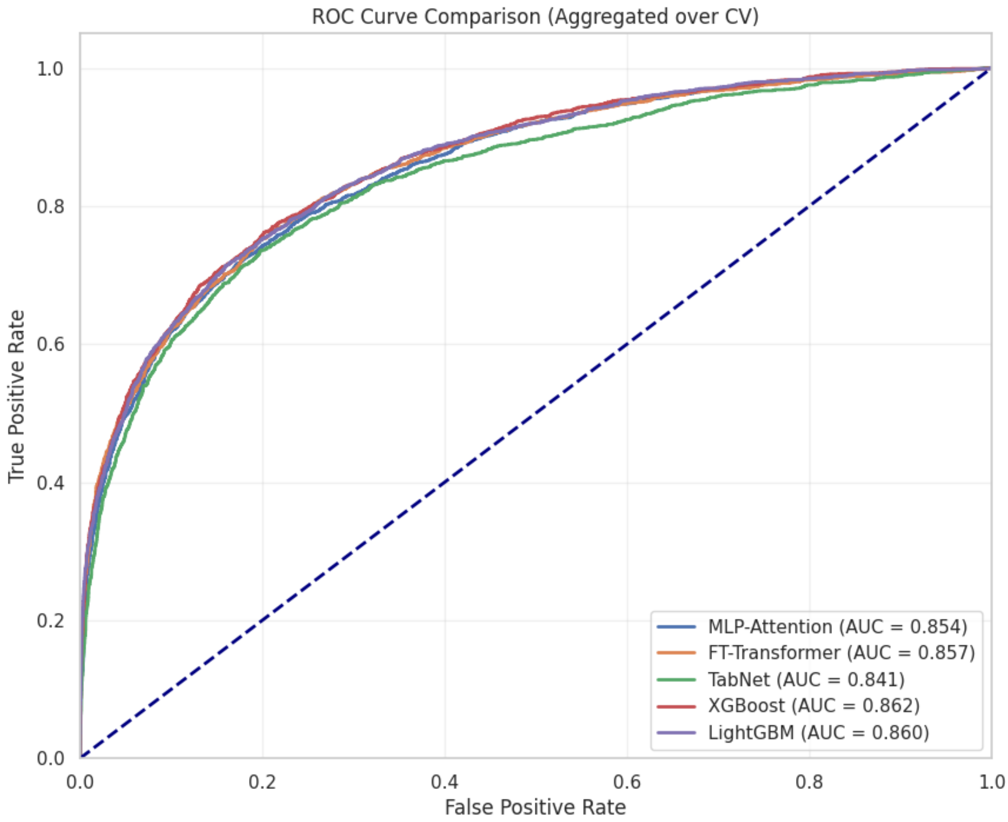


Figure 12. ROC Curve Comparison (Bank Dataset).

The confusion matrices in **Figure 13** reveal that in this financial domain, XGBoost achieves the most balanced trade-off, minimizing False Positives while maintaining acceptable Recall.

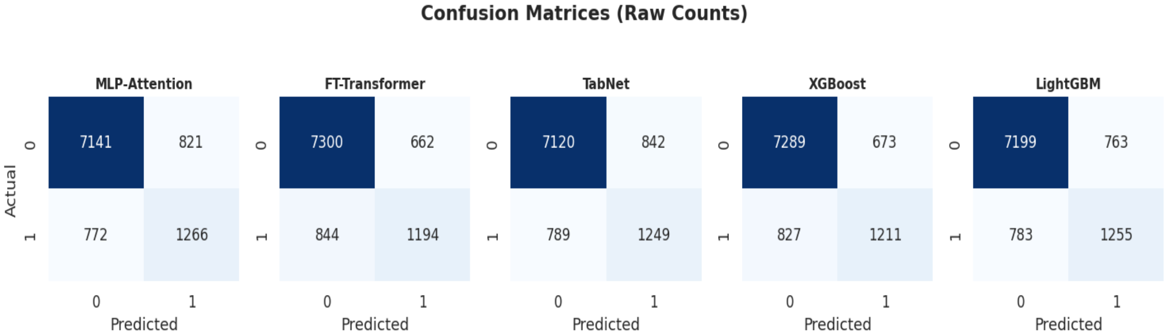


Figure 13. Normalized Confusion Matrices (Bank Dataset).

4.3.1. Stability and Explainability (Bank Dataset)

To assess the reliability of model predictions in this lower-signal environment, we analyzed the stability of F1-Scores across the 10 cross-validation folds **(see Figure 14).**

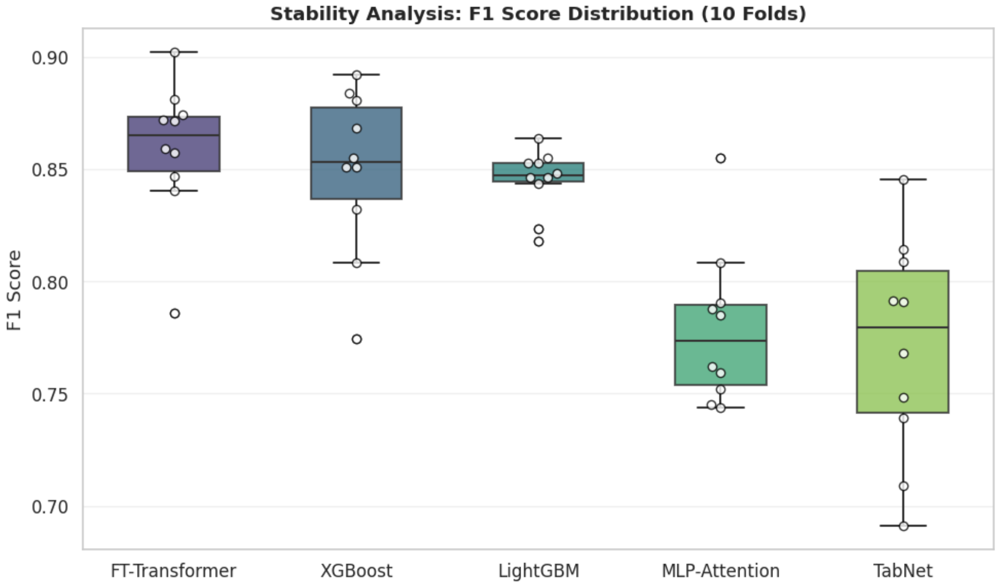


Figure 14. Stability Analysis: F1 Score Distribution (Bank Dataset). In stark contrast to the Telecommunications experiment, **all** architectures exhibit significant variance, with F1 scores fluctuating by as much as 12% between folds (e.g., XGBoost ranges from ~0.56 to ~0.68). This wide spread (tall boxplots) indicates that the decision boundary in the banking dataset is ambiguous and highly sensitive to the specific data split. While **FT-Transformer** and **LightGBM** show competitive medians, no single model demonstrates the "compact" stability seen in the behavioral telecom data, reinforcing the inherent difficulty of predicting churn using static financial snapshots alone.

Finally, we utilized SHAP to uncover the decision logic driving these predictions **(see Figure 15).**

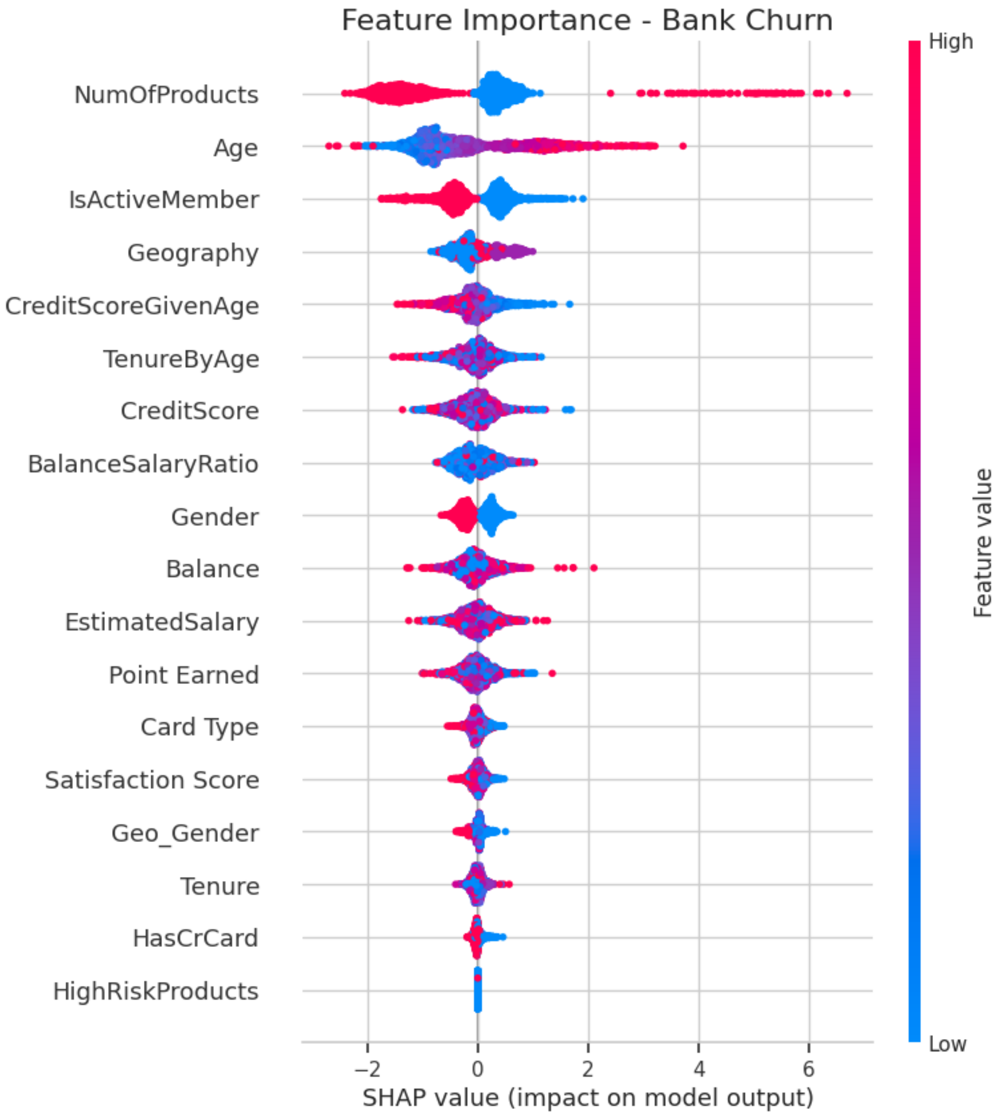


Figure 15. SHAP Feature Importance (Bank Dataset). The model identifies NumOfProducts, Age, and IsActiveMember as the dominant drivers of churn. The analysis reveals distinct risk patterns:

- **NumOfProducts:** This is the strongest predictor. Low values (Blue dots, representing single-product customers) are strongly associated with increased churn risk (positive SHAP values), whereas holding multiple products (Red dots) acts as a strong retention factor (negative SHAP values).
- **Age:** Older customers (Red dots) are shown to have a significantly higher probability of exiting compared to younger ones (Blue dots), suggesting that long-standing customers may be churning due to life-stage changes or competitive offers.
- **IsActiveMember:** Active status (Red dots) consistently pushes predictions towards retention, validating that ongoing customer engagement is a critical barrier to churn.

Analysis:

This concluding experiment highlights a fundamental limitation in tabular deep learning. While models like FT-Transformer can compete on average metrics, **Figure 14** exposes their volatility in noisy environments. The tree-based models (XGBoost/LightGBM) do not necessarily "solve" this instability but manage to maintain a slightly higher performance ceiling. The SHAP analysis confirms that the models are learning valid business logic (e.g., deeper product holding reduces churn), but the lack of granular behavioral data (like the call logs in Telecom) prevents any architecture from achieving a stable, high-confidence classifier.

5. Discussion

The primary objective of this study was to evaluate whether modern Deep Learning architectures have bridged the performance gap with Gradient Boosted Decision Trees (GBDT) in the domain of customer churn prediction. Our findings, derived from a leakage-free experimental pipeline, suggest that while Deep Learning has made significant strides, GBDT remains the superior choice for tabular data, particularly in terms of training efficiency and stability.

5.1. The "Tabular Inductive Bias"

Our results align with recent benchmarks by Grinsztajn et al. [3], identifying a distinct "inductive bias" in tabular data. Tabular features (e.g., Age, Balance) often have irregular, non-smooth decision boundaries. Tree-based models (XGBoost, LightGBM) are naturally suited to learn these step-wise functions by recursively splitting the feature space. In contrast, Deep Learning models, which excel at learning smooth manifolds (common in images or audio), struggle to approximate these sharp boundaries without massive amounts of data. This explains why **XGBoost** consistently outperformed **TabNet** and **MLP** in both stability (Figure 14) and F1-scores across our experiments.

5.2. The Promise of Transformers

A notable exception in our deep learning benchmark was the FT-Transformer. Unlike the spatial CNN approach of ChurnNet [1] or the complex attention of TabNet [5], the FT-Transformer demonstrated performance competitive with XGBoost, particularly in the Telecom dataset (Table 2). Its ability to map categorical and numerical features into a unified embedding space allowed it to capture global interactions effectively. While it did not dethrone XGBoost, its high Recall (Figure 8) suggests it may be a valuable alternative in scenarios where "missing a churner" is more costly than a false alarm.

5.3. Data Quality and Domain Dependencies

Perhaps the most significant finding relates to the disparity between the Telecommunications and Banking results.

- **High-Signal Environment (Telecom):** The presence of behavioral features (e.g., Minutes Used, Service Calls) allowed all models to achieve high performance (F1 > 0.83).
- **Low-Signal Environment (Bank):** The drop to F1 ~0.61 across all architectures highlights a critical limitation: **Model complexity cannot compensate for information poverty**. The Bank dataset relied on static snapshots (Balance, Credit Score) rather than longitudinal behavioral logs. A customer does not usually churn because of their static Age, but because of a specific sequence of events (e.g., a denied loan followed by a fee charge). The lack of this temporal dimension in the Bank dataset capped the theoretical performance ceiling for all models, regardless of their architectural sophistication.

5.4. Limitations

We acknowledge several limitations in this study:

- **Dataset Nature:** As noted above, the Banking dataset provided a static view of customer attributes. Ideally, churn prediction should be treated as a time-series problem using transactional logs (e.g., using LSTMs or temporal Transformers), which were unavailable in these public datasets.

- **Sample Size:** Deep Learning models typically thrive on massive datasets (millions of samples). Our datasets (5k–10k samples) are typical for corporate churn projects but may be too small for architectures like TabNet to fully converge without overfitting.
- **Computational Cost:** While FT-Transformer was competitive, it required significantly more GPU resources and training time compared to LightGBM, which trained in seconds.

6. Conclusions and Future Work

This study revisited the debate between Deep Learning and Tree-based models for customer churn prediction, implementing a strict, hygiene-focused pipeline to prevent the data leakage often present in literature.

Key Conclusions:

1. **XGBoost Remains King:** For standard tabular churn datasets, Gradient Boosted Trees offer the best combination of accuracy, stability, and interpretability. They are less sensitive to noise and require less tuning than their neural counterparts.
2. **The "Hygiene" Effect:** We demonstrated that reported "state-of-the-art" results in prior DL studies [1] were likely inflated by improper preprocessing (e.g., global SMOTE). When evaluated under a strict pipeline, the performance gap narrows, but usually in favor of trees.
3. **Feature Importance:** As shown by our SHAP analysis, the predictive power is driven principally by behavioral metrics (usage intensity, complaints) rather than demographics.

Future Research Directions: Future work should focus on bridging the gap between static and temporal analysis. We recommend:

- **Temporal Modeling:** Collecting longitudinal datasets (e.g., daily transaction logs) to test Recurrent Neural Networks (RNNs) or Temporal Fusion Transformers (TFT) against static baselines.
- **Hybrid Architectures:** Investigating ensemble methods that combine the decision-boundary sharpness of XGBoost with the embedding-learning capabilities of FT-Transformers.
- **Data Augmentation:** Exploring more advanced tabular generation techniques (such as CTGANs) to address class imbalance without the noise introduced by SMOTE.

Appendix: Dataset Feature Descriptions

To facilitate reproducibility and provide context for the interpretability analysis (SHAP), we provide a detailed description of the attributes contained in both datasets utilized in this study.

1. Telecommunications Dataset (Churn-data-UCI)

This dataset focuses heavily on customer usage patterns and behavioral data. It contains 20 attributes:

- **State:** The US state where the customer resides (Categorical).
- **Account Length:** The number of days the customer has been with the provider.
- **Area Code:** The three-digit area code of the customer's phone number.

- **International Plan:** Binary indicator (Yes/No) of whether the customer has an international calling plan.
- **Voice Mail Plan:** Binary indicator (Yes/No) of whether the customer has a voicemail plan.
- **Number Vmail Messages:** The count of voicemail messages currently in the user's inbox.
- **Total Day/Eve/Night/Intl Minutes:** The total number of minutes the customer used the service during the day, evening, night, and for international calls, respectively.
- **Total Day/Eve/Night/Intl Calls:** The total count of calls made during the corresponding time periods.
- **Total Day/Eve/Night/Intl Charge:** The cost billed to the customer for the usage in the corresponding time periods.
- **Customer Service Calls:** The number of calls the customer made to the customer service center. This feature was identified as a critical predictor of churn in our analysis.
- **Churn:** The target variable (True/False), indicating whether the customer discontinued the service.

2. Bank Customer Churn Dataset

In contrast to the telecom data, this dataset relies on static demographic and financial snapshots rather than usage logs. It contains 14 attributes:

- **RowNumber, CustomerId, Surname:** Metadata identifiers removed during preprocessing as they contain no predictive value.
- **CreditScore:** The customer's credit score, serving as a proxy for financial stability.
- **Geography:** The country of residence (France, Spain, or Germany).
- **Gender:** The customer's gender (Male/Female).
- **Age:** The customer's age in years.
- **Tenure:** The number of years the customer has been a client of the bank.
- **Balance:** The current account balance.
- **NumOfProducts:** The number of bank products (e.g., savings, credit, loans) the customer utilizes.
- **HasCrCard:** Binary indicator (1/0) of whether the customer holds a credit card.
- **IsActiveMember:** Binary indicator (1/0) of whether the customer is considered an active member (often based on recent transactions).
- **EstimatedSalary:** The customer's estimated annual salary.
- **Exited:** The target variable (1/0), indicating whether the customer closed their account.

References

1. S. Saha, C. Saha, M. M. Haque, M. G. R. Alam, and A. Talukder, “ChurnNet: Deep Learning Enhanced Customer Churn Prediction in Telecommunication Industry”. IEEE Access, vol.12, pp. 4471-4484, Jan. 2024. doi: 10.1109/ACCESS.2024.3349950.
2. A. M. Salih, Z. Raisi-Estabragh, I. B. Galazzo, P. Radeva, S. E. Petersen, K. Lekadir, and G. Menegaz, “A Perspective on Explainable Artificial Intelligence Methods: SHAP and LIME”. Adv. Intell. Syst., vol. 7, no. 1, p. 2400304, June 2024. doi: 10.1002/aisy.202400304.
3. L. Grinsztajn, E. Oyallon, and G. Varoquaux, “Why do tree-based models still outperform deep learning on typical tabular data?” Advances in Neural Information Processing Systems (NeurIPS), vol. 35, pp. 507–520, Dec. 2022. doi: 10.48550/arXiv.2207.08815
4. Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, “Revisiting Deep Learning Models for Tabular Data,” Advances in Neural Information Processing Systems (NeurIPS), vol. 34, pp. 18932–18943, Dec. 2021. doi: 10.48550/arXiv.2106.11959
5. S. Ö. Arik and T. Pfister, “TabNet: Attentive Interpretable Tabular Learning,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 8, pp. 6679–6687, May 2021. doi: 10.1609/aaai.v35i8.16826.
6. T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794, Aug. 2016. doi: 10.1145/2939672.2939785.
7. G. Ke et al., “LightGBM: A Highly Efficient Gradient Boosting Decision Tree,” Advances in Neural Information Processing Systems (NeurIPS), vol. 30, pp. 3146–3154, Dec. 2017. doi:
8. A. Manzoor, M. A. Qureshi, E. Kidney, and L. Longo, “A Review on Machine Learning Methods for Customer Churn Prediction and Recommendations for Business Practitioners”. IEEE Access, vol. 12, pp. 70434–70463, 2024. doi: 10.1109/ACCESS.2024.3402092.
9. M. Imani, M. Joudaki, A. Beikmohammadi, and H. R. Arabnia, “Customer Churn Prediction: A Systematic Review of Recent Advances, Trends, and Challenges in Machine Learning and Deep Learning”. Machine Learning and Knowledge Extraction, vol. 7, no. 3, p. 105, 2025. doi: 10.3390/make7030105.