SynthRank: Synthetic Data Generation of Individual's Financial Transactions Through Learning to Ranking

Weixian Waylon Li [1] **Mengyu Wang** [1] **Carsten Maple** [2,3] **Tiejun Ma** [1,3]
[1]University of Edinburgh
[2]University of Warwick
[3]The Alan Turing Institute
{W.Li-67, M.Wang-100}@sms.ed.ac.uk
cm@warwick.ac.uk
tiejun.ma@ed.ac.uk

Abstract

Obtaining financial transaction data is notably challenging, largely attributed to strict privacy regulations and the sensitivity of personal financial details. In this context, synthetic data generation can offer a practical solution. In contrast to existing approaches that utilise generative models dedicated to replicating authentic data distributions, this paper proposes SynthRank, an innovative approach founded on the learning-to-rank (LETOR) algorithm, for financial transaction synthetic data creation. Focusing specifically on the task of risky trader detection and prediction, we leverage LETOR techniques to generate ranking scores for each attribute of transaction. These scores are aggregated into a new vector, constituting the synthetic data. By segmenting data into distinct ranking groups, we produce synthetic data without quantity limitations. Crucially, our approach guarantees the privacy protection of individual data, as sensitive information becomes challenging for attackers to infer. Our comprehensive analysis demonstrates that SynthRank not only enhances prediction power but also preserves the essential distribution characteristics and privacy of the original dataset.

1 Introduction

Synthetic data refers to data produced through mathematical models or algorithms designed to address one or more data science objectives (Jordon et al. 2022). Its importance is amplified in contexts where individual data privacy needs to be preserved. In the finance sector, significant constraints exist in accessing data both within and across organisations. Financial data, such as transaction records, often contain sensitive information including personal detail, which preclude it being shared. Furthermore, limited historical data archives and skewed class distributions pose additional challenges on training trustworthy deep learning models that require substantial data (Assefa et al. 2021).

Researchers have put great efforts into generating synthetic data. Such data satisfies the increasing need for big data to train models for multiple tasks including risk assessment and money laundry detection (Fu et al. 2019; Rizzato et al. 2023). Integrating synthetic data into financial models becomes essential for business and society, not merely an area of academic exploration.

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

While synthetic data generation has advanced in capturing feature distributions of the real data, issues remain concerning its utility and privacy. The utility is often referring to task-specific effectiveness, but this may not correspond with the generated data's distributional similarity (Jordon et al. 2022). Moreover, there is a persistent risk of sensitive information disclosure, despite the models' aim to only emulate the statistical characteristics of original data. Malicious attackers can collect public information to infer protected real values from the synthetic data (Zhao et al. 2021a).

To improve the quality of synthetic financial transaction data and align with the rigorous privacy regulations, we propose **SynthRank**, a novel approach based on learning-to-rank (LETOR) algorithms. We take a case study, the task of risky trader prediction, which shares characteristics with many financial transaction related analytic tasks, such as fraud detection (Zheng et al. 2019) and money laundering detection (Paula et al. 2016). We propose to adapt LETOR techniques to extract relationships between input transaction data and risk labels to construct our synthetic transaction data. This synthetic data framework offers two key advantages. Firstly, it allows for unlimited synthetic data generation by applying different group allocations. Secondly, it significantly enhances the privacy safeguard, since inferring the original data from the synthetic dataset without both ranking model and group allocation knowledge is challenging, as demonstrated in our experiments (Section 5.3). Additionally, our synthetic data maintains higher quality than the baseline models, as evidenced by improved prediction AUC scores and preserved distribution characteristics. In summary, the main contributions of this paper are as follows:

- We adapt LETOR algorithms to the generation of synthetic financial transaction data. These algorithms are capable of extracting relationships between original data items and labels, facilitating the construction of synthetic features. The allocation of ranking groups enhances the flexibility and controllability of the synthetic process.
- We recognise the conflicting objectives of privacy, utility, and feature similarity in financial data synthesis. In response, we introduce SynthRank, a task-oriented pipeline for synthetic financial transaction data generation. Our approach achieves an exceptional balance between privacy protection against inference attacks and utility in

risky trader prediction, while maintaining an acceptable level of feature fidelity. The results indicate a potential direction for future research in synthetic data generation for other financial tasks.

2 Background

2.1 Synthetic Data Generation for Financial Domain

Regarding synthetic data generation techniques, the Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al. 2002) is noteworthy as a traditional algorithm. It is specifically designed to generate minority class samples, aiming to address the class imbalance problem. With the surge of deep learning, synthetic data generation witnessed a transformative phase. Three predominant deep learning generative models have emerged: Bayesian networks (BNs) (Zhang et al. 2017), autoencoders (AEs) (Goodfellow, Bengio, and Courville 2016), and Generative Adversarial Networks (GANs) (Goodfellow et al. 2014; Zhao et al. 2021b). Additionally, the integration of language models in synthetic data generation, particularly for textual and tabular data, has shown promising results (Borisov et al. 2023; Solatorio and Dupriez 2023).

Despite advancements in synthetic data generation, their applications in finance is constrained by limited access to real-world datasets. Recent applications of GANs for generating time-series financial data (Wiese et al. 2020; Nickerson et al. 2023) and credit card fraud detection data (Strelcenia and Prakoonwit 2022) have shown promise. However, it is crucial to point out a significant limitation acknowledged by the authors in these studies: the generation methods lack a comprehensive assessment of privacy guarantees. This might expose the data to potential inference attacks, thus posing a risk to the confidentiality (Ram Mohan Rao, Murali Krishna, and Siva Kumar 2018).

Given this privacy concern, there is an evident and pressing need for novel methods with comprehensive evaluation. These methodologies must encompass critical dimensions, including utility, similarity to real data, and privacy safeguards, particularly for generating financial transaction data (Jordon et al. 2022). This underscores the ongoing quest for robust and secure synthetic transaction data solutions.

2.2 LEarning-TO-Rank (LETOR)

Learning-to-Rank (LETOR) algorithms have firmly established their significance in various domains, such as recommendations and document retrieval, proficiently ranking items based on their inherent features (Phophalia 2011). In contrast, the potential of LETOR in synthesising financial transaction data remains largely unexplored. This highlights both the novelty and difficulty of integrating LETOR techniques into the generation of privacy-preserving financial transaction data, an endeavour that has yet to be fully realised in the field.

The challenges in this context are twofold. Firstly, the distinction between LETOR algorithms and traditional generative models lies at the definition level. LETOR originally focuses on ranking and retrieving existing items (Fuhr 1989), whereas generative models aim to create entirely new data instances. Bridging this conceptual gap between retrieval and generation is challenging and requires innovative adaptations of LETOR techniques. Secondly, ensuring that privacy is not compromised in synthetic data generation is paramount, especially in sensitive domains such as financial transactions. Integrating privacy-preserving mechanisms into LETOR-based approaches while maintaining data quality and utility is a multifaceted problem that demands advanced techniques.

However, LETOR also demonstrates its potential. Its ability to discern relationships at individual, paired, and list levels makes it a promising solution for modelling the statistical attributes of data and enhancing embeddings (Phophalia 2011). Moreover, LETOR's inherent adaptability enables the continuous generation of ranking scores across diverse group allocations. This positions LETOR as a potential avenue for synthetic data generation, despite not being originally conceived for such tasks.

3 Preliminaries

3.1 Dataset

This research utilises a unique dataset from leading UK-based trading sectors, containing individual trading transactions. The scarcity of real-world financial transaction datasets, often due to privacy, regulatory, and proprietary constraints (Hand 2018), makes our dataset particularly valuable. It provides a rare opportunity to study financial transaction synthetic data generation in-depth.

The dataset contains 13,607,120 trading records from November 2003 to July 2014, produced by 20,514 active traders. Each entry corresponds to an individual trade executed by these traders. The dataset is split as follows: 72% (9,797,100 records) for training, 8% (1,088,580 records) for validation, and 20% (2,721,440 records) for testing.

There are three feature groups constructed in the dataset: (1) Behavioural and demographic features of traders, informed by stock exchange dealing desk insights; (2) Past performance indicators, including a detailed analysis of the traders' last 20 trades; and (3) Preferences for specific markets and channels, identifying patterns in trading choices. This comprehensive dataset is instrumental in developing models to detect and analyse risky trading behaviours.

3.2 Task

Building on the aforementioned dataset, our target is to generate financial transaction data in a task-oriented way. Accordingly, we select the task for which this dataset for classifying risky traders in the context of trading and Contracts for Difference (CFDs) (Kim et al. 2020).

This task is particularly crucial given that these trading methods contribute significantly to the financial market, with an estimated 10% of the £1.2 trillion traded annually on the London Stock Exchange related to such tradings (Ma et al. 2022). Identifying traders who may exploit leverage for substantial gains, potentially through illegal means such as price manipulation or insider trading, is crucial for maintaining

market integrity (Kozlov 2014; Hilal, Gadsden, and Yawney 2022). The task is formally defined as follows.

Given a labelled dataset $D = \{y_j, x_j\}_{j=1}^{n}$, each x_j denotes the feature set of trade j, with n representing the total trade count. The inherent sparsity of individual trade data is addressed by associating each trade with its corresponding trader. This association facilitates the enhancement of feature representation by incorporating information from a trader's preceding 20 trades when evaluating trade j.

The primary objective is to determine the hedging strategy for trade j, which is based on a binary target condition. If the return from the next hundred trades subsequent to j, denoted as $\text{Return}_{i,j}$, for a specific trader i is in the top 1%, the binary target y_{ij} is set to 1; otherwise, it is assigned a value of 0. The return is given by:

$$\text{Return}_{i,j} = \frac{\sum_{1<k\leq 100} P\&L_{i,k}}{\sum_{1<k\leq 100} \text{Margin}_{i,k}}$$

Indices i and j respectively represent the trader and trade. The $P\&L$ metric stands for the profit and loss, while the Margin corresponds to the funds mandated by the market maker, typically calculated as the product of the stake size and margin requirement. To determine the label for trade j, trader i is assessed at the time of the trade's initiation. Traders achieving a return exceeding 1% from their subsequent hundred trades are categorised as risky and trades from these risky traders will be hedged.

4 Methodology

4.1 SynthRank Pipeline

Figure 1 illustrates the procedure of our synthetic financial transactions generation pipeline. We firstly train a ranking model $\mathcal{R}$ on our labelled dataset using random ranking group allocation. To illustrate this, the ranking model, $\mathcal{R}$, outputs a ranking score s_i for each input item $\boldsymbol{x}_i$, a d-dimensional vector, where d represents the feature count. The equation is as:

$$\mathcal{R}(\boldsymbol{x}_i) = s_i \tag{1}$$

Then we generate synthetic data based on this ranking model $\mathcal{R}$. The original dataset is randomly split to G ranking groups. A trading item $\boldsymbol{x}_i$ is broken down into D input items $\{\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}, \cdots, \boldsymbol{x}_{iD}\}$. Each $\boldsymbol{x}_{ij}$ represents the j-th independent feature, with all other $d-1$ features masked by 0. For example, $\boldsymbol{x}_{i1} = (x_{i1}, 0, \cdots, 0)$. The ranking model $\mathcal{R}$ outputs feature-specific ranking scores s_{ij} for all $\boldsymbol{x}_{ij}$ as:

$$\mathcal{R}(\boldsymbol{x}_{ij}) = \mathcal{R}((0, \cdots, x_{ij}, \cdots, 0)) = s_{ij} \tag{2}$$

We aggregate all s_{ij} to substitute the corresponding features and generate a synthetic data item $E(\boldsymbol{x_i})$ as:

$$E(\boldsymbol{x_i}) = (s_{i1}, s_{i2}, \cdots, s_{iD}) \tag{3}$$

We repeat this procedure for the chosen original data items to generate synthetic data. There are two main advantages of our pipeline.

Flexible and controllable generation. SynthRank exhibits flexibility, allowing for the generation of data without quantity limitations. While it also maintains controllability, providing the capability to generate synthetic data with specific labels. Since the generation process is based on ranking groups, we can reallocate these groups to generate diverse synthetic data items derived from the same original item $\boldsymbol{x_i}$. The alteration of ranking group allocation can lead to variations in ranking outputs, yielding distinct synthetic features. Nonetheless, these modified features can also retain their characteristic properties by preserving relationships with other ranked items. Consequently, there is no generation quantity limitation, facilitating dynamic augmentation of our dataset to meet specific requirements. Moreover, owing to the reliance on original data items in the generation process, precise control over the synthesis is achievable. Specifically, it is feasible to extract samples of risky traders and generate diverse data for the minority class.

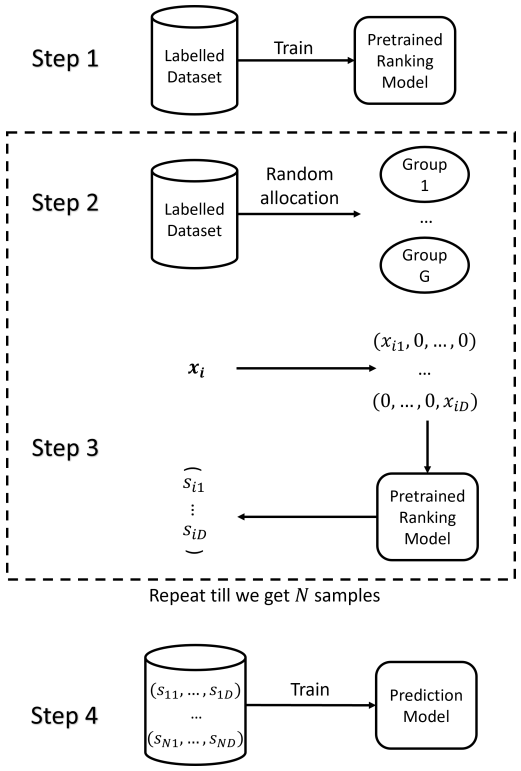

Figure 1: Illustration of SynthRank pipeline.

Privacy-preserved Generation Inference of sensitive information from our synthetic data is challenging, providing security of privacy. When using synthetic data outside the secure environment, attackers can utilise publicly available information to infer the protected individual information. However, this approach is ineffective against our synthetic data due to the incorporation of two elements of knowledge intrinsic to our generation process, the ranking group allocation and the ranking model. Both aspects are concealed from potential attackers. Despite the existence of inference attack algorithms (Zhao et al. 2021a), they are difficult to operate successfully on our synthetic data in the absence of ranking group information. Consequently, it becomes unfeasible to deduce the original values from our generated synthetic data. This assertion is substantiated by the results of our pri-

vacy inference attack assessment detailed in Section 5.3.

4.2 LETOR Algorithms Adaptation

To adapt LETOR algorithms into our SynthRank pipeline, we mainly choose the pairwise and listwise approach. LETOR algorithms can be categorised into three distinct types: pointwise, pairwise, and listwise, based on the quantity of considered documents at a time. Due to the inherent volatility of the market, the returns of trades are affected by numerous intricate factors, often failing to consistently reflect the accurate ranking of trades. Therefore, we allocate ranking groups with large sizes to pairwise or listwise approaches, ensuring the stability of training ranking models.

Group allocation plays a critical role in configuring ranking models for SynthRank. During the pretraining phase, we employed a grouping strategy in two steps: firstly, grouping traders by markets, and secondly, ensuring the inclusion of at least one risky trader in each group whenever possible. This approach aims to enhance the learning of distinctions between risky and normal traders within each group. For the generation process, we use random group allocation to ensure the diversity of the transaction data generation. The size of groups serves as a hyperparameter, where finding the right balance is crucial. Groups that are too small may miss common characteristics, while overly large groups may overlook local features. In contrast, when transforming the test set for predicting risky traders, all the data is assigned to a single group. This is to ensure the stability of the generated ranking embeddings for the test set.

The specific ranking algorithm we adopt is LambdaMART (can equipped with pairwise and listwise objective), motivated by recent studies demonstrating its superiority. These studies suggest that LambdaMART, based on gradient boosted decision trees (GBDT), typically outperforms deep learning-based ranking algorithms (Pang et al. 2020; Buyl, Missault, and Sondag 2023). Specifically, for our implementations, we employ XGBRanker[1] from XGBoost (Chen and Guestrin 2016) and LGBMRanker[2] from LightGBM (Ke et al. 2017). XGBRanker provides three objective options: *rank:pairwise*, *rank:ndcg*, and *rank:map*. In contrast, LGBMRanker exclusively employs the *lambdarank* objective.

4.3 Baselines

In this study, we select TVAE, CTGAN, CopulaGAN, and REaLTabFormer as our baseline models. This selection encompasses models from various time periods and includes a diverse range of model architectures.

TVAE and CTGAN are two prominent synthetic data generators that have been shown to outperform traditional Bayesian networks and other GAN-based generators by the year 2019 (Xu et al. 2019). Additionally, CopulaGAN, an advanced version of CTGAN, enhancing its ability to replicate both individual column features and the overall structure of datasets (Espinosa and Figueira 2023).

Furthermore, our study includes a comparative analysis of REaLTabFormer (Solatorio and Dupriez 2023), a GPT-2 based model outperforming recent models like Tab-DDPM (Kotelnikov et al. 2023) and GReaT (Borisov et al. 2023) in various deep learning architectures. Notably, its effectiveness was demonstrated using six real-world datasets, including a house pricing dataset with financial indicators such as income, underscoring its applicability in financial contexts.

We employ Synthetic Data Vault (SDV)[3], a widely-recognised open-source library, for implementing TVAE, CTGAN, and CopulaGAN. For REaLTabFormer[4], our experiments are conducted using its official implementation.

For all baseline models, we meticulously specify continuous and categorical features to maximize their performance. Each model undergoes training for 300 epochs to ensure adequate convergence. Notably, for the CopulaGAN model, we employ the default "beta" distribution, accommodating a diverse range of data shapes.

5 Experiments and Discussions

In our experimental evaluation, we aim to comprehensively assess the quality of the synthetic transaction data. We design three distinct experiments that cover the evaluation of utility, privacy, and similarity aspects of the synthetic data.

5.1 Metrics

Metrics are categorised into three groups for these evaluations: utility, privacy, and similarity, corresponding to three experiments (Section 5.2 to 5.4).

Utility Metrics: F_1, AUC score, P&L Utility of the synthetic data is assessed using a predictive task approach (see Section 5.2). For prediction performance, we use F_1 scores and Area under the ROC Curve (AUC) scores, both prevalent in prediction tasks, particularly with imbalanced datasets. Additionally, in the context of transaction data, where financial outcomes are crucial, we include the Profit & Loss (P&L) metric calculated as:

$$P\&L = \sum_{i=1}^{N} -(1 - y_i) * \text{NextProfit\_20}_i \tag{4}$$

In this formula, $y_i \in 0, 1$ is the predicted label for each of the N instances. NextProfit_20 represents the P&L for the upcoming 20 trades. If y_i is 0 (not high-risk), the P&L negatively impacts our total, indicating a loss. If y_i is 1 (high-risk), the P&L is zero due to hedging, meaning no gain or loss. Transactional costs are not included in this calculation.

Privacy Metric: PAI score Privacy evaluation is conducted using the Privacy Against Inference (PAI) score[5]. This metric measures the synthetic data's resilience against inference attacks aimed at extracting sensitive information (Choi et al. 2017; Jayaraman and Evans 2019; Yale et al. 2020). The PAI

[1]https://xgboost.readthedocs.io/en/stable/python/python_api.html\#xgboost.XGBRanker

[2]https://lightgbm.readthedocs.io/en/latest/pythonapi/lightgbm.LGBMRanker.html\#lightgbm.LGBMRanker

[3]https://docs.sdv.dev/sdv/

[4]https://github.com/worldbank/REaLTabFormer

[5]https://docs.sdv.dev/sdmetrics/metrics/metrics-in-beta/privacy-against-inference

score is derived from $(1 - Acc)$, where Acc is the accuracy of an attacker in inferring true sensitive information. Higher PAI scores indicate better privacy protection.

Similarity Metrics: KS test, KDE plot, Q-Q Plot For similarity assessment, we use the Kolmogorov-Smirnov (KS) test, kernel density estimate (KDE) plots, and Quantile-Quantile (Q-Q) plots. The KS test statistically measures how well the synthetic data matches the original data's distribution (Massey 1951). KDE and Q-Q plots provide visual insights into the distributional similarity, offering an intuitive understanding of how closely the synthetic data resembles the original.

5.2 Utility Evaluation: Prediction Task

The aims of utility evaluation is to evaluate whether the synthetic transactions can potentially reserve the predictive accuracy in identifying risky traders.

In this experiment, we utilise Random Forest (RF)[6] and Multi-layer Perceptron (MLP)[7] classifiers. These classifiers are trained on the generated synthetic transaction data and then tested on the original, uninvolved test set, which undergoes necessary feature transformations like standardisation. While more advanced classifiers exist (Kim et al. 2021; Chen et al. 2022), their evaluation is beyond this paper's scope. Similarly, while various MLP classifier implementations are available, our focus is not on comparing these packages but on demonstrating the use of ranking models for effective synthetic data generation. The impact of different MLP and ML model implementations is outside our current scope. The results are shown in Table 1.

As detailed in the result table, the original dataset sets a benchmark with an F_1 score of 0.511 and an AUC score of 0.837 for RF, and slightly higher scores for MLP.

Among the baselines, a common challenge observed is that classifiers struggle to accurately identify risky traders when trained on generated transaction data. This problem is especially evident in models like TVAE, as Kiran and Kumar points out, where they struggle to effectively represent minority classes. This is a significant issue in financial transaction data, which commonly exhibits class imbalance (Mqadi, Naicker, and Adeliyi 2021). Meanwhile, RTF, while a more recent model, shows potential with AUC scores but falls short in F_1 performance, suggesting areas for improvement in financial transaction data synthesis.

This issue is resolved by our SynthRank, evident by consistent improvement compared to baseline models and the original data across all the metrics. SynthRank achieves the highest F_1 scores (0.567 for RF and MLP) and notable AUC scores, with the best P&L values, particularly in the LGBM configuration. This highlights the superiority of our SynthRank pipeline for generating financial transactions. SynthRank not only retains but potentially enhances the predictive performance for identifying risky traders with exceptionally profits, maximising the utility.

[6]https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[7]https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

Data Source	F_1	P&L	AUC
Results with RF Classifier			
Original	0.511	127.738	0.837
Baseline Synthetic Data Generators			
TVAE	0.498	127.060	0.732
CTGAN	0.510	127.833	0.781
CopulaGAN	0.499	127.203	0.775
RTF	0.499	127.098	0.824
Our SynthRank Pipelines			
LMART(XGB:pair)	0.548	128.185	**0.863**
LMART(XGB:ndcg)	0.558	128.985	0.849
LMART(XGB:map)	**0.567**	129.160	0.858
LMART(LGBM)	0.566	**131.269**	0.857
Results with MLP Classifier			
Original	0.522	127.968	0.838
Baseline Synthetic Data Generators			
TVAE	0.509	127.539	0.648
CTGAN	0.531	124.721	0.742
CopulaGAN	0.517	125.907	0.729
RTF	0.498	127.131	0.805
Our SynthRank Pipelines			
LMART(XGB:pair)	0.543	131.207	**0.862**
LMART(XGB:ndcg)	0.536	129.858	0.848
LMART(XGB:map)	0.560	128.451	0.845
LMART(LGBM)	**0.567**	**131.547**	0.856

Table 1: Performance comparison of RF and MLP models on the original dataset versus synthetic datasets generated using different benchmarking models.

5.3 Privacy Evaluation: Privacy Against Inference Test

While generating financial transaction data, our primary focus will be on privacy considerations. In other words, we aim to share a dataset without compromising information about any specific entity within it (Assefa et al. 2021).

For assessing privacy protection, we employ the PAI Test, where attackers, equipped with machine learning or deep learning techniques, gain access to synthetic datasets alongside specific real index features, the true values of which are known to them. These attackers, having trained on the synthetic data, aim to deduce sensitive features in the actual dataset using the provided information.

In this experiment, we assume attackers with access to *accountid*, *Period*, and *MarketCluster* attempt to infer sensitive information such as age and transaction size. Specifically, we choose synthetic data generated by SynthRank with LambdaMART (LGBM) as a representative implementation. To ensure robustness, we implement three attackers based on various architectures: K-Nearest-Neighbour (KNN), Random Forest (RF), and Multi-layer Perception (MLP) models. We also conduct a 10-fold cross-validation using 10K randomly selected samples from the real data and the entire synthetic dataset, which contains over 500K samples per fold. Table 2 presents the privacy evaluation results for synthetic transactions produced by the benchmarking methods.

The results shows that SynthRank consistently outperforms the baseline models across all three attacker mod-

Model	RF	KNN	MLP
TVAE	0.4510	0.5155	0.4414
CTGAN	0.4374	0.5021	0.4311
CopulaGAN	0.4288	0.4850	0.4293
RTF	0.3973	0.1442	0.4324
SynthRank	**0.9453**	**0.9453**	**0.9117**

Table 2: Average PAI scores for synthetic data from baseline models and SynthRank with LambdaMART (LGBM).

els, achieving scores close to 0.9453 in both KNN and RF, and around 0.9117 for the MLP model. Contrastingly, other models like CTGAN, TVAE, CopulaGAN, and RTF demonstrate varying vulnerabilities. TVAE, with the best performance among the baseline models, shows fluctuating PAI scores between 0.4414 and 0.5021, indicating potential exposure to advanced attacks.

To determine the exact value of PAI score indicating a strong defense against inference attack, we calculate the random guessing threshold. This threshold represents a reference line above which an attacker's performance is worse than random guessing. Since our dataset comprises five age groups and three segment groups, a random guessing model assigns a 0.2 and $1/3$ probability for predicting each age group and segment group respectively. The expected accuracy for such a model, per instance, is calculated as $(0.2+1/3)/2 = 0.2667$, regardless of the actual distribution of these categorical features. Hence, the random guessing threshold is $1 - 0.2667 = 0.7333$.

SynthRank exceeds the threshold, which implies that the generated transaction data offers a stronger privacy resilience against information leakage attempts by attackers. Consequently, when the primary concern is safeguarding sensitive attributes, SynthRank appears to be a more suitable choice compared to other baseline models.

5.4 Similarity Evaluation: Statistics and Visualisation

To thoroughly assess how closely the synthetic transaction data mirrors original feature distributions, we use both statistical tests and visual methods. For equitable comparison, both the original and synthetic data are standardised prior to subsequent evaluations.

Feature Type	TVAE	CTGAN	CoGAN	RTF	SynthRank
Continuous	0.188	0.147	0.152	**0.129**	0.153
Discrete	0.082	0.097	0.145	**0.003**	0.460
All	0.128	0.119	0.148	**0.058**	0.326

Table 3: Average KS test statistics of continuous and discrete features. Notably, "CoGAN" stands for "CopulaGAN".

KS Test The statistical results are shown in Table 3. A small KS statistic implies that the two samples are likely drawn from the same distribution.

KDE Plots Figure 2 provides a visualised comparison for 10 selected features (5 continuous and 5 discrete) due to the page limit. Each subplot corresponds to a distinct feature, overlaying the synthetic data distribution over the original data's distribution (in red).

Q-Q Plots As illustrated in Figure 3, we selected 10 continuous features, trying to avoid overlap with those presented in the KDE plots. If the datasets share the same distribution, the points on the Q-Q plot will align with the line $y = x$.

Continuous Features Table 3 shows RTF as the strongest performer in modelling continuous features, with average KS test statistics of 0.129, indicating a high similarity to the original dataset. This highlights the effectiveness of RTF in capturing the distribution of continuous individual's financial behaviour. This is visually corroborated by the well-aligned Q-Q plots of RTF in Figure 3. SynthRank, while achieving similar performance as CopulaGAN and CTGAN in KS statistics, exhibits acceptable alignment with the original dataset for continuous features.

Discrete Features For discrete features, the transformation into continuous forms leads to elevated KS statistics for SynthRank compared to other baselines. Despite this, SynthRank demonstrates its ability to capture the peaks of the original discrete distributions, as seen in Figure 2. This ability is particularly relevant in financial contexts where precise replication of the original distribution is not always essential, and discrete features often contain sensitive information. SynthRank effectively balances data privacy concerns with the need to mirror distributions.

5.5 Trade-offs implications: Utility, Privacy, and Similarity

Utility vs. Similarity Given the aforementioned analysis, a critical trade-off can be observed between utility and feature similarity. RTF, as the latest baseline model, while achieving high similarity (KS statistic of 0.058), compromises on utility with a lower F_1 score (0.498 with the RF classifier). In contrast, SynthRank enhances utility with an F_1 score up to 0.567 with RF, despite not perfectly replicating every feature distribution, indicating a preferable balance for predictive accuracy in financial data.

Privacy vs. Similarity Balancing feature similarity and privacy is challenging. While RTF and CTGAN show high similarity, they potentially weaken the privacy protection with PAI score gaps of over 0.4 compared to SynthRank. Conversely, SynthRank, with a moderate similarity (average KS statistic of 0.326), significantly improves privacy preservation (PAI scores > 0.9 against all the attackers), representing a strategic balance in protecting sensitive information.

Utility vs. Privacy SynthRank demonstrates an exceptional balance between utility and privacy. It prioritises privacy security without compromising on predictive utility, making it highly suitable for financial transaction data. It combines strong privacy measures (evidenced by high PAI scores) with high predictive utility (up to 0.567 F_1 score with the MLP classifier), addressing the critical need for both data protection and predictive accuracy in financial contexts.

6 Conclusion and Future Work

In conclusion, this study marks the first attempt towards using learning-to-rank algorithms for the generation of synthetic financial transaction data. By re-envisioning the synthetic data generation paradigm, we strategically prioritise a

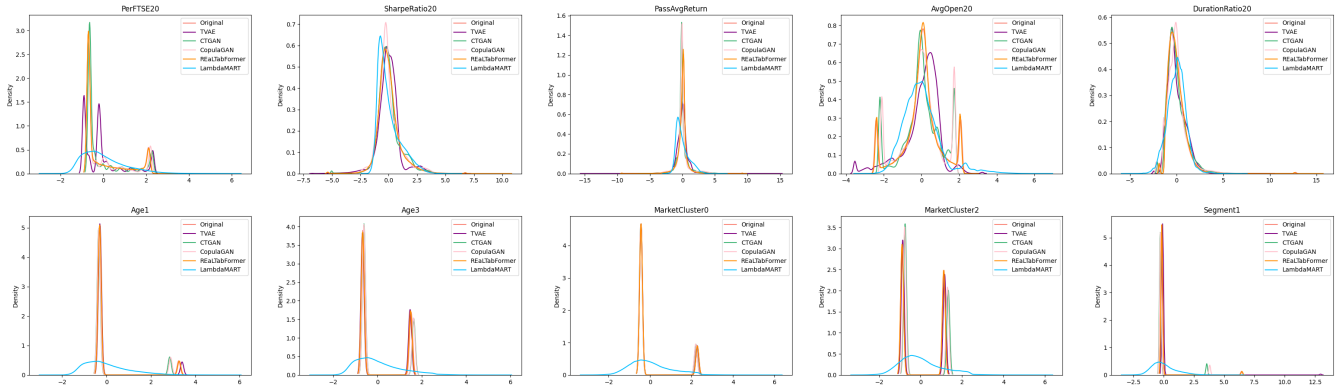

Figure 2: KDE Plots of 10 Selected Features from Original and Synthetic Datasets.

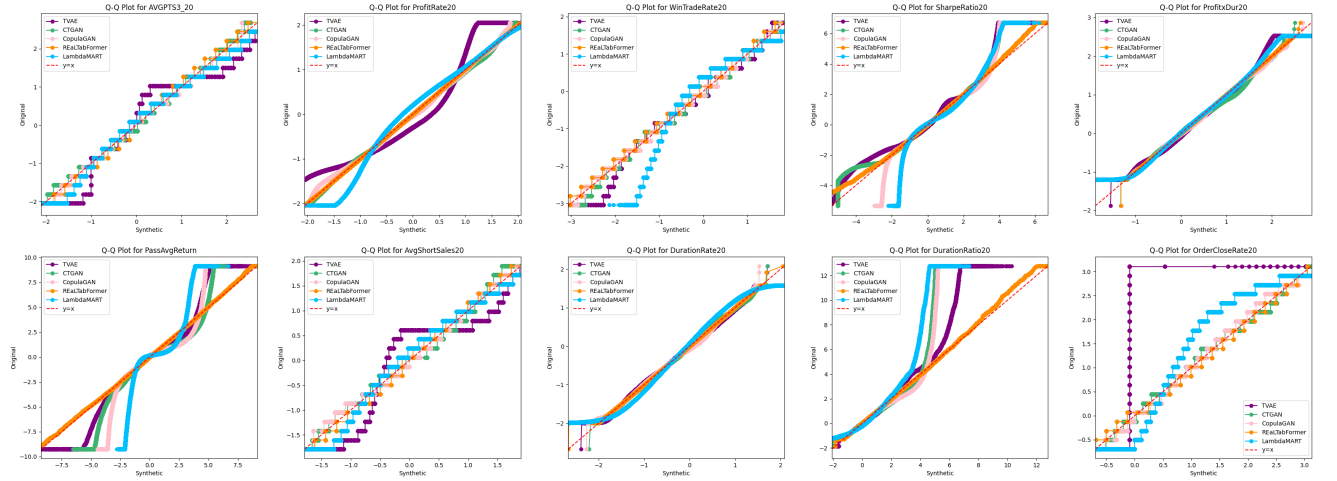

Figure 3: Q-Q Plots for Original vs. Synthetic Datasets Generated by benchmarking models on 10 Selected Continuous Features.

task-oriented synthetic data generation. Therefore, we introduce the ranking embeddings, which emerged as a potent solution in amplifying predictive capabilities. Empirical evaluations underscored the merit of this strategy, as evidenced by the enhanced efficacy in predicting the top 1% risky traders.

Additionally, our findings shed light on an interesting observation: while the synthetic data produced using ranking models might not exactly mirror the original distribution, it remarkably preserves pivotal characteristics. This assertion is substantiated by comprehensive statistical assessments that showcased the synthetic data's fidelity to the original, especially in capturing underlying trends and patterns.

Moreover, in the context of financial transaction data, which often contains sensitive information, ranging from personal identifiers to transactional specifics, the priority shifts towards ensuring privacy and utility. By obfuscating these sensitive facets, our SynthRank approach aligns with this requirement. Its ability to provide accurate predictions while securing sensitive data positions it as a preferred choice in this scenario. Therefore, while statistical similarity is a key metric, it should be balanced with task-specific needs and privacy considerations in financial applications.

Moving forward, we observe the absence of domain-specific insights in our approach. While the predictor achieves the highest F_1, it might not guarantee to maximise P&L at the same time. Our future work will focus on refining the ranking models and infusing them with domain-specific knowledge to bridge this gap.

Furthermore, while we have proposed an innovative approach to synthetic data generation that goes beyond merely replicating the original data distribution, it's important to acknowledge that LETOR algorithms may just be a stepping stone in this journey. There's a vast landscape of methodologies awaiting exploration, and future research should remain open to alternative techniques that could provide enhanced robust and privacy-preserving synthetic datasets.

Acknowledgement

This research was supported by The Alan Turing Institute and the Engineering and Physical Sciences Research Council (EPSRC). We extend our sincere gratitude to both The Alan Turing Institute and EPSRC for their vital support and contributions to our research endeavours.

References

Assefa, S. A.; Dervovic, D.; Mahfouz, M.; Tillman, R. E.; Reddy, P.; and Veloso, M. 2021. Generating Synthetic Data in Finance: Opportunities, Challenges and Pitfalls. In *Proceedings of the First ACM International Conference on AI in*

Finance, ICAIF '20. New York, NY, USA: Association for Computing Machinery. ISBN 9781450375849.

Borisov, V.; Seßler, K.; Leemann, T.; Pawelczyk, M.; and Kasneci, G. 2023. Language Models are Realistic Tabular Data Generators. arXiv:2210.06280.

Buyl, M.; Missault, P.; and Sondag, P.-A. 2023. RankFormer: Listwise Learning-to-Rank Using Listwide Labels. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, 3762–3773. New York, NY, USA: Association for Computing Machinery. ISBN 9798400701030.

Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; and Kegelmeyer, W. P. 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16: 321–357.

Chen, J.; Liao, K.; Wan, Y.; Chen, D. Z.; and Wu, J. 2022. DANets: Deep Abstract Networks for Tabular Data Classification and Regression. In *AAAI*.

Chen, T.; and Guestrin, C. 2016. XGBoost: A Scalable Tree Boosting System. *CoRR*, abs/1603.02754.

Choi, E.; Biswal, S.; Malin, B.; Duke, J.; Stewart, W. F.; and Sun, J. 2017. Generating Multi-label Discrete Patient Records using Generative Adversarial Networks. In Doshi-Velez, F.; Fackler, J.; Kale, D.; Ranganath, R.; Wallace, B.; and Wiens, J., eds., *Proceedings of the 2nd Machine Learning for Healthcare Conference*, volume 68 of *Proceedings of Machine Learning Research*, 286–305. PMLR.

Espinosa, E.; and Figueira, A. 2023. On the Quality of Synthetic Generated Tabular Data. *Mathematics*, 11(15).

Fu, R.; Chen, J.; Zeng, S.; zhuang, y.; and Sudjianto, A. 2019. Time Series Simulation by Conditional Generative Adversarial Net. *SSRN Electronic Journal*.

Fuhr, N. 1989. Optimum Polynomial Retrieval Functions Based on the Probability Ranking Principle. *ACM Trans. Inf. Syst.*, 7(3): 183–204.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Goodfellow, I. J.; Bengio, Y.; and Courville, A. 2016. *Deep Learning*. Cambridge, MA, USA: MIT Press. http://www.deeplearningbook.org.

Hand, D. J. 2018. Statistical Challenges of Administrative and Transaction Data. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 181(3): 555–605.

Hilal, W.; Gadsden, S. A.; and Yawney, J. 2022. Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances. *Expert Systems with Applications*, 193: 116429.

Jayaraman, B.; and Evans, D. 2019. Evaluating Differentially Private Machine Learning in Practice. In *Proceedings of the 28th USENIX Conference on Security Symposium*, SEC'19, 1895–1912. USA: USENIX Association. ISBN 9781939133069.

Jordon, J.; Szpruch, L.; Houssiau, F.; Bottarelli, M.; Cherubin, G.; Maple, C.; Cohen, S. N.; and Weller, A. 2022. Synthetic Data – what, why and how? arXiv:2205.03257.

Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; and Liu, T.-Y. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Kim, A.; Yang, Y.; Lessmann, S.; Ma, T.; Sung, M.-C.; and Johnson, J. 2020. Can deep learning predict risky retail investors? A case study in financial risk behavior forecasting. *European Journal of Operational Research*, 283(1): 217–234.

Kim, S.; Choi, K.; Choi, H.-S.; Lee, B.; and Yoon, S. 2021. Towards a Rigorous Evaluation of Time-series Anomaly Detection. In *AAAI Conference on Artificial Intelligence*.

Kiran, A.; and Kumar, S. S. 2023. A Comparative Analysis of GAN and VAE based Synthetic Data Generators for High Dimensional, Imbalanced Tabular data. In *2023 2nd International Conference for Innovation in Technology (INOCON)*, 1–6.

Kotelnikov, A.; Baranchuk, D.; Rubachev, I.; and Babenko, A. 2023. TabDDPM: Modelling Tabular Data with Diffusion Models. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 17564–17579. PMLR.

Kozlov, N. 2014. Contracts for difference: risks faced by generators under the new renewables support scheme in the UK. *The Journal of World Energy Law Business*, 7(3).

Ma, T.; Fraser-Mackenzie, P.; Sung, M.; Kansara, A.; and Johnson, J. 2022. Are the least successful traders those most likely to exit the market? A survival analysis contribution to the efficient market debate. *European Journal of Operational Research*, 299(1): 330–345.

Massey, F. J. 1951. The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association*, 46(253): 68–78.

Mqadi, N.; Naicker, N.; and Adeliyi, T. 2021. Solving Misclassification of the Credit Card Imbalance Problem Using Near Miss. *Mathematical Problems in Engineering*, 2021.

Nickerson, K.; Tricco, T.; Kolokolova, A.; Shoeleh, F.; Robertson, C.; Hawkin, J.; and Hu, T. 2023. Banksformer: A Deep Generative Model for Synthetic Transaction Sequences. In Amini, M.-R.; Canu, S.; Fischer, A.; Guns, T.; Kralj Novak, P.; and Tsoumakas, G., eds., *Machine Learning and Knowledge Discovery in Databases*, 121–136. Cham: Springer Nature Switzerland. ISBN 978-3-031-26422-1.

Pang, L.; Xu, J.; Ai, Q.; Lan, Y.; Cheng, X.; and Wen, J. 2020. Setrank: Learning a permutation-invariant ranking model for information retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 499–508.

Paula, E. L.; Ladeira, M.; Carvalho, R. N.; and Marzagão, T. 2016. Deep Learning Anomaly Detection as Support Fraud

Investigation in Brazilian Exports and Anti-Money Laundering. In *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 954–960.

Phophalia, A. 2011. A survey on Learning to Rank (LETOR) approaches in information retrieval. 1–6. ISBN 978-1-4577-2169-4.

Ram Mohan Rao, P.; Murali Krishna, S.; and Siva Kumar, A. P. 2018. Privacy preservation techniques in big data analytics: a survey. *Journal of Big Data*, 5(1): 33.

Rizzato, M.; Wallart, J.; Geissler, C.; Morizet, N.; and Boumlaik, N. 2023. Generative Adversarial Networks applied to synthetic financial scenarios generation. *Physica A: Statistical Mechanics and its Applications*, 623: 128899.

Solatorio, A. V.; and Dupriez, O. 2023. REaLTabFormer: Generating Realistic Relational and Tabular Data using Transformers. arXiv:2302.02041.

Strelcenia, E.; and Prakoonwit, S. 2022. Generating Synthetic Data for Credit Card Fraud Detection Using GANs. In *2022 International Conference on Computers and Artificial Intelligence Technologies (CAIT)*, 42–47.

Wiese, M.; Knobloch, R.; Korn, R.; and Kretschmer, P. 2020. Quant GANs: deep generation of financial time series. *Quantitative Finance*, 20(9): 1419–1440.

Xu, L.; Skoularidou, M.; Cuesta-Infante, A.; and Veeramachaneni, K. 2019. *Modeling Tabular Data Using Conditional GAN*. Red Hook, NY, USA: Curran Associates Inc.

Yale, A.; Dash, S.; Dutta, R.; Guyon, I.; Pavao, A.; and Bennett, K. P. 2020. Generation and evaluation of privacy preserving synthetic health data. *Neurocomputing*, 416.

Zhang, J.; Cormode, G.; Procopiuc, C. M.; Srivastava, D.; and Xiao, X. 2017. PrivBayes: Private Data Release via Bayesian Networks. *ACM Trans. Database Syst.*, 42(4).

Zhao, B.; Agrawal, A.; Coburn, C.; Asghar, H.; Bhaskar, R.; Kaafar, M.; Webb, D.; and Dickinson, P. 2021a. On the (In)Feasibility of Attribute Inference Attacks on Machine Learning Models. In *2021 IEEE European Symposium on Security and Privacy (EuroSamp;P)*, 232–251. Los Alamitos, CA, USA: IEEE Computer Society.

Zhao, Z.; Kunar, A.; Birke, R.; and Chen, L. Y. 2021b. CTAB-GAN: Effective Table Data Synthesizing. In Balasubramanian, V. N.; and Tsang, I., eds., *Proceedings of The 13th Asian Conference on Machine Learning*, volume 157 of *Proceedings of Machine Learning Research*, 97–112. PMLR.

Zheng, P.; Yuan, S.; Wu, X.; Li, J.; and Lu, A. 2019. One-class adversarial nets for fraud detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 1286–1293.