

LATTE: Learning Aligned Transactions and Textual Embeddings for Bank Clients

Egor Fadeev¹, Dzhambulat Mollaev¹, Aleksei Shestov¹, Dmitry Korolev¹, Omar Zoloev¹, Ivan Kireev¹, Andrey Savchenko^{1,2}, Maksim Makarenko¹

¹Sber AI Lab, Moscow, Russia

²ISP RAS Research Center for Trusted Artificial Intelligence, Moscow, Russia

Abstract

Learning clients embeddings from sequences of their historic communications is central to financial applications. While large language models (LLMs) offer general world knowledge, their direct use on long event sequences is computationally expensive and impractical in real-world pipelines. In this paper, we propose **LATTE**, a contrastive learning framework that aligns raw event embeddings with description-based semantic embeddings from frozen LLMs. Behavioral features based on statistical user descriptions are summarized into short prompts, embedded by the LLM, and used as supervision via contrastive loss. The proposed approach significantly reduces inference cost and input size compared to the conventional processing of complete sequences by LLM. We experimentally show that our method outperforms state-of-the-art techniques for learning event sequence representations on real-world financial datasets while remaining deployable in latency-sensitive environments.

1 Introduction

Research in natural language processing (NLP) has traditionally focused on unstructured text (Bagheri et al., 2023). In contrast, many industrial applications of healthcare (Wang et al., 2024b), education (Liu et al., 2023), e-commerce (Dai et al., 2023; Liu et al., 2025), and, especially, finance (Babaev et al., 2019; Luetto et al., 2025), generate hundreds of streams of structured event (temporally ordered, high-dimensional, and often sparse tabular) data (Osin et al., 2025), such as transaction logs, payment histories, and customer interactions, which are sequential, high-dimensional, and sparse (Zhang et al., 2023; Osin et al., 2025). These data underpin a broad spectrum of business-critical tasks, including churn prediction, risk assessment, credit scoring, and personalized targeting (Mollaev et al., 2025). Transaction sequences differ from

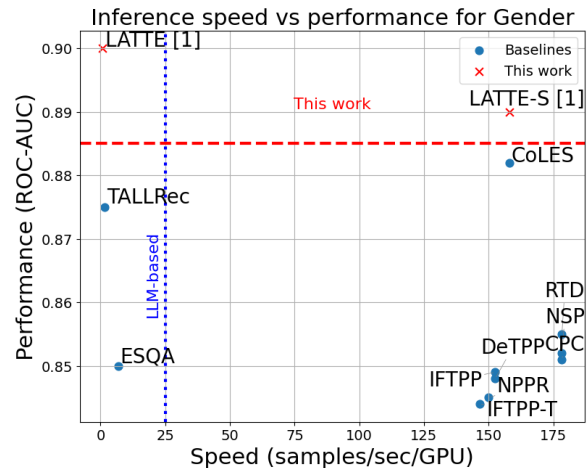


Figure 1: Figure of Merit comparing ROC-AUC performance and inference speed (samples/sec/GPU) on the Gender prediction task. Compared methods include **LATTE** [1], **LATTE-S** [1], CoLES (Babaev et al., 2022) RTD (Clark et al., 2019), CPC (Oord et al., 2018), NSP (Devlin et al., 2019), NPPR (Skalski et al., 2023), DeTPP (Karpukhin and Savchenko, 2024), IFTPP (Shchur et al., 2020), IFTPP-T (Shchur et al., 2020), ESQA (Abdullaeva et al., 2024), and TALL-Rec (Bao et al., 2023).

text in three key ways: they are much longer (thousands of events in open datasets, millions in proprietary banking logs) (Mollaev et al., 2025), each event includes multiple categorical and numerical attributes (Zhang et al., 2023), and the main tasks are classification or regression rather than broad semantic benchmarks (Muennighoff et al., 2023).

Recent works on applying Large Language Models (LLMs) for structured data (Shi et al., 2023; Yu et al., 2025) highlight that progress in this domain depends on methods adapted to the unique statistical and causal structure of event data, rather than direct transfer of techniques from NLP. Moreover, a direct application of LLMs to serialized sequential tabular data incurs substantial computational overhead due to the large token counts per user. For example, typical banking transaction datasets often contain hundreds of records per user,

each serialized into dozens of tokens, exceeding practical context window limits and increasing inference and training times (Shestov et al., 2025). Figure 1 shows that LLM-based models such as TALLRec (Bao et al., 2023) and ESQA (Abdullaeva et al., 2024) do not exceed an inference speed of 10 users per second, which severely limits their applicability in banking production environments.

To address the limitations of existing techniques, we propose **LATTE**¹, a scalable framework for Learning Aligned Transactions and Textual Embeddings. Instead of feeding entire sequences into LLMs, we extract compact client-level statistics and use an instruction-tuned LLM to generate natural language summaries. These summaries serve as weak labels, aligned with pretrained sequence embeddings from a lightweight encoder via contrastive learning. At inference, **LATTE** supports two modes: a standalone encoder (**LATTE-S**) that retains LLM-level semantics without added overhead, and a combined encoder (**LATTE**) that fuses textual representations enriched with statistical features and structural embeddings.

To evaluate trade-offs between performance and efficiency, we introduce a Figure of Merit (FOM) comparing ROC-AUC and inference speed on the standard banking dataset of gender prediction task (Sberbank, 2021a) (Figure 1). We evaluate **LATTE** in two variants, combining two embedding strategies: structural-only (**LATTE-S**) and concatenated with textual features (**LATTE**). In terms of performance, all versions of **LATTE** outperform the typical baseline financial methods. The structural-only variant (**LATTE-S**) achieves a ROC-AUC of 0.891 on the Gender task with an inference speed of 162 samples/sec/GPU, surpassing LLM-based models such as TALLRec and ESQA, being over 14× faster. Across all types of textual encoders, the combined variant (**LATTE**) consistently achieves the highest overall ROC-AUC.

2 Related Works

Learning representations from structured event sequences (Udovichenko et al., 2024; Yeshchenko and Mendling, 2022; Kolosnjaji et al., 2016; Weiss and Hirsh, 1998; Guo et al., 2020) remains a core challenge in industrial applications. Despite abundant customer interaction data, high-quality labels for event sequences in most typical downstream

tasks (campaigning, churn prediction, etc.) remain limited (Mollaev et al., 2025). This shortage of timely supervision hinders the scalability of supervised learning in production settings. It highlights the need for self-supervised approaches to derive robust and semantically rich representations directly from raw behavioral sequences (Gui et al., 2024). Prevailing self-supervised approaches for modeling event sequences adopt such objectives as contrastive learning (Babaev et al., 2022), next-event prediction (Skalski et al., 2023), and latent sequence modeling techniques, e.g., Contrastive Predictive Coding (CPC) (Oord et al., 2018), aiming to capture temporal dependencies and user intent without relying on manual supervision.

Appearance of LLMs offer new opportunities to enhance representation learning from event sequences. Trained on diverse and large-scale corpora, LLMs encode elements often implicit or absent in structured event datasets, e.g., rich semantic priors about behavioral patterns, temporal dynamics, and domain knowledge. Leveraging this external knowledge can significantly improve the quality of user representations, particularly in financial applications (Ruan et al., 2024). Motivated by this potential, recent studies have explored adaptations of LLMs to structured data. For example, TALLRec (Bao et al., 2023), LLM-TRSR (Zheng et al., 2024), and HKFR (Yin et al., 2023) transfer rich text understanding abilities of LLMs to recommender systems; TabLLM (Hegselmann et al., 2023) targets tabular classification tasks; TEST (Sun et al., 2024) and Time-LLM (Jin et al., 2024) address time series; while ESQA (Abdullaeva et al., 2024) applies LLMs to event-sequence question answering.

Existing methods to mitigate this issue fall into two main categories. The first reduces context length by summarizing user histories with general-purpose LLMs (Yin et al., 2023; Zheng et al., 2024), which risks losing domain-specific information critical for accurate modeling. The second class of methods attempts to bypass context length constraints by encoding sequences in non-textual formats (Sun et al., 2024; Jin et al., 2024). However, these representations often discard the semantic content present in item descriptions (e.g., transaction categories or merchant details). Moreover, as user histories grow longer, these models either incur increased computational cost or suffer performance degradation due to limited model capacity.

Recent advances in event sequence modeling re-

¹The source code is available at <https://github.com/tsebaka/LATTE>

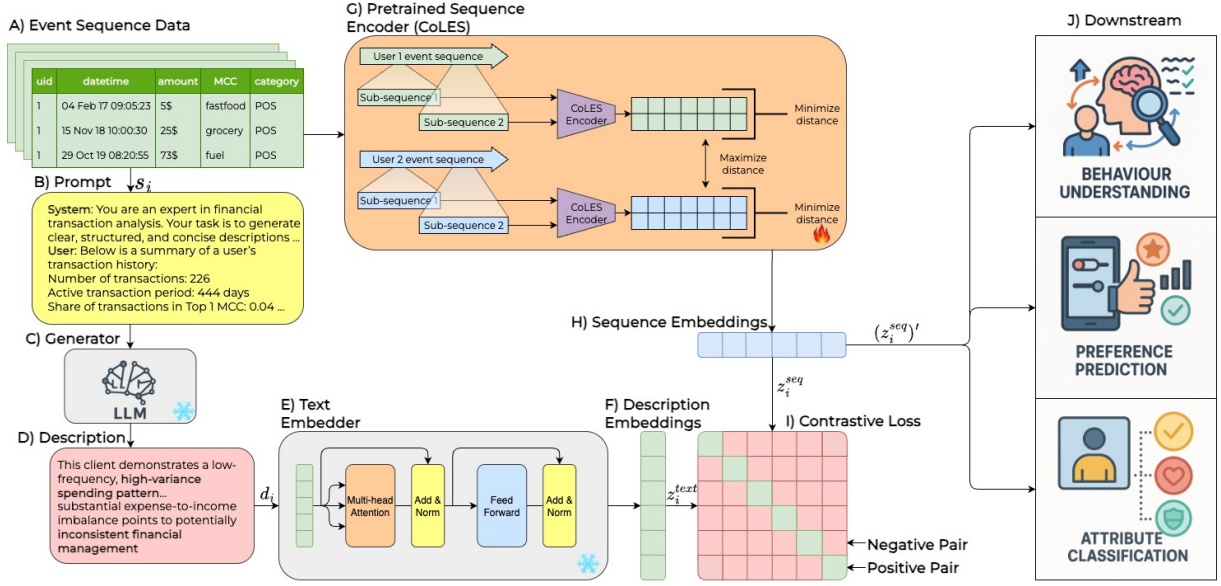


Figure 2: Overview of the proposed **LATTE** pipeline. (a) Event sequences serve as the input data source. (b) A summary prompt is crafted to query the event sequence. (c) An LLM generator produces a natural language description based on the prompt. (d) The resulting textual description captures salient features of the event sequence. (e) A text embedder converts the description into a vector representation. (f) This description embedding encodes the generated text. (g) In parallel, a sequence encoder embeds the original event sequence. (h) The resulting sequence embedding captures structural and temporal patterns. (i) A contrastive alignment module trains the model to align textual and sequence embeddings in a shared representation space. (j) The aligned embeddings can be used for various downstream tasks such as classification, retrieval, or prediction.

inforce the distinction between unstructured texts from NLP and structured data. Work on spatio-temporal clustering shows that standard neural point process models fail to capture hierarchical spatial structures and multi-type dependencies, requiring new architectures tailored to these properties (Yu et al., 2025). Other studies demonstrate that even transformer-based approaches underperform when causal relations between event types are ignored, motivating causality-aware attention mechanisms (Shou et al., 2023). Further results indicate that while Large Language Models (LLMs) can aid event prediction through abductive reasoning, they are effective only when combined with specialized sequence models (Shi et al., 2023). Together, these findings highlight that progress in this domain depends on methods adapted to the unique statistical and causal structure of event data, rather than direct transfer of techniques from NLP.

3 Proposed Approach

We aim to improve the quality of representations learned from transactional event sequences by introducing an auxiliary textual modality that verbalizes statistical properties of user behavior. To this end, we propose a three-stage pipeline, **LATTE**,

illustrated in Figure 2, which maps raw transaction sequences into rich embeddings suitable for downstream tasks.

Let $T_i = x_1, x_2, \dots, x_n$ denote the transaction sequence for client i , where each x_j contains time-stamped categorical and numerical attributes (e.g., amount, merchant category). As shown in Fig. 2a, we first compute a vector of summary features s_i that aggregates behavioral patterns over the sequence: frequency of activity, merchant diversity, transaction types, temporal coverage, and income-expense structure. Behavioral features are then transformed into meaningful textual descriptions rather than raw indices, allowing these summaries to be further enriched with the semantic knowledge of the LLM. The prompt template and a sample generated description are presented in Appendix C. This profile is then serialized into a textual prompt (Fig. 2b), which is passed to a frozen instruction-tuned LLM (Fig. 2c) to generate a natural language description d_i of the client’s behavior (Fig. 2d). The prompt includes a system message and a structured representation of s_i designed to elicit coherent and interpretable responses.

Simultaneously, the raw sequence T_i is processed by a GRU-based sequence encoder (Fig. 2g)

trained under a self-supervised CoLES objective (Babaev et al., 2022), where each training example consists of two overlapping subsequences (positives) and contrastive negatives from other clients. This produces the sequence embedding z_i^{seq} optimized to capture client-specific behavioral dynamics. In parallel, the description d_i is passed through a frozen multilingual sentence encoder (Fig. 2f) with mean pooling, producing the text embedding z_i^{text} enriched with raw statistical features derived from the original sequence.

The embeddings z_i^{seq} and z_i^{text} are then aligned using one of two cross-modal contrastive losses (Fig. 2i), while keeping the text encoder fixed. The updated transaction embeddings $(z_i^{\text{seq}})'$ (Fig. 2h), aligned with the textual representations, are evaluated on downstream tasks such as churn prediction (Fig. 2j). Appendix A provides additional details regarding the pipeline.

To perform cross-modal alignment, we introduce two different contrastive heads: Symmetric Softmax and Orthogonal Regularized. Each was inspired by prior work on multimodal learning (Radford et al., 2021; Jiang et al., 2023). These heads differ in their alignment geometry and regularization, and we refer to them as **LATTE [1]**, and **LATTE [2]**, respectively. Both heads use frozen text encoders and update only the transaction encoder. Downstream tasks are evaluated using the resulting $(z_i^{\text{seq}})'$, while the textual modality provides semantically grounded alignment by enriching the training signal with contextual knowledge of categorical attributes that the sequence encoder alone cannot interpret.

LATTE [1]: Symmetric Softmax Contrastive Head promotes bidirectional alignment between modalities using a symmetric InfoNCE-style loss:

$$\mathcal{L}_{\text{softmax}} = \frac{1}{2}(\mathcal{L}_{\text{seq} \rightarrow \text{text}} + \mathcal{L}_{\text{text} \rightarrow \text{seq}}), \quad (1)$$

where

$$\mathcal{L}_{\text{seq} \rightarrow \text{text}} = -\frac{1}{N} \sum_{i=1}^N \log \frac{\exp(\langle z_i^{\text{seq}}, z_i^{\text{text}} \rangle / \tau)}{\sum_{j=1}^N \exp(\langle z_i^{\text{seq}}, z_j^{\text{text}} \rangle / \tau)}.$$

The second term, $\mathcal{L}_{\text{text} \rightarrow \text{seq}}$, is defined similarly with the roles of sequence and text reversed. Here, $\langle \cdot, \cdot \rangle$ denotes the cosine similarity between the L2-normalized embeddings of the sequence (z_i^{seq}) and the corresponding text (z_i^{text}), and τ is a temperature hyperparameter controlling the sharpness of

the similarity distribution.

LATTE [2]: Orthogonal Regularized Contrastive Head augments the softmax-based loss with a geometric regularization term that disentangles modality-specific and shared information. To achieve this, we introduce an auxiliary projection head that maps each transaction embedding z_i^{seq} to a representation composed of two parts: Z^{shared} , which captures components aligned with textual information, and Z^{spec} , which preserves information specific to the transaction modality:

$$\mathcal{L}_{\text{reg}} = \mathcal{L}_{\text{softmax}} + \lambda_{\text{ortho}} \cdot \mathcal{L}_{\text{ortho}}, \quad (2)$$

where $\mathcal{L}_{\text{ortho}} = \|(Z^{\text{shared}})^\top Z^{\text{spec}}\|_F^2$.

This separation promotes disentangled features by penalizing correlation between shared and specific subspaces, where λ_{ortho} controls the strength of this regularization and thus the emphasis on preserving modality-specific information.

4 Experimental Setup

This section provides the core details of our experimental setup, including validation strategy, datasets, and baseline methods. Additional implementation details are provided in Appendix A.

4.1 Data

We evaluate our method on three real-world datasets containing anonymized credit card transaction sequences from major financial institutions. Each dataset comprises client-level sequences with numerical and categorical attributes (e.g., amount, merchant category, transaction type), and includes an unlabeled subset used exclusively for representation learning. *Churn* (Rosbank, 2021) includes approximately 10K Rosbank clients labeled by future inactivity. *Gender* (Sberbank, 2021a) and *Age Group* (Sberbank, 2021b), provided by Sberbank, contain 15K and 50K clients respectively, annotated with demographic labels.

4.2 Validation Strategy

Each dataset is split into disjoint training and test partitions by reserving 10% of the labeled clients for evaluation. The remaining 90% of labeled users, together with all available unlabeled users, are used for training the embedding models. To assess the quality of learned representations, we adopt a 5-fold cross-validation procedure. Specifically, the labeled portion of the training data is divided into five equal-sized folds. For each fold v , we: (1)

train a LightGBM (Ke et al., 2017) classifier on embeddings from the remaining four folds, and (2) evaluate it on the held-out test fold, computing a downstream performance metric M_v . For binary classification tasks (churn, gender), we report ROC-AUC; for multiclass age prediction, we report classification accuracy. The final performance is summarized as $\mu_M \pm \sigma_M$, where μ_M is the mean and σ_M is the standard deviation over all five folds.

4.3 Baselines

We compare **LATTE** against a diverse set of baselines spanning five methodological families:

Event sequence models. CPC (Oord et al., 2018) learns to predict future representations from past context via a contrastive loss. CoLES (Babaev et al., 2022) improves temporal consistency by aligning overlapping subsequences using InfoNCE. NPPR (Skalski et al., 2023) employs autoregressive training with dual objectives: predicting the next and reconstructing the previous event from masked sequences. All models operate on raw event sequences without external modalities.

Temporal point process models. DeTPP (Karpukhin and Savchenko, 2024) models event timing and types using parametric point processes. IFTPP and IFTPP-T (Shchur et al., 2020) use Transformer and GRU backbones with combined MAE and classification losses.

Natural language processing objectives. RTD (Clark et al., 2019) randomly replaces 15% of event tokens and predicts the original token; NSP (Devlin et al., 2019) extends BERT’s next sentence prediction to sequences of events.

LLM-based approaches. We adapt TALL-Rec (Bao et al., 2023) and HKFR (Yin et al., 2023) from recommender systems for user embedding extraction by fine-tuning LLMs on serialized user-item histories using next-token prediction. Embeddings are derived via mean pooling over the final layer, following recent best practices (BehnamGhader et al., 2024; Muenighoff et al., 2024). Additionally, we replace older backbones (e.g., LLaMA 7B (Touvron et al., 2023), ChatGLM-6B (Du et al., 2022)) with LLaMA 3.2 3B (Touvron et al., 2024) to leverage architectural improvements and efficiency.

Tabular feature aggregation. As a non-sequential baseline, *agg* aggregates transaction features using summary statistics such as mean, standard deviation, min-max and grouped frequency statistics (for categorical features).

5 Experimental Results

5.1 Main Results

Model	Churn (AUC)	Age Group (Acc)	Gender (AUC)
agg	0.827 $\pm$ 0.010	0.629 $\pm$ 0.002	0.877 $\pm$ 0.004
CPC	0.792 $\pm$ 0.015	0.602 $\pm$ 0.004	0.851 $\pm$ 0.006
RTD	0.771 $\pm$ 0.016	0.631 $\pm$ 0.006	0.855 $\pm$ 0.008
CoLES	0.841 $\pm$ 0.005	0.644 $\pm$ 0.005	0.882 $\pm$ 0.004
NSP	0.828 $\pm$ 0.012	0.621 $\pm$ 0.005	0.852 $\pm$ 0.011
NPPR	0.845 $\pm$ 0.003	0.642 $\pm$ 0.001	-
DeTPP	0.823 $\pm$ 0.002	0.632 $\pm$ 0.004	-
IFTTP	0.828 $\pm$ 0.004	0.632 $\pm$ 0.003	0.863 $\pm$ 0.003
IFTTP-T	0.814 $\pm$ 0.004	0.620 $\pm$ 0.002	0.852 $\pm$ 0.005
TALLRec	0.839 $\pm$ 0.003	0.659 $\pm$ 0.004	0.875 $\pm$ 0.004
HKFR	0.823 $\pm$ 0.006	-	-
LATTE [1]	0.869 $\pm$ 0.004	0.665 $\pm$ 0.005	0.900 $\pm$ 0.005
LATTE [2]	0.872 $\pm$ 0.004	0.663 $\pm$ 0.003	0.898 $\pm$ 0.006

Table 1: Performance of client embeddings on downstream tasks. **Bold** indicates best result; underline indicates second-best

Table 1 presents the performance of client embeddings on three downstream tasks: churn prediction (AUC), age group classification (accuracy), and gender prediction (AUC). While traditional self-supervised objectives such as CPC, RTD, and NSP lag behind stronger baselines like CoLES and TALLRec, several variants of our proposed LATTE framework demonstrate clear gains across tasks. **LATTE [2]** achieves the best result on churn prediction (0.872 AUC), while **LATTE [1]** leads on both age group classification (0.665 accuracy) and gender prediction (0.900 AUC). These improvements indicate that incorporating statistic-based textual supervision leads to consistently stronger representations of transactional behavior.

5.2 Runtime Analysis

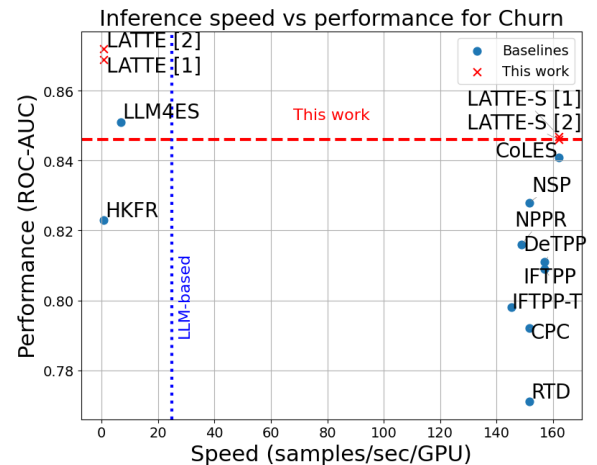


Figure 3: Figure of Merit comparing model performance in ROC-AUC and inference speed in samples per second per GPU on Churn dataset.

In banking applications, inference speed is critical due to the need to process millions of multi-

source user records in a hardware environment with limited access to GPUs. In this section, we investigate an important trade-off between inference resource utilization (compute time, memory utilization) and the performance of the proposed methods compared to baseline architectures.

Figure 3 showcases the performance on the Churn dataset. Generative models (**LATTE**, **TALL-Rec**) achieve the highest ROC-AUC scores, exceeding 0.868, but at the cost of extremely low inference speeds—processing only a few samples per second—and high memory consumption, with over 3 billion parameters. In contrast, lightweight **LATTE-S** variants attain slightly lower ROC-AUC values while operating over 160 samples per second and a compact size of just a few million parameters. These models match the efficiency of contrastive baselines, offering a favorable trade-off between accuracy, speed, and memory footprint.

5.3 Contrastive Fine-tuning Alignment

Method	Churn (AUC)	Age Group (Acc)	Gender (AUC)
Descriptions	0.772 $\pm$ 0.011	0.432 $\pm$ 0.007	0.644 $\pm$ 0.009
CoLES + z^{text}	0.863 $\pm$ 0.007	0.650 $\pm$ 0.004	0.890 $\pm$ 0.003
LATTE-S [1]	0.847 $\pm$ 0.004	0.657 $\pm$ 0.003	0.891 $\pm$ 0.004
LATTE [1]	0.869 $\pm$ 0.004	0.665 $\pm$ 0.005	0.900 $\pm$ 0.005
LATTE-S [2]	0.846 $\pm$ 0.005	0.655 $\pm$ 0.004	0.888 $\pm$ 0.003
LATTE [2]	0.872 $\pm$ 0.004	0.663 $\pm$ 0.003	0.898 $\pm$ 0.006

Table 2: Ablation: Impact of contrastive fine-tuning and modality concatenation on downstream task.

Table 2 presents an ablation study on the effect of contrastive fine-tuning and modality concatenation. Incorporating LLM-generated behavioral descriptions markedly improves downstream performance compared to CoLES, particularly for churn and age prediction tasks. The contrastive alignment step remains crucial: all **LATTE** variants outperform the non-aligned baseline. Among the evaluated heads, **LATTE** [2] achieves the highest AUC on churn (0.872), while **LATTE** [1] attains the best accuracy on age (0.665) and gender (0.900). Nevertheless, unaligned concatenation remains a competitive baseline (CoLES + z^{text}), indicating that the statistical–semantic descriptions alone already provide a strong inductive bias.

5.4 Evaluation of the quality of LLM summarization

In this section, we study whether the textual descriptions generated by the LLM faithfully capture the underlying statistics that were used to construct the prompts. We asked an independent LLM critic (Llama 3.1 8B) to extract key statistics (e.g., domi-

Feature	Churn	Gender	Age
mcc_0 Usage %	35.59	24.61	37.93
mcc_0 Acc %	100	100	100
mcc_1 Usage %	38.98	26.61	39.66
mcc_1 Acc %	100	100	100
trx_period Usage %	100	100	100
trx_period Acc %	98.31	99.02	99.14
trx_days_share Usage %	93.22	95.41	93.97
trx_days_share Acc %	98.18	92.38	99.07

Table 3: Four key LLM statistics usage (%) and accuracy (%) across tasks.

nant merchant categories, transaction period length, share of active days) from the subsample of generated descriptions. We then applied a rule-based matching procedure to compare these extracted factors against the ground-truth statistics.

Table 3 reports both the usage rate (how frequently a given statistic was mentioned in the LLM description) and the accuracy rate (the percentage of mentions that correctly reflect the underlying value) across three downstream tasks. We use 200 random samples per dataset. The results show that core statistics such as transaction_period and transaction_days_share are not only used very frequently (over 90% of cases), but also described with a high accuracy (above 92%). As expected, categorical statistics such as merchant-category (mcc_0, mcc_1) are consistently mentioned with 100% correctness when they appear.

6 Analysis of Behavioral Embedding Structure

Metric	CoLES	LATTE
Total spending	0.627	0.693
Total expense	0.593	0.730
Average daily spending	0.519	0.689
Total spending in MCC (Eating places)	0.705	0.775
First transaction day	0.652	0.622
Std of daily spending	0.858	0.789

Table 4: Separability of user groups in the lowest 1% and highest 99% quantiles of behavioral statistics using logistic regression. **LATTE** embeddings show stronger separation across most statistics compared to CoLES.

In this section, we investigate how behavioral statistics are encoded within the embedding spaces of CoLES and **LATTE** (Figure 4). For each statistic, we highlight clients belonging to the lowest 1% (blue) and highest 99% (red) quantiles, and visualize the resulting structure using a UMAP projection with an overlaid logistic regression decision boundary. Compared to CoLES, **LATTE** embeddings exhibit more distinct geometric separation across representative behavioral dimensions—such as total

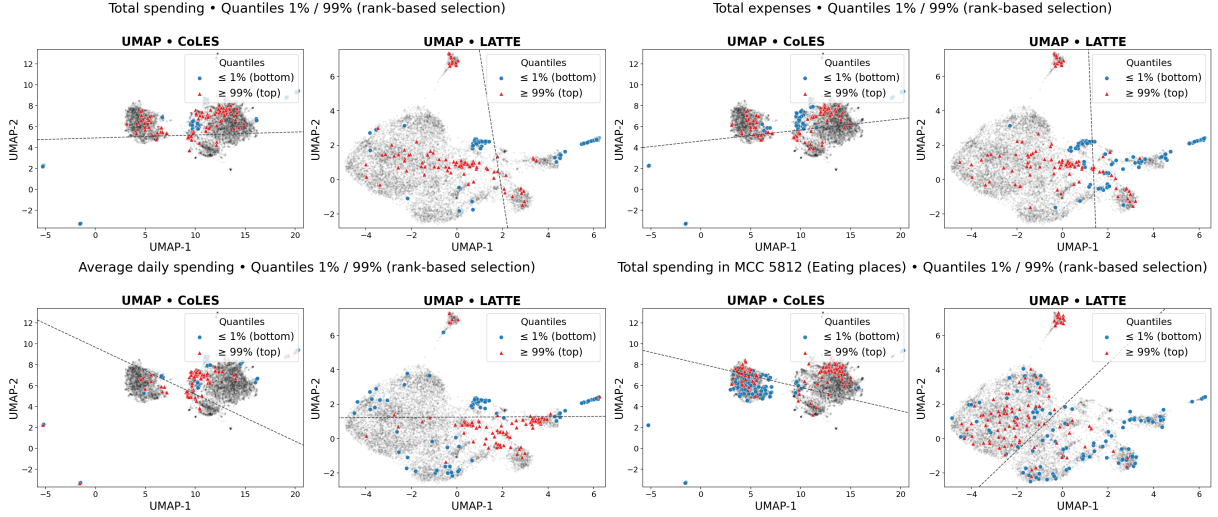


Figure 4: UMAP visualizations of CoLES (left) and LATTE (right) embeddings colored by quantiles of different behavioral statistics. LATTE embeddings exhibit slightly better linear separation across some statistics such as (a) Total spending, (b) Total expense, (c) Average daily spending, and (d) Total spending in MCC (Eating places).

spending, total expenses, average daily spending, and spending within the Eating places MCC category—indicating a stronger alignment between embedding geometry and behavioral variance.

To complement these qualitative observations, we conduct a quantitative assessment of separability for users with extreme behavioral profiles. As reported in Table 4, LATTE consistently achieves higher separability scores than CoLES, particularly for total expenses, average spending, and spending in top MCC categories, while performance remains comparable for total sum and weekday spending.

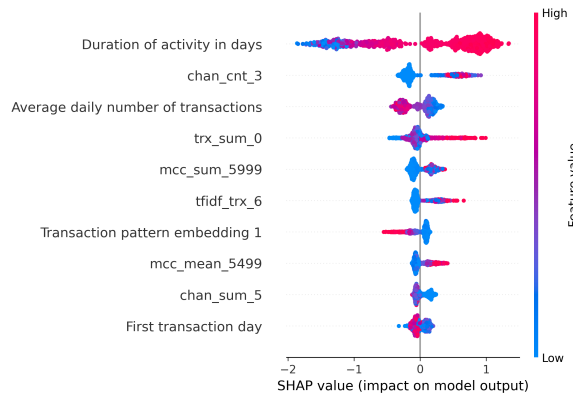


Figure 5: SHAP summary plot showing the most influential features contributing to model predictions.

Finally, the SHAP summary plot (Figure 5) provides further insight into feature importance within the predictive model. The most influential predictors include duration of activity, channel count, and average daily number of transactions, which show strong positive SHAP values for clients with higher

feature magnitudes. Additionally, MCC-specific spending features and transaction-pattern embeddings make substantial contributions, implying that both aggregated behavioral indicators and categorical spending patterns play a crucial role in model predictions.

7 Conclusion

We presented a novel method (Fig. 2) for contrastive representation learning from event sequences that leverages synthetically generated textual descriptions as a complementary modality. By aligning structured transaction data with natural language summaries produced by a frozen instruction-tuned LLM, the proposed approach introduces textual priors into the embedding space without requiring labeled data or LLM fine-tuning. Our **LATTE** achieves state-of-the-art results across three key open-source banking tasks, with relative improvements of 6.1% in gender prediction, around 1.0% in age group classification, and 3.7% in churn prediction compared to the baseline. The proposed **LATTE-S** is resource-efficient (a few million parameters, up to 200 samples/sec speed), which is essential for industrial applications where behavioral logs are abundant but supervision is limited.

A particularly promising future direction is to explore richer forms of text-to-sequence alignment, where natural language summaries are coupled with the underlying event dynamics. This approach could yield to inherently interpretable embeddings.

Limitations

While our approach achieves strong empirical performance, it remains constrained by its reliance on a fixed set of pre-computed statistical features that condition the LLM-generated textual descriptions. This dependency limits adaptability when key behavioral patterns are not adequately captured by the chosen statistics. Furthermore, the method assumes that the generated descriptions faithfully reflect the underlying sequence dynamics, making performance sensitive to prompt design and the generalization capacity of the frozen LLM. Because the text encoder is not updated during training, alignment fidelity may further degrade under distributional shifts. Moreover, although the framework requires neither labels nor fine-tuning, it introduces moderate training overhead compared to lightweight contrastive objectives due to large-scale LLM-based generation.

Finally, in this paper, only financial transactions are used. The proposed sequence-to-text alignment framework can be extended to a wide range of domains that generate structured event logs-data types where LLMs often struggle due to sparsity, heterogeneity, and long temporal dependencies. Examples include healthcare (Wang et al., 2024b), education (Liu et al., 2023), e-commerce (Dai et al., 2023; Liu et al., 2025). In these settings, LATTE could leverage LLM-generated descriptions to inject semantic priors into structural representations.

Acknowledgments

The work of A. Savchenko was supported by a grant, provided by the Ministry of Economic Development of the Russian Federation in accordance with the subsidy agreement (agreement identifier 000000C313925P4G0002) and the agreement with the Ivannikov Institute for System Programming of the Russian Academy of Sciences dated June 20, 2025 No. 139-15-2025-011.

The authors would like to thank Artem Sakhno for his valuable assistance in refactoring the source code and conducting additional experiments during the rebuttal phase.

References

Irina Abdullaeva, Andrei Filatov, Mikhail Orlov, Ivan Karpukhin, Viacheslav Vasilev, Denis Dimitrov, Andrey Kuznetsov, Ivan Kireev, and Andrey Savchenko. 2024. ESQA: Event sequences question answering. *arXiv preprint arXiv:2407.12833*.

Dmitrii Babaev, Nikita Ovsov, Ivan Kireev, Maria Ivanova, Gleb Gusev, Ivan Nazarov, and Alexander Tuzhilin. 2022. Coles: Contrastive learning for event sequences with self-supervision. In *Proceedings of the 2022 International Conference on Management of Data*, pages 1190–1199.

Dmitrii Babaev, Maxim Savchenko, Alexander Tuzhilin, and Dmitrii Umerenkov. 2019. Et-rnn: Applying deep learning to credit loan applications. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2183–2190.

Ayoub Bagheri, Anastasia Giachanou, Pablo Mosteiro, and Suzan Verberne. 2023. Natural language processing and text mining (turning unstructured data into structured). In *Clinical Applications of Artificial Intelligence in Real-World Data*, pages 69–93. Springer.

Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2023. TallRec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 1007–1014.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. LLM2vec: Large language models are secretly powerful text encoders. In *First Conference on Language Modeling*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Shitong Dai, Jiongnan Liu, Zhicheng Dou, Haonan Wang, Lin Liu, Bo Long, and Ji-Rong Wen. 2023. Contrastive learning for user sequence representation in personalized product search. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23)*, pages 380–389. ACM.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Jie Gui, Tuo Chen, Jing Zhang, Qiong Cao, Zhenan Sun, Hao Luo, and Dacheng Tao. 2024. A survey on self-supervised learning: Algorithms, applications, and

- future trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Yi Guo, Shunan Guo, Zhuochen Jin, Smiti Kaul, David Gotz, and Nan Cao. 2020. [Survey on visual analysis of event sequence data](#). *Preprint*, arXiv:2006.14291.
- Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. TabLLM: Few-shot classification of tabular data with large language models. In *International Conference on Artificial Intelligence and Statistics*, pages 5549–5581. PMLR.
- Qian Jiang, Changyou Chen, Han Zhao, Liqun Chen, Qing Ping, Son Dinh Tran, Yi Xu, Belinda Zeng, and Trishul Chilimbi. 2023. Understanding and constructing latent modality structures in multi-modal representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7661–7671.
- Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and 1 others. 2024. Time-LLM: Time series forecasting by reprogramming large language models. In *The Twelfth International Conference on Learning Representations*.
- Ivan Karpukhin and Andrey Savchenko. 2024. DeTPP: Leveraging object detection for robust long-horizon event prediction. *arXiv preprint arXiv:2408.13131*.
- Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.
- Bojan Kolosnjaji, Apostolis Zarras, George Webster, and Claudia Eckert. 2016. Deep learning for classification of malware system call sequences. In *AI 2016: Advances in Artificial Intelligence: 29th Australasian Joint Conference*, pages 137–149.
- Jiongnan Liu, Zhicheng Dou, Jian-Yun Nie, Zhenlin Chen, Guoyu Tang, Sulong Xu, and Ji-Rong Wen. 2025. Enhancing sequential personalized product search with external out-of-sequence knowledge. *ACM Transactions on Information Systems*, 43(4):1–25.
- Zitao Liu, Qionqiong Liu, Teng Guo, Jiahao Chen, Shuyan Huang, Xiangyu Zhao, Jiliang Tang, Weiqi Luo, and Jian Weng. 2023. [XES3G5M: A knowledge tracing benchmark dataset with auxiliary information](#). In *Advances in Neural Information Processing Systems 36: Datasets and Benchmarks Track (NeurIPS 2023)*.
- Simone Luetto, Fabrizio Garuti, Enver Sangineto, Lorenzo Forni, and Rita Cucchiara. 2025. One transformer for all time series: Representing and training with time-dependent heterogeneous tabular data. *Machine Learning*, 114(6):1–27.
- Dzhambulat Mollaev, Alexander Kostin, Maria Postnova, Ivan Karpukhin, Ivan Kireev, Gleb Gusev, and Andrey Savchenko. 2025. [Multimodal banking dataset: Understanding client needs through event sequences](#). *Preprint*, arXiv:2409.17587.
- Niklas Muennighoff, SU Hongjin, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. 2024. Generative representational instruction tuning. In *ICLR 2024 Workshop: How Far Are We From AGI*.
- Niklas Muennighoff, Nouamane Tazi, Loïc Magne, Holger Schwenk, Guillaume Lample, Matthijs Douze, Akiko Aizawa, and Edouard Grave. 2023. MTEB: Massive Text Embedding Benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.
- Dmitry Osin, Igor Udovichenko, Viktor Moskvoretskii, Egor Shvetsov, and Evgeny Burnaev. 2025. [EBES: Easy benchmarking for event sequences](#). *Preprint*, arXiv:2410.03399.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.
- Rosbank. 2021. Churn prediction challenge. <https://boosters.pro/championship/rosbank1/overview>. Accessed: 2025-07-04.
- Yucheng Ruan, Xiang Lan, Jingying Ma, Yizhi Dong, Kai He, and Mengling Feng. 2024. [Language modeling on tabular data: A survey of foundations, techniques and evolution](#). *Preprint*, arXiv:2408.10548.
- Sberbank. 2021a. Python and analyze data: Final project (gender). <https://www.kaggle.com/competitions/python-and-analyze-data-final-project>. Accessed: 2025-07-04.
- Sberbank. 2021b. Sirius age group competition. <https://ods.ai/competitions/sberbank-sirius-lesson>. Accessed: 2025-07-04.
- Oleksandr Shchur, Marin Biloš, and Stephan Günnemann. 2020. Intensity-free learning of temporal point processes. In *International Conference on Learning Representations*.
- Aleksei Shestov, Omar Zoloev, Maksim Makarenko, Mikhail Orlov, Egor Fadeev, Ivan Kireev, and Andrey Savchenko. 2025. LLM4ES: Learning user embeddings from event sequences via large language models. *arXiv preprint arXiv:2508.05688*.

- Xin Shi, Shizhe Xue, Kun Wang, Feng Zhou, Jiawei Zhang, Jingren Zhou, and Hongyu Mei. 2023. Language Models Can Improve Event Prediction by Few-Shot Abductive Reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 29532–29557.
- Xiaohan Shou, Debarun Bhattacharjya, Tong Gao, Devika Subramanian, Oktie Hassanzadeh, and Kristin P. Bennett. 2023. Pairwise Causality Guided Transformers for Event Sequences. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 46520–46533.
- Piotr Skalski, David Sutton, Stuart Burrell, Iker Perez, and Jason Wong. 2023. Towards a foundation purchasing model: Pretrained generative autoregression on transaction sequences. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pages 141–149.
- Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. 2024. Test: Text prototype aligned embedding to activate llm’s ability for time series. In *The Twelfth International Conference on Learning Representations*.
- Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivi re, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timoth e Lacroix, Baptiste Rozi re, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Hugo Touvron and 1 others. 2024. [The LLaMA 3 herd of models](#). *arXiv preprint arXiv:2407.21783*.
- Igor Udovichenko, Egor Shvetsov, Denis Divitsky, Dmitry Osin, Ilya Trofimov, Ivan Sukharev, Anatoliy Glushenko, Dmitry Berestnev, and Evgeny Burnaev. 2024. [Seqnas: Neural architecture search for event sequence classification](#). *IEEE Access*, 12:3898–3909.
- Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024a. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*.
- Yue Wang, Tianfan Fu, Yinlong Xu, Zihan Ma, Hongxia Xu, Bang Du, Yingzhou Lu, Honghao Gao, Jian Wu, and Jintai Chen. 2024b. Twin-gpt: digital twins for clinical trials via large language model. *ACM Transactions on Multimedia Computing, Communications and Applications*.
- Gary M. Weiss and Haym Hirsh. 1998. Learning to predict rare events in event sequences. In *KDD*, pages 359–363.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Anton Yeshchenko and Jan Mendling. 2022. [A survey of approaches for event sequence analysis and visualization using the esevis framework](#). *Preprint*, arXiv:2202.07941.
- Bin Yin, Junjie Xie, Yu Qin, Zixiang Ding, Zhichao Feng, Xiang Li, and Wei Lin. 2023. Heterogeneous knowledge fusion: A novel approach for personalized recommendation via LLM. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 599–601.
- Shuai Yu, Dongjie Guo, Yanjie Fu, and Wen Jin. 2025. EventFormer: A Hierarchical Neural Point Process Framework for Spatio-Temporal Clustering Events Prediction. *Journal of Big Data*, 12(1):162.
- Wei Zhang, Chao Liu, Yuxuan Wang, Tao Li, Junjie Chen, and Weiqiang Wang. 2023. FATA-Trans: Field and Time-Aware Transformer for Sequential Tabular Data. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM)*.
- Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, and 1 others. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.
- Zhi Zheng, Wen-Shuo Chao, Zhaopeng Qiu, Hengshu Zhu, and Hui Xiong. 2024. Harnessing large language models for text-rich sequential recommendation. In *The Web Conference 2024*.

A Experimental Details

Experimental Setup. For all experiments, natural language descriptions were generated using several instruction-tuned large language models of different types and scales, including Gemma-3-27B-Instruct, Gemma-2-27B-Instruct (Team et al., 2025), and Qwen3-Instruct-32B and Qwen3-Instruct-4B (Zhang et al., 2025). To encode the resulting behavioral descriptions into fixed-length vectors, we employed the Qwen3-Embedding-8B (Zhang et al., 2025) model, a multilingual encoder optimized for semantic retrieval. The transaction encoder was instantiated as a GRU-based model trained under the CoLES self-supervised framework, serving as the base sequence encoder across all contrastive alignment heads. Both training and inference for the full pipeline—including LLM prompting, embedding computation, and contrastive alignment—were performed on eight NVIDIA Tesla A100 80 GB GPUs.

Pipeline Details. A lightweight GRU-based transaction encoder was pretrained under the CoLES objective and later tuned via lightweight contrastive alignment heads, while keeping the text encoders frozen. Alignment was performed between sequence embeddings and combined textual–statistical embeddings derived from LLM-generated descriptions enriched with numerical features. For fair comparison across downstream models, feature selection was applied to all representations on the validation set, fixing the embedding dimensionality to 512 or 1024.

B Additional Experiments

Contrastive Head	Method	Churn (AUC)	Age Group (Acc)	Gender (AUC)
LATTE [1]	LLM encoder	0.870 ± 0.005	0.665 ± 0.004	0.895 ± 0.002
LATTE [1]	MeanPool	0.871 ± 0.003	0.663 ± 0.003	0.896 ± 0.001

Table 5: Ablation: Comparing description embeddings obtained from a dedicated sentence encoder (ours) vs. directly from the generator LLM.

Impact of Text Embedding Extraction Strategy. We compare two strategies for obtaining text embeddings from behavioral descriptions. In our default setup, we use a frozen sentence encoder (Qwen3-Embedding-8B) to compute embeddings, separating the generation and encoding processes. As an alternative, we extract the embedding directly from the generator LLM (Gemma-3-27B) by mean-pooling hidden states. Following recent work suggesting that mean pooling over

all token embeddings outperforms using the EOS token (BehnamGhader et al., 2024; Muennighoff et al., 2024), we average hidden states over the final $k = 8$ transformer layers. Both versions provides similar embeddings quality.

Metric	Gemma 3 4B	Qwen 3 32B	Gemma 3 27B
Churn (AUC)	0.872 ± 0.003	0.870 ± 0.005	0.869 ± 0.005
Age Group (Acc)	0.658 ± 0.007	0.665 ± 0.005	0.657 ± 0.004
Gender (AUC)	0.897 ± 0.005	0.895 ± 0.002	0.898 ± 0.003

Table 6: Ablation: Effect of language model choice for generating behavioral descriptions.

Language Model Choice for Description Generation. Table 6 examines how the choice of large language model for generating behavioral descriptions influences downstream performance. Among the compared generators, Qwen 3 32B achieves the highest average accuracy across tasks, leading on age group classification (0.665 Acc) and delivering competitive results on churn and gender prediction. Gemma 3 27B attains the best gender AUC (0.898) and remains comparable in other metrics, confirming that mid-scale models with strong instruction tuning can match much larger ones. In contrast, the compact Gemma 3 4B variant underperforms across all tasks. Overall, the results suggest that richer generative capacity and stronger instruction following in the LLM used for description synthesis translate into more informative and transferable sequence representations

Contrastive Head	Text Encoder	Churn (AUC)	Age Group (Acc)	Gender (AUC)
LATTE [1]	mE5-large-instr	0.869 ± 0.005	0.662 ± 0.008	0.896 ± 0.002
LATTE [1]	Qwen3-Emb-0.6B	0.862 ± 0.005	0.662 ± 0.004	0.899 ± 0.003
LATTE [1]	Qwen3-Emb-8B	0.870 ± 0.005	0.665 ± 0.005	0.895 ± 0.002

Table 7: Ablation: Impact of the text embedding model on downstream task performance.

Text Embedding Model Selection. Table 7 analyzes how the choice of frozen text encoder for obtaining z_i^{text} affects downstream task performance. We compare mE5-large-instruct (Wang et al., 2024a) with two Qwen3 embedding variants (Yang et al., 2025): a compact Qwen3-Emb-0.6B and a larger Qwen3-Emb-8B. The results show that all encoders perform comparably, with differences within a narrow margin of 0.5–1.0 pp across tasks. mE5-large-instruct yields the highest AUC on churn prediction (0.869), while Qwen3-Emb-8B slightly leads in age classification (0.665 Acc). The Qwen3-Emb-0.6B model achieves the best gender AUC (0.899), despite being the smallest.

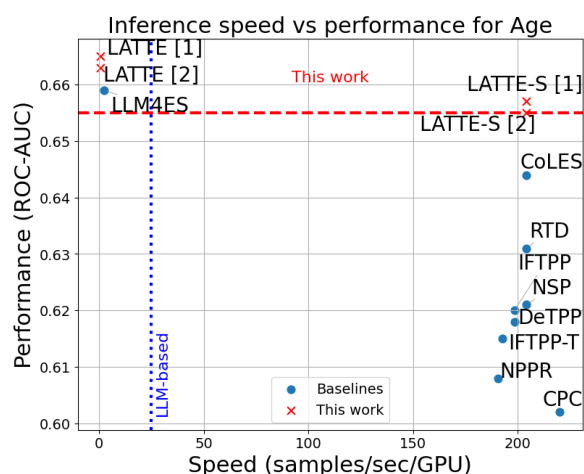


Figure 6: Figure of Merit comparison of model performance in ROC-AUC and inference speed in samples per second per GPU on Age dataset.

Runtime Analysis Figure 6 presents results for the Age prediction task. We observe trends consistent with those in the Churn dataset (Figure 3): generative models attain the highest accuracy (up to 0.665) but exhibit limited inference throughput. In contrast, lightweight LATTE-S variants achieve a favorable trade-off, maintaining competitive performance (0.57 ROC-AUC) while delivering significantly higher inference speeds, exceeding 200 samples per second.

C Text Generation Protocol

We generate natural language descriptions from statistical user summaries by prompting an instruction-tuned LLM. Table 8 illustrates the full prompt used for behavioral description generation, while Table 9 provides an example output produced by the model.

System: *You are an expert in financial transaction analysis. Your task is to generate clear, structured, and concise descriptions of user financial behavior based on given transaction data. Use data-driven insights and avoid speculation.*

Guidelines:

- Do **not** include phrases like “Here’s a financial behavior description for User X”
- Start **directly** with behavioral insights
- Interpret numbers into **patterns** (e.g., burst spending, routine payments)
- Highlight **spending habits, risk factors, financial consistency**
- Avoid unsupported assumptions; maintain clarity and conciseness

User: Below is a summary of a user’s transaction history:

- Number of transactions: 226
- Active transaction period: 444 days
- Avg transactions per day: 0.51
- Avg transaction interval: 1.97 days
- Top MCCs: Supermarkets, cash withdrawals, telecommunications
- Share of transactions in Top 1 MCC: 0.04
- Avg amount for Top 1 MCC: 2770.47 RUB
- Avg amount for Top 2 MCC: 45969.41 RUB
- Avg amount for Top 3 MCC: 8139.52 RUB
- Share of days with transactions: 0.36
- Total income: 2,706,553.10 RUB
- Total expenses: 3,956,007.97 RUB
- Avg outgoing amount: 17,981.85 RUB
- Avg incoming amount: 451,092.18 RUB

Instructions: - Analyze behavioral patterns

- Identify transaction regularity and category reliance
 - Assess potential risk factors and financial planning traits
 - Write in a structured and engaging way while staying factual
-

Table 8: Prompt template for behavioral description generation

Output:

This user demonstrates a **low-frequency, high-variance spending pattern** over a substantial period (444 days). Transactions occur on approximately 36% of days, averaging one every two days, indicating infrequent but potentially substantial financial activity.

Spending is heavily weighted towards **cash withdrawals from financial institutions** (4% of transactions, average 45969.41 RUB), suggesting a preference for cash-based expenses or large withdrawals for specific purposes. Significant spending also occurs at **grocery stores/supermarkets** (average 8139.52 RUB), representing a consistent, though smaller, portion of overall expenditure. A smaller number of transactions are related to **phone services** (average 2770.47 RUB).

Expenses significantly exceed income (3,956,007.97 RUB vs. 2,706,553.10 RUB), implying reliance on savings, credit, or other income sources not captured in this data. The high average outgoing transaction amount (17,981.85 RUB) coupled with a large average incoming amount (451,092.18 RUB) suggests **irregular but substantial income events** are occurring, potentially offsetting consistent overspending. The combination of infrequent transactions, high cash withdrawals, and a substantial expense-to-income imbalance points to **potentially inconsistent financial management**. While not indicative of reckless spending, the reliance on infrequent income and high cash usage warrants further investigation to determine financial stability and potential risk factors. There’s a lack of diversification in spending, with the top three MCCs dominating activity.

Table 9: Example output generated from the LLM