

Peer Review

Review of: "SampleLLM: Optimizing Tabular Data Synthesis in Recommendations"

Evgenii Arsentev¹

1. Independent researcher

1. What the work claims

The authors propose SampleLLM, a two-stage pipeline for generating synthetic tabular data in recommendation settings. In the first stage, a quantised Llama-3-70B (§3.1.3, "INT4 quantized version of Llama-3-70b-instruct-awq") receives a hand-picked instruction, refined iteratively with dataset documentation (Eq. 4, at most 5 iterations), plus a exemplars chosen one per K-means cluster (Eq. 5), and writes b rows per call, repeated Q times. In the second stage, the raw synthetic set is reweighted by importance sampling. The weight of a row is the ratio of its probability under the original data to its probability under the synthetic data (Eq. 14), where the joint probability is factorised under a "semi-independence" assumption: features are grouped by thresholding a second-order, Hessian-based interaction map computed on a trained predictor (Eq. 10–12), group probabilities are frequency counts, and the label term comes from a discriminative model trained on each dataset (Eq. 13).

The evidence is offline and online. Offline, on three CTR datasets (ML-1M, Amazon, Douban) and two general ones (HELOC, Covertypes), SampleLLM is compared with seven generators, from CTGAN to REaLTabFormer, in two protocols: adding 10% synthetic rows to the real training set (augmentation utility, Table 2) and training only on synthetic rows (MLE utility, Table 3). An ablation on Douban removes the LLM, the cluster sampling, or the importance sampling (Figure 4, Table 4). Online, §4 reports a one-week deployment on the Huawei app advertising platform for cold-start apps, with +2% RPM and +1.86% ECPM.

2. Whether the conclusion holds on the data

I liked several things here, and they are not small. The second stage is the real contribution: instead of trusting the LLM to reproduce the distribution, the authors correct it afterwards with an explicit density ratio, and they make that ratio tractable by letting the data decide which features must be modelled jointly (Eq. 12, Table 5). That is a sensible, transferable idea. The baseline set is broad and includes both LLM-based generators (GReaT, REaLTabFormer). Both utility protocols are reported, every experiment is averaged over 10 runs (§3.1.3), and the paper goes beyond benchmarks with an industrial test. The ablation in Table 4 points the same way as the main tables: SDV similarity drops from 94.83% (SampleLLM) to 92.32% (-LLM), 94.13% (-CS), and 93.26% (-IS).

Where I think the claims run ahead of the numbers is in their size. In the augmentation protocol, the gain over training on real data alone is very small on the three recommendation datasets. I computed it from Table 2, AUC columns, SampleLLM row minus Original row: ML-1M $0.8180 - 0.8173 = 0.0007$; Amazon $0.7082 - 0.7075 = 0.0007$; Douban $0.8027 - 0.8016 = 0.0011$. Among the AUC datasets, only HELOC moves visibly ($0.7732 - 0.7638 = 0.0094$), and HELOC is a general, not a recommendation, dataset; Covertypes, the other general one, gains $0.7364 - 0.7250 = 0.0114$ in F1. The asterisks in Table 2 mark significance against the best baseline, not against Original, and the paper reports no standard deviations, so I cannot tell whether a 0.0007 AUC lift over Original is distinguishable from run-to-run noise. Many baselines in Table 2 sit below Original on ML-1M and Amazon, which is honest and informative; it would help to say plainly that on these datasets, the main value of SampleLLM is that it does not hurt and helps slightly.

The MLE protocol tells a similar story with a different emphasis. On ML-1M, Table 3, AUC column, every method trained on synthetic data lands between 0.4944 (PATE-GAN) and 0.5668 (SampleLLM), against 0.8173 for Original. SampleLLM is the best of the group, but 0.5668 is close to chance for a CTR task. The text in §3.2.2 acknowledges the degradation; I would carry that sentence into the abstract because "significantly surpasses existing methods" reads very differently to a practitioner who sees 0.5668.

Two points do not sit well with the tables, and both are easy to address.

- §3.2.2 says SampleLLM "outperforms other baselines across all datasets on both augmentation and MLE utility measures". In Table 2, Covertypes, Recall column, GReaT has 0.7440 and SampleLLM 0.7437 (no asterisk). So the claim holds in every cell but one.
- In Table 3, HELOC, Logloss column, Original is 0.6445, while SampleLLM reaches 0.5799 and even TVAE reaches 0.5856, after training on roughly 10% of 8,367 rows (Table 1). At the same time, HELOC

AUC for Original (0.7638) is higher than for TVAE (0.7627). A model trained on about 837 synthetic rows having much lower log loss than one trained on the full real set, with similar AUC, is surprising. In Table 2, all HELOC log losses sit between 0.6403 and 0.6461. §3.2.2 explains SampleLLM beating Original on HELOC by the dataset's data scarcity, but the same low log loss appears for TVAE, TabDDPM, GReaT, and REaLTabFormer, so I suspect a calibration issue or a copied cell in the Original row of Table 3; I could not resolve it from the text.

On statistics: the "two-sided t-test with $p < 0.05$ " over 10 runs is a reasonable choice, but it matters what varies between runs. If only the DNN seed changes while the synthetic set is generated once, the test measures training noise and says little about the generator. If the LLM generation and the importance sampling are also redrawn, the test is far stronger. I could not find which is the case.

The ablation is limited to one dataset (Douban), and as rendered on Qeios Figure 4 is an image whose bar values I cannot read, so I can verify only the SDV part (Table 4). The -LLM variant replaces the LLM with TabDDPM and keeps importance sampling; that is useful. The complementary test is missing: importance sampling applied on top of GReaT or REaLTabFormer outputs. Without it, a reader cannot separate "the LLM generation with cluster exemplars is better" from "the reweighting step would help any generator".

The online test is a welcome addition, but it is described too thinly to carry weight. It ran for one week (§4), with no stated traffic split, no confidence interval for the +2% RPM and +1.86% ECPM, a different LLM (Qwen2.5 rather than Llama-3), labels predicted by the LLM rather than sampled from logs, and feature groups obtained by "Hessian-based methods on a previously trained recommendation model" rather than by the exact Eq. 10–12 procedure. I read it as encouraging evidence of practical relevance, not as a validation of the offline method as specified.

One more question the paper raises but does not answer. §2.1 motivates synthesis partly by privacy ("without exposing sensitive information"), and ML-1M, Amazon, and Covertypes are public datasets that a 70B model may well have seen during pretraining. A distance-to-closest-record check against the training split would show whether the LLM copies rows. This bears on both the privacy motivation and the interpretation of the gains.

3. Reproducibility

The data side is in good shape. All five datasets are public and linked in the footnotes, the 8:1:1 split is stated, and Table 1 gives the exact counts (I checked ML-1M: $800,167 / (800,167 + 100,020 + 100,022) =$

800,167 / 1,000,209 = 80.0%). The predictive model and its grid (1–4 layers, 16–128 neurons) are given in §3.1.3, and the feature groups actually used are listed in Table 5.

The generation side is much harder to reproduce. I could not find the following.

- **LLM snapshot and decoding.** "Llama-3-70b-instruct-awq" does not name a specific checkpoint or repository revision. Temperature, top-p, maximum new tokens, and any stop sequences are not given. The same holds for Qwen2.5 in §4 (size and version are not stated).
- **Prompts.** Figure 3 is "a simplified instruction example". The five candidate instructions per dataset (§2.3), the CoT refinement prompt behind Eq. 4, and the final instructions are not published.
- **Hyper-parameters of the method itself.** The values of a , b , Q (§2.2, §2.4), and the threshold γ (Eq. 12) are never reported. Nor is the size of the raw set D_s relative to the final 10%, or whether importance sampling draws with or without replacement.
- **Instruction selection.** §2.3 picks the instruction "that delivers the best performance" with 1% synthetic data, but does not say on which split. If it was the test split, that is a mild leak; if validation, saying so closes the question.
- **Code and compute.** I found no code link, no seeds, and no generation cost (GPU type, hours, number of LLM calls). For Douban, 10% of the training set is about 134,840 rows (10% of 1,348,399, Table 1), which is a nontrivial generation budget.
- **Stability of groups.** Appendix B notes that interaction maps and feature groups "may exhibit slight variations" across trainings of the predictor. It would help to report how often the groups in Table 5 change across the 10 runs.
- **Algorithm 1.** Appendix A refers to lines 1–12 of Algorithm 1, but as rendered on Qeios, the pseudo-code itself is not visible; only the surrounding text is.

4. What specifically to fix

1. **Abstract.** Replace "significantly surpasses existing methods" with the actual size of the effect: AUC gains over Original of 0.0007 (ML-1M), 0.0007 (Amazon), and 0.0011 (Douban) in Table 2, and the near-chance MLE AUC on ML-1M (Table 3, 0.5668).
2. Tables 2 and 3: add standard deviations over the 10 runs for every cell, and add a significance test of SampleLLM against Original, not only against the best baseline.
3. §3.1.3: say what is resampled across the 10 runs (DNN seed only, or also LLM generation and importance sampling).

4. §3.2.2, last paragraph. Soften "across all datasets on both augmentation and MLE," since Table 2, Coverttype, Recall shows GReaT 0.7440 above SampleLLM 0.7437.
5. Table 3, HELOC, Original row: please recheck the Logloss of 0.6445; it is higher than five of the eight synthetic-only models in the same column (for example, SampleLLM 0.5799, TVAE 0.5856), which is hard to square with the AUC column.
6. §2.2, §2.4, and Eq. 12: report a , b , Q , γ , the size of D_s , and the sampling scheme of the second stage. A short sensitivity plot for γ on one dataset would show how much the groups in Table 5 matter.
7. §3.3. I would add "GReaT + IS" and "REaLTabFormer + IS" rows, extend the ablation to at least one more dataset (HELOC is cheap), and give Figure 4's values as a table so they survive text rendering.
8. §3.1.3 and §4: name the exact Llama-3 and Qwen2.5 checkpoints, decoding parameters, GPU, and generation time; publish the full prompts of Figure 3 and Eq. 4 in an appendix, together with code.
9. Eq. 9–10: the notation drifts. Eq. 9 takes the expectation over $x' \sim D_b$ but uses x_b inside; Eq. 10 takes it over $x \sim D_b$ and evaluates the Hessian at x_b . Please state once which variable is the sample and which is the baseline, and use ∂ consistently (Eq. 9 as rendered shows δ).
10. Eq. 14 as rendered on Qeios places " $(d \in \{o, s'\})$ " between two factors of the product; moving it to the end of the line would make the formula readable.
11. §4. Report the traffic split, the number of cold-start apps and impressions, and an interval for RPM and ECPM. State that the online variant differs from the offline method (Qwen2.5, LLM-predicted labels, Hessian-based groups) so readers do not over-read it.
12. Privacy and memorisation (§2.1 motivation): a distance-to-closest-record comparison between synthetic rows and the training split, at least for HELOC and ML-1M.
13. Small items. §2.1 says "data argumentation," which should be "data augmentation." Janizek et al. (2021) and the Synthcity arXiv paper (Qian et al., 2023) appear twice each in the reference list, and the Borisov et al. entry contains a raw LaTeX escape (" $\text{\ssler}$ "). Table 4 and Table 6 carry identical captions; adding the dataset name (Douban, HELOC) would remove the ambiguity.

5. Recommendation

Minor revisions. The method is coherent, the second-stage reweighting is a genuinely useful idea, and the comparison against seven generators on five datasets plus an industrial test is more than most papers in this area provide. The fixes I ask for are mostly about stating effect sizes honestly, adding variance and the missing generation details, fixing one text-table mismatch, and rechecking one table

cell. None of them requires redesigning the study, though the extra ablation rows and the HELOC log-loss check may take some work.

Declarations

Potential competing interests: No potential competing interests to declare.