Instruction Tuning of Large Language Models for Tabular Data Generation—in One Day

Milad Abdollahzadeh [1 2] **Abdul Raheem** [2] **Zilong Zhao** [3 2] **Uzair Javaid** [2] **Kevin Yee** [2]
Nalam Venkata Abhishek [1] **Tram Truong-Huu** [1] **Biplab Sikdar** [3]

Abstract

Tabular instruction tuning has emerged as a promising research direction for improving LLMs' understanding of tabular data. However, the majority of existing works only consider question-answering and reasoning tasks over tabular data, leaving tabular data generation largely unnoticed. In this work, for the first time, we explore the efficacy of instruction tuning in improving LLMs' tabular data generation capabilities. More specifically, given the high data and computation requirements of tabular instruction tuning, **we aim to address the possibility of instruction tuning for tabular data generation with limited data and computational resources**. To achieve this, we first create a high-quality instruction dataset for tabular data, enabling efficient LLM comprehension. We then instruction-tune an open-source LLM (Llama3.1-8B-Instruct) on the training set of this dataset to improve its tabular data generation performance. Our experimental results show that by using our high-quality dataset and instruction-tuning on only 7K instructions with an A100 GPU, for less than 6 hours, **we achieve tabular data generation performance on par with the most capable commercial LLM, GPT-4o**.

1. Introduction

Large Language Models (LLMs), trained on web-scale corpora, have demonstrated impressive performance across a wide range of natural language processing (NLP) tasks (Vaswani et al., 2017; Radford et al., 2018; Wang et al., 2018; Hendrycks et al., 2021), and also surprisingly strong performance in following instructions (Wei et al., 2022a; Ouyang et al., 2022) and reasoning over textual data (Wei et al., 2022c; Huang & Chang, 2023). These models are widely regarded as emergent repositories of world knowledge (Wei et al., 2022b; Schaeffer et al., 2023; Roberts et al., 2020). However, as their pretraining objectives are inherently optimized for the text modality, which may have some tabular data in the training data, their performance on table-based tasks remains suboptimal (Yang et al., 2024; Lin et al., 2025). Recent studies suggest that this limitation stems from the structural mismatch between tabular and textual data: tabular data exhibits a bi-dimensional and relational structure, whereas LLMs are trained using a uni-dimensional, autoregressive (or masked language modeling) objective, leading to misalignment in inductive biases and representational capacities (Liu et al., 2024; Su et al., 2024).

Tabular Instruction Tuning has recently emerged as a promising research direction, drawing inspiration from the success of instruction tuning in enhancing the capability of LLMs to handle novel tasks (Ouyang et al., 2022; Zhang et al., 2023). Specifically, recent works (Zhang et al., 2024c;b; Deng & Mihalcea, 2025) have proposed generating natural language instructions based on tabular data and using these for instruction-tuning LLMs on table-related tasks. Studies show that this approach leads to notable improvements in LLMs' understanding of tabular structures and their performance on tasks involving structured data, such as table-based reasoning and question answering (Deng et al., 2022; Cheng et al., 2021; Chen et al., 2019).

Research Gap. Although several works have explored instruction tuning over tabular data, they primarily focus on question answering (QA) and reasoning tasks (Parikh et al., 2020; Aly et al., 2021; Zhong et al., 2017; Chen et al., 2020). ***The task of generating tabular data, however, remains largely unaddressed.*** Beyond understanding tabular data, which has been the main focus of prior research, the ability to generate realistic and domain-relevant tabular data is increasingly important, especially given the widespread presence of such data in the scientific community and its critical role across various real-world applications (Van Breugel & Van Der Schaar, 2024; Hollmann et al., 2023; 2025). En-

[1]Singapore Institute of Technology (SIT), Singapore [2]Betterdata AI, Singapore [3]National University of Singapore (NUS), Singapore. Correspondence to: Milad Abdollahzadeh <milad@betterdata.ai>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

abling LLMs to generate synthetic tabular data can help augment limited real-world datasets and accelerate the adoption of machine learning techniques in data-scarce domains. This work aims to fill this gap by investigating the effectiveness of instruction tuning for enhancing the tabular data generation capabilities of LLMs.

Limitations. The main limitation in exploring the efficacy of instruction tuning for tabular data generation is the ***high requirements for large-scale data and extensive computational resources***. For example, the recent state-of-the-art model TableLlama uses around ***2 million tabular instructions*** and ***48 A100 GPUs*** to instruction-tune the base LLM and improve its performance on table-based question answering and reasoning tasks.

In this paper, we aim to answer the following question: ***Can we improve the tabular data generation capabilities of LLMs by instruction tuning these models on limited data and with a limited amount of compute?***
To answer this question, we first create a high-quality instruction dataset for conditional tabular data generation, including 10K instructions. This dataset is gathered from various domains, and extensive metadata is included, together with a snapshot of the input table, to help the LLM follow the context better. We then fine-tune an open-source LLM on this instruction dataset using a single A100 GPU (for less than 6 hours). We show that this instruction-tuning on a limited but high-quality dataset can significantly increase the base LLM's capability in tabular data generation with competitive results compared to the most capable commercial LLM, GPT-4o. Our main contributions are:

- To the best of our knowledge, for the first time in the literature, we explore the efficacy of instruction tuning on improving the performance of the LLMs for tabular data generation.
- We create a high-quality instruction dataset for the tabular data generation task to steer the LLM to more precise tabular data generation by including the general and column-wise description of the table as metadata.
- Experimental results show that instruction tuning with limited resources and on this limited but high-quality instruction dataset can considerably improve the performance of the base LLM on tabular data generation, and deliver a performance on par with powerful models like GPT-4o.

2. Related Work

Tabular Instruction Tuning. TableLLM (Zhang et al., 2024c) performs tabular instruction tuning on LLMs to enable handling various operations on tabular data with LLMs like QA, and Pandas code generation for visualization and analysis purposes. TableLlama (Zhang et al., 2024b) creates a large instruction dataset for table-based QA and reasoning tasks and instruction-tunes LLM on this dataset. TAMA (Deng & Mihalcea, 2025) analyzes the impact of hyperparameter selection on efficient tabular instruction tuning. However, none of these works addresses the tabular data generation task with instruction tuning.

Tabular Data Generation. Before the emergence of LLMs, generative models like GANs (Zhao et al., 2021; 2024), VAEs (Wang & Nguyen, 2025), and Diffusion Models (Shi et al., 2025) were the primary methods for generating tabular data. Recently, leveraging LLMs' strong text generation capabilities, multiple works have focused on converting tabular data into text and then fine-tuning LLMs for tabular data generation (Borisov et al., 2022; Zhao et al., 2023; Wang et al., 2024). However, these models often struggle to follow table-based instructions (Zhang et al., 2024b;c).

3. Problem Setup

Let $\mathcal{T}$ denote a table with $\mathcal{R}$ rows and $\mathcal{C}$ columns, and $\mathcal{M}$ represent its associated metadata (e.g., table title, description). The objective of the instruction following for the tabular data generation with an LLM f_θ is to generate a new table $\mathcal{T}'$. This generation is conditioned on the input table $\mathcal{T}$, its metadata $\mathcal{M}$, and an instruction $\mathcal{I}$ describing the desired generation task for $\mathcal{T}'$:

$$f_\theta(\mathcal{I}, \mathcal{T}, \mathcal{M}) \rightarrow \mathcal{T}' \tag{1}$$

Ideally, $\mathcal{T}'$ should follow the distribution of $\mathcal{T}$. This means $\mathcal{T}'$ should possess the same column structure (features) as $\mathcal{T}$ and preserve both the intra-column distributions for each column and the inter-column relationships observed in $\mathcal{T}$.

4. Proposed Method

In this section, we propose our **I**nstruction **T**uning for **T**abular data **Gen**eration (**ITT-GEN**). To improve tabular data generation with LLMs, we perform two main steps: first, we create an instruction dataset for tabular data generation; and next, we fine-tune an open-source LLM on these instructions. In what follows, we discuss the details.

4.1. Creating Instruction Dataset for Tabular Data Generation

Data Collection. We sample 20 publicly available tabular datasets that cover 10 different topics. We separate them and select 14 tables for training and in-domain evaluation, and the remaining 6 tables as held-out unseen datasets for out-of-domain (OoD) evaluation. The list of these datasets with their topics is shown in Supp., Section A.1 (Table 3).

Creating Instruction Dataset. For each dataset, in our training set, we construct 500 training instances and 100

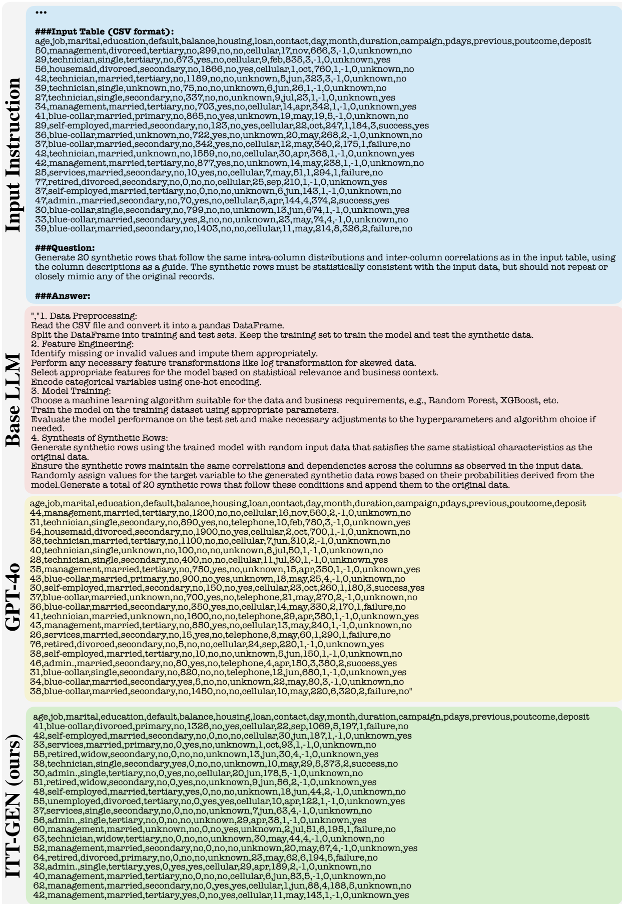

Figure 1. Example of output response of different LLMs for our instruction for tabular data generation. Base LLM generated some unrelated instructions. However, GPT-4o and our proposed ITT-GEN produce 20 rows of tabular data that follow the same structure, and also the distribution of the input table. Only a part of the input instruction is included due to space limitations. Better viewed when zoomed in.

evaluation instances. For evaluation datasets, we only construct 100 evaluation instances. Each instance in our instruction dataset includes an instruction $\mathcal{I}$ which describes the generation task, an input table $\mathcal{T}$ and its metadata $\mathcal{M}$, and the expected output table $\mathcal{T}'$. The details of constructing each part are as follows:

- We manually design the instruction $\mathcal{I}$ to describe the tabular data generation task.
- The metadata $\mathcal{M}$ of each table consists of a general description of the table (topic, the general structure, and the applications), and a column-wise detailed description that includes column name, the data types (numerical, categorical, or textual) for each column. We obtain metadata of each table by passing the whole table into GPT-4o (Hurst et al., 2024) and prompting it to generate this information. We manually go through all generated descriptions to ensure their quality and correctness (More details in Supp., Section A.3).
- For input and output tables, we randomly select N rows ($N = 20$ in our experiments) of the corresponding table. Our empirical results show that using a set of rows as (expected) output during instruction tuning leads to better results compared to the next token prediction used in previous works (Wang et al., 2024).

An example of the created instruction is shown in Supp., Section A.2 (Figure 2).

4.2. Instruction-tuning LLM

After creating the instruction dataset for tabular data generation, we fine-tune an LLM on the training set of this dataset to improve its tabular generation capabilities. We use Llama3.1-8B-Instruct as our base model. This is a compact model from Llama3 herd of models (Grattafiori et al., 2024), where a post fine-tuning (Rafailov et al., 2023) is performed on Llama3.1-8B to enhance its textual instruction following behavior.

Note that our approach is agnostic to the choice of base LLM. In Supp., Section B.1, we provide additional experimental results to show that our approach also improves tabular data generation performance of TableLlama (Zhang et al., 2024b) (SOTA open-source model for table understanding tasks) as base LLM.

5. Experiments

5.1. Experimental Setup

Details of Training and Inference. As mentioned in Section 4.2, in our experiments, we used Llama3.1-8B-Instruct (Grattafiori et al., 2024) as our base model. We fine-tuned Llama3.1-8B-Instruct on our proposed instruction dataset for tabular data generation with the Huggingface transformers library (Wolf et al., 2020). Considering that we have 500 instructions for each of the 14 datasets used for training, we mixed all these 7000 instructions and randomly shuffle them. We used a learning rate of 2e-5 with a batch size of 3. We trained our model on an A100 80GB GPU for 2 epochs. We employed DeepSeed training with ZeRO-2 stage (Rajbhandari et al., 2020) for more efficient training.

Models for Comparison. To the best of our knowledge, there are no similar works in the literature that perform instruction tuning for tabular data generation. Therefore, we compare our proposed model with two models: i) Llama3.1-8B-Instruct (Grattafiori et al., 2024) as the base LLM used in this study, and ii) GPT-4o (Hurst et al., 2024), which is one of the most capable commercial LLMs at the time of writing this paper (Shahriar et al., 2024).

Evaluation Metrics. We follow the existing tabular data generation works (Zhao et al., 2021; 2024; Li et al., 2025; Shi et al., 2025) and evaluate our approach using fidelity and utility metrics. The details are as follows:

- **Fidelity** measures the distributional similarity between generated and tabular data. Two well-known metrics for measuring fidelity of the generated data are: *i) Shape,* which measures the similarity between the marginal distribution of the real and generated data for each column (Zhang et al., 2024a), and *ii) Trend* which measures the capability of the generated data to capture the correlation between different columns (Shi et al., 2025). Higher values of *Shape* and *Trend* metrics indicate a higher data fidelity.
- **Utility** evaluates whether generated tabular data is useful for a downstream task. To evaluate the utility, we use Train-on-Synthetic, Test-on-Real (*TSTR*) framework (Xu et al., 2019), which trains the model on generated (synthetic) tabular data, and then performs the evaluation on held-out real tabular data. For this framework, we use three different models (for training and evaluation), including linear, random forest (Breiman, 2001), and XGBoost (XGB) (Chen & Guestrin, 2016).

5.2. Experimental Results

An example of generated output for our input instruction is shown in Figure 1. As one can see, our proposed ITT-GEN and GPT-4o are able to generate tabular data that follows the same distribution as the input table. However, base LLM (Llama3.1-8B-Instruct) fails to follow our instructions to generate tabular data and starts to generate some irrelevant instructions. We remark that a similar behavior happens for most of the instructions, and only for some instructions, the base LLM is able to generate limited rows (not the whole 20 rows asked) of tabular data. Nevertheless, we collect all generated tabular data and filter out the irrelevant parts to be able to report fidelity and utility metrics for the base LLM.

Fidelity Results. Table 1 shows the fidelity results for generated tabular data with different algorithms. As one can see, the proposed ITT-GEN approach has on-par performance with the powerful GPT-4o model. Note that for base LLM, even though the metrics show competitive performance, these are calculated only for the portion of the output that is tabular data (20%), and the remaining unrelvenet generated data (80% of generated output with base LLM) is discarded for the sake of only being able to report these metrics.

Utility Results. Table 2 shows the utility results for different approaches. Similarly, the proposed ITT-GEN yields a performance on par with GPT-4o indicating that generated tabular data with our instruction-tuned LLM can be efficiently used for downstream tabular tasks.

Table 1. Fidelity results across different algorithms.

Dataset	Base LLM		ITT-GEN (OURS)		GPT-4o	
	Shape	Trends	Shape	Trends	Shape	Trends
adult	87.48	75.13	85.73	52.54	92.34	87.96
bank	75.63	65.08	85.57	86.34	93.42	91.7
bestseller	89.12	90.5	89.56	93.16	-	86.4
biodeg	89.59	80.04	91.68	86.61	94.12	86.54
boston	88.91	87.47	92.38	88.98	90.87	93.02
breast_cancer	55.31	37.07	84.12	69.36	78.65	64.16
BTC-USD_stock	90.19	95.06	88.2	99.31	93.52	98
california_housing	88.7	90.52	73.29	80.06	96.27	97.84
car_prediction_data	74.17	54.44	84.59	60.77	78.8	61.97
credit-g	88.3	78.38	86.29	75.05	93.12	86.67
diabetes	89.45	91.02	83.41	88.77	89.93	88.11
healthcare_insurance	88.52	74.35	91.76	86.74	93.14	88.39
iris	82.69	55.39	88.17	77.86	89.58	87.13
job_posting	40.52	22.4	54.55	36.15	64.56	41.01
Players2024	34.84	11.48	53.55	16.69	53.09	16.13
room_occupancy	81.56	74.99	86.89	81.56	88.42	91.11
supermarket_store_branches	90.36	97.88	83.2	90.45	93.85	96.46
tour_travels_customer_churn	84.33	70.19	91.59	75.18	90.69	75.86
twitter_astrazeneca_anti_covid	84.7	96.47	91.83	98.65	75.67	98.03
wdbc	85.89	88.26	87.43	92.62	90.21	96.08

Table 2. Utility result for synthetic data. Averaged AUC and R2 scores are reported for classification and regression datasets, respectively. '–' indicate the output can not be used to train an ML model. Note that we only report a subset of datasets here. Others follow the same trend.

Dataset	Real	BaseLLM	ITT-GEN	GPT-4o
adult	0.8796	0.655867	0.826533	0.873200
bank	0.800720	0.353441	0.616246	0.819928
bestseller	0.781972	–	0.743701	0.710766
biodeg	0.917188	0.816096	0.862471	0.922341
boston	0.745258	0.677436	0.655484	0.729943
berast_cancer	0.9942	–	0.9831	0.9919
BTC-USD-stock	0.995497	0.917406	0.993921	0.990918
California housing	0.640855	0.393930	0.497865	0.589859
Diabetes	0.82038	0.821207	0.798160	0.797334
Healthcare insurance	0.737844	0.360006	0.695602	0.716192
Iris	1.0000	–	0.987143	0.997149
Players 2024	0.380000	0.327586	0.425532	0.464481
Room Occupancy	0.993658	0.976697	0.993144	0.994749
Tour & Travels Cusomer Chorn	0.767578	0.685234	0.543672	0.706484
Twitter Atrazenca Anti Covid	0.9457	0.89584	0.93094	0.93573
Wdbc	0.99235	0.982966	0.979396	0.988066

6. Conclusion

In this paper, for the first time in the literature, we explore the potential of leveraging instruction tuning to improve tabular data generation performance. For this, we create an instruction dataset for the tabular data generation task, and instruction-tune an open-source base LLM on this dataset. Our results suggest that instruction-tuning on our small but high-quality dataset with only one A100 GPU and for less then 6 hours, can yield a performance on par with GPT-4o, the most capable commercial LLM.

References

Aly, R., Guo, Z., Schlichtkrull, M., Thorne, J., Vlachos, A., Christodoulopoulos, C., Cocarascu, O., and Mittal, A. Feverous: Fact extraction and verification over unstructured and structured information. *arXiv preprint arXiv:2106.05707*, 2021.

Borisov, V., Seßler, K., Leemann, T., Pawelczyk, M., and Kasneci, G. Language models are realistic tabular data generators. *arXiv preprint arXiv:2210.06280*, 2022.

Breiman, L. Random forests. *Machine learning*, 45:5–32, 2001.

Chen, T. and Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Chen, W., Wang, H., Chen, J., Zhang, Y., Wang, H., Li, S., Zhou, X., and Wang, W. Y. Tabfact: A large-scale dataset for table-based fact verification. *arXiv preprint arXiv:1909.02164*, 2019.

Chen, W., Zha, H., Chen, Z., Xiong, W., Wang, H., and Wang, W. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. *arXiv preprint arXiv:2004.07347*, 2020.

Chen, Y., Qian, S., Tang, H., Lai, X., Liu, Z., Han, S., and Jia, J. LongloRA: Efficient fine-tuning of long-context large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Cheng, Z., Dong, H., Wang, Z., Jia, R., Guo, J., Gao, Y., Han, S., Lou, J.-G., and Zhang, D. Hitab: A hierarchical table dataset for question answering and natural language generation. *arXiv preprint arXiv:2108.06712*, 2021.

Deng, N. and Mihalcea, R. Rethinking table instruction tuning. *arXiv preprint arXiv:2501.14693*, 2025.

Deng, X., Sun, H., Lees, A., Wu, Y., and Yu, C. Turl: Table understanding through representation learning. *ACM SIGMOD Record*, 51(1):33–40, 2022.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding, 2021. URL `https://arxiv.org/abs/2009.03300`.

Hollmann, N., Müller, S., Eggensperger, K., and Hutter, F. Tabpfn: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations*, 2023.

Hollmann, N., Müller, S., Purucker, L., Krishnakumar, A., Körfer, M., Hoo, S. B., Schirrmeister, R. T., and Hutter, F. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Huang, J. and Chang, K. C.-C. Towards reasoning in large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1049–1065, 2023.

Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Li, J., Zhao, B., Zhao, Z., Yee, K., Javaid, U., Lao, Y., and Sikdar, B. Tabtreeformer: Tree augmented tabular data generation using transformers. *arXiv preprint arXiv:2501.01216*, 2025.

Lin, X., Xu, C., Yang, M., and Cheng, G. Ctsyn: A foundational model for cross tabular data generation. In *International Conference on Learning Representations*, 2025.

Liu, T., Wang, F., and Chen, M. Rethinking tabular data understanding with large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 450–482, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.26. URL `https://aclanthology.org/2024.naacl-long.26/`.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Parikh, A. P., Wang, X., Gehrmann, S., Faruqui, M., Dhingra, B., Yang, D., and Das, D. Totto: A controlled table-to-text generation dataset. *arXiv preprint arXiv:2004.14373*, 2020.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. Improving language understanding by generative pre-training. 2018.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36: 53728–53741, 2023.

Rajbhandari, S., Rasley, J., Ruwase, O., and He, Y. Zero: memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking,* *Storage and Analysis*, SC '20. IEEE Press, 2020. ISBN 9781728199986.

Roberts, A., Raffel, C., and Shazeer, N. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5418–5426, 2020.

Schaeffer, R., Miranda, B., and Koyejo, S. Are emergent abilities of large language models a mirage? *Advances in Neural Information Processing Systems*, 36:55565–55581, 2023.

Shahriar, S., Lund, B. D., Mannuru, N. R., Arshad, M. A., Hayawi, K., Bevara, R. V. K., Mannuru, A., and Batool, L. Putting gpt-4o to the sword: A comprehensive evaluation of language, vision, speech, and multimodal proficiency. *Applied Sciences*, 14(17):7782, 2024.

Shi, J., Xu, M., Hua, H., Zhang, H., Ermon, S., and Leskovec, J. Tabdiff: a mixed-type diffusion model for tabular data generation. In *The Thirteenth International Conference on Learning Representations*, 2025.

Su, A., Wang, A., Ye, C., Zhou, C., Zhang, G., Chen, G., Zhu, G., Wang, H., Xu, H., Chen, H., Li, H., Lan, H., Tian, J., Yuan, J., Zhao, J., Zhou, J., Shou, K., Zha, L., Long, L., Li, L., Wu, P., Zhang, Q., Huang, Q., Yang, S., Zhang, T., Ye, W., Zhu, W., Hu, X., Gu, X., Sun, X., Li, X., Yang, Y., and Xiao, Z. Tablegpt2: A large multimodal model with tabular data integration, 2024. URL `https://arxiv.org/abs/2411.02059`.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M.-A., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. Llama 2: Open foundation and fine-tuned chat models, 2023. URL `https://arxiv.org/abs/2307.09288`.

Van Breugel, B. and Van Der Schaar, M. Position: Why tabular foundation models should be a research priority. In *International Conference on Machine Learning*, pp. 48976–48993. PMLR, 2024.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics, 2018.

Wang, A. X. and Nguyen, B. P. Ttvae: Transformer-based generative modeling for tabular data generation. *Artificial Intelligence*, pp. 104292, 2025.

Wang, Y., Feng, D., Dai, Y., Chen, Z., Huang, J., Ananiadou, S., Xie, Q., and Wang, H. Harmonic: Harnessing llms for tabular data synthesis and privacy protection. *arXiv preprint arXiv:2408.02927*, 2024.

Wei, J., Bosma, M., Zhao, V. Y., Guu, K., Yu, A. W., Lester, B., Du, N., Dai, A. M., and Le, Q. V. Finetuned language models are zero-shot learners, 2022a. URL `https://arxiv.org/abs/2109.01652`.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022b.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022c.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. Transformers: State-of-the-art natural language processing. In Liu, Q. and Schlangen, D. (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL `https://aclanthology.org/2020.emnlp-demos.6/`.

Xu, L., Skoularidou, M., Cuesta-Infante, A., and Veeramachaneni, K. Modeling tabular data using conditional gan. *Advances in neural information processing systems*, 32, 2019.

Yang, Y., Wang, Y., Liu, G., Wu, L., and Liu, Q. Unitabe: A universal pretraining protocol for tabular foundation model in data science. In *International Conference on Learning Representations*, 2024.

Zhang, H., Zhang, J., Shen, Z., Srinivasan, B., Qin, X., Faloutsos, C., Rangwala, H., and Karypis, G. Mixed-type tabular data synthesis with score-based diffusion in latent space. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=4Ay23yeuz0.

Zhang, S., Dong, L., Li, X., Zhang, S., Sun, X., Wang, S., Li, J., Hu, R., Zhang, T., Wu, F., et al. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*, 2023.

Zhang, T., Yue, X., Li, Y., and Sun, H. Tablellama: Towards open large generalist models for tables. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6024–6044, 2024b.

Zhang, X., Luo, S., Zhang, B., Ma, Z., Zhang, J., Li, Y., Li, G., Yao, Z., Xu, K., Zhou, J., et al. Tablellm: Enabling tabular data manipulation by llms in real office usage scenarios. *arXiv preprint arXiv:2403.19318*, 2024c.

Zhao, Z., Kunar, A., Birke, R., and Chen, L. Y. Ctab-gan: Effective table data synthesizing. In *Asian Conference on Machine Learning*, pp. 97–112. PMLR, 2021.

Zhao, Z., Birke, R., and Chen, L. Tabula: Harnessing language models for tabular data synthesis. *arXiv preprint arXiv:2310.12746*, 2023.

Zhao, Z., Kunar, A., Birke, R., Van der Scheer, H., and Chen, L. Y. Ctab-gan+: Enhancing tabular data synthesis. *Frontiers in big Data*, 6:1296508, 2024.

Zhong, V., Xiong, C., and Socher, R. Seq2sql: Generating structured queries from natural language using reinforcement learning. *arXiv preprint arXiv:1709.00103*, 2017.

A. Additional Details on Our Instruction Dataset

A.1. Details of the Datasets

Table 3 tabulates the details of the public datasets used to create our instruction dataset for tabular data generation.

Table 3. We sample 20 publicly available datasets to create our instruction dataset for tabular data generation. To ensure diversity, these datasets are sampled from 10 different topics. For each dataset (table), $\mathcal{R}$ and $\mathcal{C}$ denote the number of rows (samples) and the number of columns (features), respectively. TRAIN indicates whether a dataset is used during training.

TOPIC	DATASET	$\mathcal{R}$	$\mathcal{C}$	TRAIN
CONSUMER AND MARKET ANALYSIS	AMAZON TOP 50 BESTSELLING BOOKS (2009-2019)	550	7	✓
	BITCOIN BTC-USD STOCK DATASET	2836	7	✓
	CAR PRICE PREDICTION DATASET	1000	7	✓
	SUPERMARKET STORE BRANCHES SALES ANALYSIS	896	5	✗
HEALTHCARE AND MEDICAL RESEARCH	US HEALTH INSURANCE DATASET	1338	7	✓
	BREAST CANCER WISCONSIN	699	10	✓
	DIABETES	768	9	✓
	WDBC - BREAST CANCER DIAGNOSIS	569	31	✗
FINANCE AND CREDIT RISK ANALYSIS	ADULT INCOME (UCI CENSUS INCOME)	48842	15	✓
	BANK MARKETING	45211	17	✓
	CREDIT-G	1000	21	✗
EMPLOYMENT AND WORKFORCE ANALYTICS	FOOTBALL PLAYERS SEASON 2024	5935	7	✓
	JOB POSTING	1095	6	✓
REAL ESTATE AND HOUSING ECONOMICS	BOSTON HOUSING	506	14	✓
	CALIFORNIA HOUSING	20640	10	✗
ENERGY AND SMART BUILDING SYSTEMS	ROOM OCCUPANCY DATASET	2665	6	✓
TRANSPORTATION AND TRAVEL INDUSTRY	TOUR & TRAVELS CUSTOMER CHURN PREDICTION	954	7	✓
SOCIAL MEDIA ANALYTICS	TWITTER ASTRAZENECA ANTICOVID	1553	5	✗
CHEMISTRY AND ENVIRONMENTAL SCIENCE	QSAR-BIODEG	1055	42	✗
GENERAL MACHINE LEARNING BENCHMARKS	IRIS	150	5	✓

A.2. Example of Created Instruction in Our Instruction Dataset

An example of a created training instruction is shown in Figure 2.

A.3. Details of Metadata Generation for Our Instructions

As mentioned in the main paper, the metadata for each table includes a general description of the table and column-wise details. Some of the tables lack such metadata, and for some, various descriptions are available online. Our preliminary experimental results suggest the importance of high-quality metadata in steering LLMs for proper tabular data generation. Therefore, to ensure the quality of the metadata used in our instructions, we leverage GPT-4o for metadata generation.

Specifically, to unify the format of the descriptions and ensure that all required details (e.g., column name, column data type, etc.) are present in the generated description, we manually extract the general and column-wise descriptions for one of the tables. We then use this as context and design a template prompt as input to GPT-4o. This template prompt is shown in Figure 3, and it is used to obtain the table descriptions for all tables. After obtaining these descriptions from GPT-4o, we review all generated descriptions to ensure their quality and accuracy.

Please take a look at the instruction below which describes the task, and examine the input that provides context. Then, respond to the question accordingly.

###Instruction:
Your task is to generate synthetic tabular data based on the provided input table and its column-wise descriptions. The synthetic data should:
- Preserve the intra-column distribution (distribution of values within each column).
- Mimic the inter-column relationships (correlation or dependence between columns).
- Not duplicate or reuse any data from the original input table.
- Ensure the output data reflects a plausible extension of the same underlying data generation process.
Please note that the input is in CSV format, and each column is described in detail to guide your generation process.

###Table Description:
General Description:
This table represents metadata about the top 50 bestselling books on Amazon for each year from 2009 to 2019. The dataset includes information about each book's title, author, user rating, number of reviews, price, publication year, and genre. It provides insights into consumer preferences, pricing trends, and popular authors in the book market over an 11-year span.
Column-wise Details:
- Name: [Type: Textual] – Title of the book as listed on Amazon. This is a free-text string and may include subtitles or series names.
- Author: [Type: Textual] – Name(s) of the author(s) of the book. In some cases, this includes organizations (e.g., National Geographic Kids).
- User Rating: [Type: Numerical (Float)] – Average user rating for the book on Amazon, on a scale typically ranging from 1.0 to 5.0.
- Reviews: [Type: Numerical (Integer)] – Total number of user reviews submitted for the book on Amazon.
- Price: [Type: Numerical (Integer)] – Retail price of the book in US dollars. This can range from low-cost books to more expensive titles.
- Year: [Type: Numerical (Integer)] – The year in which the book appeared on the top 50 bestseller list. Ranges from 2009 to 2019.
- Genre: [Type: Categorical] – The general classification of the book, such as "Fiction" or "Non Fiction".

###Input Table (CSV format):
Name,Author,User Rating,Reviews,Price,Year,Genre
The Wright Brothers,David McCullough,4.7,6169,16,2015,Non Fiction
"Broke: The Plan to Restore Our Trust, Truth and Treasure",Glenn Beck,4.5,471,8,2010,Non Fiction
Eclipse (Twilight),Stephenie Meyer,4.7,5505,18,2009,Fiction
How to Win Friends & Influence People,Dale Carnegie,4.7,25001,11,2018,Non Fiction
Howard Stern Comes Again,Howard Stern,4.3,5272,16,2019,Non Fiction
The Girl Who Kicked the Hornet's Nest (Millennium Trilogy),Stieg Larsson,4.7,7747,14,2010,Fiction
"Adult Coloring Book Designs: Stress Relief Coloring Book: Garden Designs, Mandalas, Animals, and Paisley Patterns",Adult Coloring Book Designs,4.5,2313,4,2016,Non Fiction
Thomas Jefferson: The Art of Power,Jon Meacham,4.5,1904,23,2012,Non Fiction
"Oh, the Places You'll Go!",Dr. Seuss,4.9,21834,8,2017,Fiction
Harry Potter Paperback Box Set (Books 1-7),J. K. Rowling,4.8,13471,52,2016,Fiction

###Question:
Generate 10 synthetic rows that follow the same intra-column distributions and inter-column correlations as in the input table, using the column descriptions as a guide. The synthetic rows must be statistically consistent with the input data, but should not repeat or closely mimic any of the original records.

###Answer:
Name, Author, User Rating, Reviews, Price, Year, Genre
The Lost Symbol,Dan Brown,4.2,8747,19,2009,Fiction
StrengthsFinder 2.0,Gallup,4.0,5069,17,2016,Non Fiction
The Four Agreements: A Practical Guide to Personal Freedom (A Toltec Wisdom Book),Don Miguel Ruiz,4.7,23308,6,2019,Non Fiction
The Wonky Donkey,Craig Smith,4.8,30183,4,2018,Fiction
The Instant Pot Electric Pressure Cooker Cookbook: Easy Recipes for Fast & Healthy Meals,Laurel Randolph,4.3,7368,7,2017,Non Fiction
"The Mark of Athena (Heroes of Olympus, Book 3)",Rick Riordan,4.8,6247,10,2012,Fiction
Mockingjay (The Hunger Games),Suzanne Collins,4.5,26741,8,2011,Fiction
"Publication Manual of the American Psychological Association, 6th Edition",American Psychological Association,4.5,8580,46,2010,Non Fiction
Thug Kitchen: The Official Cookbook: Eat Like You Give a F*ck (Thug Kitchen Cookbooks),Thug Kitchen,4.6,11128,23,2014,Non Fiction
Dog Man: Fetch-22: From the Creator of Captain Underpants (Dog Man #8),Dav Pilkey,4.9,12619,8,2019,Fiction

Figure 2. An example of the instruction created in our dataset for tabular data generation.

B. Additional Experimental Results

In this section, we include additional experimental results that could not be presented in the main submission due to space limitations. To demonstrate that our proposed approach is model-agnostic, we also include instruction-tuning results with another base LLM, TableLlama (Zhang et al., 2024b).

B.1. Instruction-tuning with Another Base LLM

In this section, we provide additional experimental results using TableLlama (Zhang et al., 2024b) as our base LLM. TableLlama is pre-trained on a variety of table-based tasks, including question answering, reasoning, table fact verification, and table-to-text generation. It is considered a state-of-the-art open-source LLM for table-based tasks, outperforming GPT-3.5 and demonstrating competitive performance compared to GPT-4. TableLlama is obtained by fine-tuning LongLoRA 7B (Chen et al., 2024) on 3M table-based Q&A and reasoning instructions. Note that LongLoRA 7B itself is derived from Llama 2 (Touvron et al., 2023) by replacing vanilla attention with shift short attention, thereby increasing the context window size to 8192 tokens. We fine-tune TableLlama on our proposed instruction dataset for conditional generation using the Huggingface Transformers library (Wolf et al., 2020).

The results of instruction-tuning TableLlama on our dataset for tabular data generation are shown in Table 4 for the fidelity metric and in Table 5 for the utility metric. As one can see, the base LLM does not perform well on tabular data generation, even though it is trained on a large set of table-based tasks. In fact, these results emphasize the point that tabular data generation is a distinct task compared to Q&A and reasoning. However, after fine-tuning, the model's performance improves significantly in terms of both the fidelity and utility of the generated responses. This illustrates that the base LLM struggles

This prompt details a request for generating structured descriptions of tabular data, specifically for a CSV file. The generated descriptions are intended for integration into instructions used for training machine learning models.

Objective: To generate both a general overview and detailed column-wise descriptions for a provided CSV file, adhering to a predefined format.

Context and Format Example:

The following example illustrates the desired output format for a different CSV file, which describes historical data related to used cars:

###Table Description:
General Description: This table provides historical data related to used cars listed for resale. It includes attributes about the car's make, manufacturing year, usage, price information, fuel type, seller and ownership status. This dataset is commonly used for training machine learning models to predict the selling price of a car based on these features.

Column-wise Details:
Car_Name: [Type: Textual] – Name or brand/model of the car, e.g., "ritz", "ciaz", "swift".
Year: [Type: Numerical (Integer)] – Year the car was manufactured.
Selling_Price: [Type: Numerical (Float)] – The price (in lakhs of INR) at which the car was sold. This is the target variable in price prediction tasks.
Present_Price: [Type: Numerical (Float)] – The car's price when it was new (i.e., the original showroom price in lakhs).
Kms_Driven: [Type: Numerical (Integer)] – The total distance the car has been driven, in kilometers.
Fuel_Type: [Type: Categorical] – Type of fuel the car uses. Common values include "Petrol", "Diesel", and sometimes "CNG".
Seller_Type: [Type: Categorical] – Indicates whether the seller is a "Dealer" or an "Individual".
Transmission: [Type: Categorical] – Type of gearbox in the car, such as "Manual" or "Automatic".
Owner: [Type: Numerical (Integer)] – The number of previous owners (e.g., 0 for first-hand cars, 1 or more for second-hand or beyond).

Task: Given an attached CSV file, analyze its content to provide a comprehensive general description and detailed column-wise descriptions. The output must strictly follow the format exemplified above.

Figure 3. Template used to prompt GPT-4o for generating table descriptions.

to generate meaningful output in the context of tabular data generation, but after instruction tuning, the generated data improves significantly. It better follows the structure of the tabular data, and the output more closely mimics the intra-column distributions and inter-column relationships. Note that since the base LLM used in TableLlama (Llama2) is relatively outdated, even after instruction tuning, there remains a considerable performance gap compared to a strong commercial model like GPT-4o, which is trained on far more tokens and has significantly higher capacity.

Table 4. Fidelity result for synthetic data using TableLlama (Zhang et al., 2024b) as base LLM for our instruction tuning. Note that '–' indicates that the output of the base LLM (TableLlama) does not follow the structure of the tabular data, and therefore can not be used for fidelity calculation.

Dataset	Algorithm	Shape	Trends
California	TableLlama	–	–
	ITT-GEN (Ours)	78.57	79.75
	GPT-4o	94.8	86.55
Credit	TableLlama	–	–
	ITT-GEN (Ours)	60.23	37.25
	GPT-4o	90.99	80.15
Boston	TableLlama	–	–
	ITT-GEN (Ours)	75.84	75.63
	GPT-4o	89.92	89.63
Diabetes	TableLlama	–	–
	ITT-GEN	66.14	70.9
	GPT-4o	92.1	91.15

Table 5. Utility result for synthetic data using TableLlama (Zhang et al., 2024b) as base LLM for our instruction tuning. Average AUC and MAPE are reported as utility metrics. Note that '–' indicates that the output of the base LLM (TableLlama) can not be used to train a machine learning model on tabular data.

Dataset	TableLlama	GPT4o	Ours
Boston (↓)	–	0.187	0.257
California (↓)	–	0.334	0.428
Credit (↑)	–	0.767	0.487
Diabetes (↑)	–	0.773	0.721