Towards Better Understanding Table Instruction Tuning: Decoupling the Effects from Data versus Models

Anonymous ACL submission

Abstract

Recent advances in natural language processing have leveraged instruction tuning to enhance Large Language Models (LLMs) for table-related tasks. However, previous works train different base models with different training data, lacking an apples-to-apples comparison across the result table LLMs. To address this, we fine-tune base models from the Mistral, OLMo, and Phi families on existing public training datasets. Our replication achieves performance on par with or surpassing existing table LLMs, establishing new state-of-the-art performance on Hitab, a table question-answering dataset. More importantly, through systematic out-of-domain evaluation, we decouple the contributions of training data and the base model, providing insight into their individual impacts. In addition, we assess the effects of table-specific instruction tuning on general-purpose benchmarks, revealing trade-offs between specialization and generalization.

1 Introduction

Researchers have tried to improve table understanding abilities of Large Language Models (LLMs) through instruction tuning on table tasks, aiming to construct a generalist model for table understanding. Existing work adapts table understanding benchmarks for table instruction tuning (Zhang et al., 2024a; Zheng et al., 2024) or synthesize instruction-answer pairs using LLMs to fine-tune smaller scale models (Li et al., 2023; Zhang et al., 2024b; Wu et al., 2024).

However, these studies utilize different pre-trained model architectures, training datasets and evaluation benchmarks, making it difficult to determine the specific sources of their contributions. For example, TableLlama (Zhang et al., 2024a) is trained from LongLoRA (Chen et al., 2024), a variant of Llama 2 (Touvron et al., 2023). TableLLM (Zhang et al., 2024b) is based on CodeLlama Instruct. TableBenchLLM (Wu et al., 2024) employs models such as Llama 3.1 (Dubey et al., 2024) and Qwen2 (Yang et al., 2024), and models such as Table-GPT (Li et al., 2023) are closed-source trained on GPT-3.5 (Ouyang et al., 2022). In addition, these models use different training datasets, with some relying on existing benchmarks (e.g., TableLlama) and others introducing proprietary training datasets (e.g., Table-GPT). Furthermore, previous work chooses different sets of evaluation benchmarks and seldomly compares their models with others, leaving an unclear image of how much progress researchers have made in building table LLMs.

For a fair and comprehensive study, we fine-tune the same base models from the Mistral (Jiang et al., 2023), OLMo (Groeneveld et al., 2024), and Phi (Abdin et al., 2024) families using the respective training data from each work. Our experiments demonstrate that these fine-tuned models perform comparably to or outperform current table LLMs on their respective evaluation datasets. Notably, our models establish new state-of-the-art (SOTA) results on HiTab, a table question answering benchmark. Beyond replication, we conduct systematic evaluations on diverse table tasks, including table question answering (Table QA), table-to-text generation, table fact verification, and beyond. These tasks span real-world datasets (Nan et al., 2022; Cheng et al., 2022) and synthetic data proposed in existing works (Li et al., 2023; Wu et al., 2024). To assess trade-offs, we also evaluate these models on general instruction following and reasoning benchmarks, analyzing how specializing in table tasks affects their general capabilities. Our work disentangles the effects of training data and base models.

Specifically, we find that 1) The base models demonstrate competitive performance on table benchmarks compared to the fine-tuned models; 2) Certain training data (e.g. the training data from

TableLLM) consistently outperforms the others on the out-of-domain table tasks; 3) SOTA-chasing can be meaningless as the model's performance does not generalize to datasets in the same task category; 4) Transferability exists across table tasks; 5) The effects of the training data depend on the choice of base models, and strong base models lead to better performance; 6) Proper fine-tuning does not necessarily compromise the model's general capabilities. We hope our findings provide actionable insights into model selection and dataset construction for building effective table LLMs.

In summary, our contributions are three fold.

- We replicate existing table LLMs by fine-tuning various base models, achieving comparable or superior performance on benchmarks reported in the existing works, respectively.
- We decouple the contributions of the training data and base models, revealing that different base models perform differently with the same set of training data. We provide our findings on the effects of training data and base models.
- We expand the evaluation topology of table LLMs to general benchmarks, revealing that proper fine-tuning does not necessarily compromise the model's general capabilities.

2 Related Works

Table-Related Tasks. Earlier work has focused on extracting table content from HTML (Chen et al., 2000; Tengli et al., 2004). The deep learning era has seen more diverse table-related tasks such as table question answering (table QA), the task of answering a question given the table and certain context in the format of multiple-choice (Jauhar et al., 2016) and free-form answer (Nan et al., 2022); table fact verification, the task of determining whether a given claim is supported or refuted by the table content (Chen et al., 2019; Gupta et al., 2020); table-to-text, the task of generating a description given the table or some highlighted table cells (Parikh et al., 2020). These proposed benchmarks cover a diverse set of domains, including Wikipedia tables (Parikh et al., 2020), financial tables (Chen et al., 2021), scientific tables (Moosavi et al., 2021), which serve as invaluable sources for developing and testing general table understanding models.

Methods for Table Understanding. Researchers have explored various methods for table understanding in the past decade such as adapting the model's internal structures to align with table structures (Lebret et al., 2016; Liu et al., 2018; Yang et al., 2022), synthesizing a large table pre-training corpus and designing table-specific training objectives (Yin et al., 2020; Herzig et al., 2020). Recently, the LLM era has witnessed a paradigm shift for research on tables. As LLMs have inherent abilities on table understanding, researchers employ prompt engineering on these LLMs for better performance on tables (Chang and Fosler-Lussier, 2023; Deng et al., 2024). Another line of research involves instruction tuning LLMs by adapting existing table-related benchmarks. This leads to various table LLMs such as Table-GPT (Li et al., 2023), TableLlama (Zhang et al., 2024a), TableLlava (Zheng et al., 2024), and TableLLM (Zhang et al., 2024b). Among them, while TableLlama achieves decent performance on in-domain data, it suffers a significant performance drop on unseen table tasks (Zheng et al., 2024). Su et al. (2024) introduce TableGPT-2, a concurrent work to ours that utilizes synthesized training data for model training. Due to the overlapping timelines, we do not include their model in our study. Though representing tables as images is a promising direction, recent works reveal that its performance still lags behind representing tables as texts (Deng et al., 2024). Therefore, in this paper, we focus on the text representation of tables.

Model	Base Model	Data Size	Data Source	Open Model?	Open Data?
TableGPT (2023)	-	-	-	✗	✗
Table-GPT (2023)	GPT-3.5	66K	S	✗	✓
TableLlama (2024a)	LongLoRA 7B†	2M	R	✓	✓
TableLLM (2024b)	CodeLlama 7B & 13B Instruct	80.5K	R + S	✓	✓
TableBenchLLM (2024)	Llama 3.1-8B & others	20K	S	✓	✓

Table 1: Information for current table instruction tuned models. For the "Data Source", "S" represents synthesized data, and "R" represents the real data. †: TableLlama has adopted the LongLoRA (Chen et al., 2024) base model, which is a variant based on the Llama 2 7B model with a longer context window.

3 Replicating Existing Table LLMs

Issues with Comparing Existing Table LLMs Table 1 outlines the base models used in existing table LLMs. These base models, ranging from various Llama models to closed-source models such

Base Models	FeTaQA (BLEU)	HiTab (Acc)	TabFact (Acc)	FEVEROUS (Acc)	HybridQA (Acc)	KVRET ($F1_{Micro}$)	ToTTo (BLEU)	WikiSQL (Acc)	WikiTQA (Acc)
Original (Zhang et al., 2024a)									
LongLoRA 7B‡	**39.0**	64.7	82.5	73.8	**39.4**	48.7	20.8	50.5	35.0
Ours									
Mistral v0.3 7B Instruct	38.7	**70.6**†	**86.8**	75.9	27.2	46.6	28.5	**64.5**	47.4
OLMo 7B Instruct	36.8	67.9	83.8	69.8	20.3	44.6	20.8	56.9	38.8
Phi 3 Small Instruct (7B)	38.1	63.6	86.2	**78.3**	33.6	**56.0**	**29.6**	63.3	**47.7**

Table 2: Performance comparison between the original TableLlama and our fine-tuned models from different model families on the in-domain tuned (left three columns) and out-of-domain (right six columns) datasets. The number is bold if it is the best among the four, and underscored if it is the second. †: we surpass the previous SOTA performance (64.7 by TableLlama) on HiTab.

as GPT-3.5, differ significantly in their architecture designs, model sizes, and training recipes. In addition, each table LLM introduces its own unique training data, making it challenging to disentangle the impact of the training data from that of the base model.

To explore the contribution of the training data used in existing table LLM works, we train the same base models on datasets utilized in each of the existing works. We first demonstrate that our implementation yields comparable or better results than the performance reported in the existing works in Section 4. We then evaluate our trained models across various setups in Sections 5 and 6.

Foundational LLM Selections. For the training data from each existing work, we fine-tune `Mistral-7B-Instruct-v0.3` (Jiang et al., 2023), OLMo 7B Instruct (Groeneveld et al., 2024) and Phi 3 Small Instruct (7B) (Abdin et al., 2024). Following Zhang et al. (2024a,b); Wu et al. (2024), we fine-tune all the models through full parameter fine-tuning.

Experimental Setups. To rule out the effects of the learning rate, we train all three models using a set of learning rates: 5e-5, 1e-5, 5e-6, 1e-6, 5e-7, 1e-7, 5e-8, and 1e-8. Empirically, we find that they achieve the best when the learning rate is 5e-7. We do not see significant performance changes as we increase the training steps. For consistency, we fine-tune our models for three epochs across all the experiments. More details in Appendix A.

4 Comparison of Our Models v.s. the Original Models

Here we report the performance of our fine-tuned models based on Mistral v0.3 7B Instruct, OLMo 7B Instruct, and Phi 3 Small Instruct (7B) versus the original models on the datasets reported in each of the original work.

4.1 Replicating TableLlama

Training Datasets. The original TableLlama (Zhang et al., 2024a) uses 2 million data points in its instruction tuning stage, which can be unnecessarily large. In addition, we do not have enough computing resources to instruction-tune our model on a dataset of such a scale. Therefore, we rule out the table operation datasets and only maintain the training data for FeTaQA (Nan et al., 2022), HiTab (Cheng et al., 2022), and TabFact (Chen et al., 2019) to fine-tune our model, which results in 107K training instances.

Evaluation Datasets. Following Zhang et al. (2024a), we use the FeTaQA (Nan et al., 2022), HiTab (Cheng et al., 2022), and TabFact (Chen et al., 2019) as the in-domain evaluation sets. In addition, we compare our fine-tuned models versus the original TableLlama on FEVEROUS (Aly et al., 2021), HybridQA (Chen et al., 2020b), KVRET (Eric and Manning, 2017), ToTTo (Parikh et al., 2020), WikiSQL (Zhong et al., 2017), and WikiTQ (Pasupat and Liang, 2015).

Comparison. Table 2 compares the original TableLlama model (first row) versus our fine-tuned models. Our fine-tuned models yield similar or better performance than the original TableLlama model in most cases. In addition, we achieve the new SOTA performance on HiTab by fine-tuning the Mistral model. As we only use 107K (5% of the 2M data points used by the original TableLlama), our results demonstrate that *with proper instruction-tuning, we can achieve compet-*

Base Models	WikiTQ$_m$ (Acc$_p$)	TATQA$_m$ (Acc$_p$)	FeTaQA$_m$ (BLEU)	OTT-QA$_m$ (Acc$_p$)
Original (Zhang et al., 2024b)				
CodeLlama‡	72.5	51.1	8.4	57.3
Ours				
Mistral	**76.0**	55.4	10.6	**64.3**
OLMo	66.8	50.2	10.5	58.1
Phi	75.4	**57.8**	**12.1**	63.3

Table 3: Performance comparison between the original TableLLM and our fine-tuned models. All four models are 7B and instruction-tuned. We denote the evaluation datasets with a subscript "m" as they are adapted by Zhang et al. (2024b).

itive results on table tasks with much fewer data.

4.2 Replicating TableLLM

Training Datasets. We use the original instruction-tuning set by Zhang et al. (2024b), which includes 80.5K training instances.

Evaluation Datasets. Following Zhang et al. (2024b), we use the modified version of WikiTQ (Pasupat and Liang, 2015), TATQA (Zhu et al., 2021), and FeTaQA (Nan et al., 2022) as the in-domain evaluation sets, and OTT-QA (Chen et al., 2020a) as the out-of-domain evaluation set.

Comparison. Table 3 compares the original TableLLM versus our fine-tuned models. We note that our evaluation metrics are distinct from what Zhang et al. (2024b) have used originally. Zhang et al. (2024b) use CritiqueLLM (Ke et al., 2024) as a judge to decide the correctness of the answers. However, the model judgments are made in Chinese[1], a different language from the language in all the training and evaluation datasets. In addition, the scores assigned by the CritiqueLLM is not consistent for a single evaluation example. Therefore, for WikiTQ$_m$, TATQA$_m$, and OTT-QA$_m$, we report the Acc$_p$ scores, where we calculate whether the gold answer entities appear in the model's response. We find that our fine-tuned models based on the Mistral and Phi models consistently outperform the original TableLLM model on these datasets, and we attribute the performance improvement to the stronger base model (Mistral v0.3 7B Instruct and Phi 3 Small Instruct) we have versus theirs (CodeLlama 7B Instruct).

[1]Zhang et al. (2024b)'s inference results are available at https://github.com/RUCKBReasoning/TableLLM/blob/main/inference/results/TableLLM-7b/Grade_fetaqa.jsonl

Base Models	TableBench$_{eval}$ (R-L)
Original (Wu et al., 2024)	
Llama 3.1 8B ‡	27.2
Ours	
Mistral v0.3 7B Instruct	27.2
OLMo 7B Instruct	19.3
Phi 3 Small Instruct (7B)	**27.8**

Table 4: Performance comparison between the original TablebBenchLLM based on Llama 3.1 8B and our fine-tuned models. "R-L" denotes the ROUGE-L score.

Base Models	Beer (F1)	DeepM (Recall)	DI (Acc)	ED (F1)	C (F1)	CF (Acc)	Wiki (Acc)	CTA (F1)
Original (Li et al., 2023)								
GPT-3.5‡	72.7	**100.0**	**55.8**	**56.5**	29.4	**71.3**	**48.6**	**88.6**
Ours								
Mistral	**100.0**	98.0	46.4	46.0	23.8	25.3	25.5	68.3
OLMo	96.2	**100.0**	45.4	35.3	19.9	29.3	16.4	62.5
Phi	98.9	98.8	49.4	55.4	24.8	45.2	30.0	68.3

Table 5: Performance comparison between the original Table-GPT and our fine-tuned models.

4.3 Replicating TableBenchLLM

Training Datasets. We use the original instruction-tuning set by Wu et al. (2024), which includes 20K training instances.

Evaluation Datasets. Following Wu et al. (2024), we only evaluate the model on their constructed test set, which we denote as TableBench$_{eval}$ in Table 4.

Comparison. Following Wu et al. (2024), we report the ROUGE-L score of our Mistral-TableBenchLLM. In Table 4, we compare our model with the scores reported by Wu et al. (2024) in the original paper, corresponding to the version of TableBenchLLM fine-tuned based on Llama 3.1 8B model. Our Mistral-TableBenchLLM and Phi-TableBenchLLM achieve similar performance scores of 27.2 and 27.8, respectively, compared to the original TableBenchLLM's 27.2.

4.4 Replicating Table-GPT

Training Dataset. We use the instruction-tuning dataset provided by Li et al. (2023) that contains 66K instances.

	Beer (F1)	DeepM (Recall)	DI (Acc)	ED (F1)	C (F1)	CF (Acc)	Wiki (Acc)	CTA (F1)
13K	98.9	92.9	45.9	43.8	**29.4**	21.2	29.2	66.8
66K	**100.0**	**98.0**	**46.4**	**46.0**	23.8	**25.3**	**29.8**	**68.3**

Table 6: Performance comparison between training Mistral v0.3 7B Instruct on 13K instances versus 66K instances provided by Li et al. (2023).

Evaluation Datasets. We select four in-domain test sets by Li et al. (2023), Beer for entity matching, DeepM for schema matching, Spreadsheet-DI (DI) for data imputation, and Spreadsheet-Real (ED) for error detection. Furthermore, we report the out-of-domain performance on Column-No-Separator (C) for missing value identification, Spreadsheet-CF (CF) for column finding, WikiTQ (Wiki) for table question answering, and Efthymiou (CTA) for column type annotation.

Comparison. Table 5 reports the results. We note that though the size of our fine-tuned models are all 7B, they achieve better performance than Table-GPT which is based on GPT-3.5 on Beer, and comparable performance on DeepM. However, on the out-of-domain datasets, we can see that Mistral-TableGPT underperforms the original Table-GPT. We attribute such performance differences to the differences between the base models. Since GPT-3.5 is stronger than these open-source 7B models, its innate table understanding ability as well as its generalization ability leads to better performance on these out-of-domain table datasets for Table-GPT. This reinforces our motivations of conducting the comparisons using the same base model, as *the performance difference may be because of the base model's capability*, therefore we need the same base model to conduct an apple-to-apple comparison.

Side Findings. There is a smaller training set provided by Li et al. (2023) containing 13K training instances. We report the performance comparison by training the Mistral v0.3 7B Instruct model on the two sets in Table 6 We do not find a significant performance boost when we use the larger 66K dataset. And on one of the out-of-domain datasets, C, training on 13K instances even yields a better score of 29.4 than training on 66K instances' 23.8. This echoes with the findings by Zhou et al. (2024) that limited instruction tuning instances are able to yield a strong model.

5 Out-of-Domain Table Tasks Evaluation

In Sections 5 and 6, we evaluate our fine-tuned models from Section 4.

5.1 Datasets

We evaluate these models on eight existing real-world datasets covering the tasks of table question answering (table QA), table fact verification, and table-to-text generation. **FeTaQA (FeT)** (Nan et al., 2022) is a free-form table QA dataset sourced from Wikipedia-based tables. **HiTab (HiT)** (Cheng et al., 2022) is a table QA dataset sourced from statistical reports and Wikipedia pages on hierarchical tables. **TabMWP (TabM)** (Lu et al., 2022) is an open-domain grade-level table question-answering dataset involving mathematical reasoning. **TATQA (TAT)** (Zhu et al., 2021) is a table QA dataset sourced from real-world financial reports. **WikiTQ (Wiki)** (Pasupat and Liang, 2015) is a table QA dataset sourced from Wikipedia. **TabFact (TabF)** (Chen et al., 2019) is a table fact verification dataset sourced from Wikipedia. **InfoTabs (Inf)** (Gupta et al., 2020) is a table fact verification dataset with human-written textual hypotheses based on tables extracted from Wikipedia info-boxes. **ToTTo (ToT)** (Parikh et al., 2020) is a table-to-text dataset sourced from Wikipedia tables.

In addition, we evaluate these models on eight synthesized datasets including **Beer**, **DeepM**, **Spreadsheet-DI (DI)**, **Spreadsheet-Real (ED)**, **Column-No-Separator (C)**, **Spreadsheet-CF (CF)**, and **Efthymiou (CTA)** (Li et al., 2023) on schema reasoning ability (introduced in Section 4.4), and **TabB$_{\text{eval}}$** (Wu et al., 2024) on miscellaneous table tasks. We include a more detailed evaluation setup in Appendix B and provide examples from these datasets in Appendix E.

5.2 Results for Same Base Model, Different Training Data

Table 7 presents our fine-tuned Mistral models' performance on various out-of-domain table datasets. Appendix D presents the performance of all the models we have fine-tuned in Section 4.

Base model's performance. We find that *the base model is a strong baseline*. In Table 7, the original Mistral model maintains the best performance on five out-of-domain table datasets. For the original OLMo and Phi model in Table 10,

Train Data	Real								Synthesized							
	Table QA					Fact Veri.		Tab2T	Schema Reasoning							Misc.
	FeT	HiT	TabM	TAT	Wiki	TabF	Inf	ToT	Beer	DeepM	DI	ED	C	CF	CTA	$\text{TabB}_{\text{eval}}$
	BLEU	Acc	Acc	Acc	Acc	Acc	Acc	BLEU	F1	Recall	Acc	F1	F1	Acc	F1	R-L
👑 N/A	**20.0**	35.5	66.9	18.0	27.9	**62.3**	**42.8**	11.5	**97.2**	42.9	27.9	24.1	**30.2**	19.1	63.8	18.9
TableLlama	38.7	70.6	71.2	5.6	23.8	86.8	27.7	**28.5**	25.8	70.0	13.4	25.1	17.4	0.5	34.9	**19.6**
👑 TableLLM	10.2	**44.1**	**75.0**	25.0	32.3	11.9	15.4	6.7	45.0	**78.6**	**33.1**	**43.1**	25.6	15.0	66.9	3.7
TableBench	7.9	**44.1**	70.6	**25.7**	**37.4**	36.5	27.5	3.5	88.5	50.0	32.0	20.3	27.4	13.3	**72.2**	27.2
TableGPT	19.5	35.8	62.2	14.1	25.5	61.4	35.8	4.5	100.0	98.0	46.4	46.0	23.8	**25.3**	68.3	13.1

Table 7: Out-of-domain evaluation of Mistral v0.3 7B Instruct model fine-tuned with different training data. “N/A” denotes the untuned Mistral v0.3 7B Instruct model. The number is in gray if the “train data” includes the training set for the corresponding dataset. “👑” indicates the training data that leads to the most number of top performance for these table datasets. Under “Train Data”, names refer to the datasets used in Section 4 for fine-tuning (e.g. TableLlama refers to the training data in Section 4.1).

Table	Year Name Translated Name Type 1961 Hallaç Carder Short ...
Q	How many works did Leyla Erbil publish in total?
Gold	Leyla Erbil published a total of 11 works. *This can be determined by counting the number of entries in the “Name” column in the provided table.*

Table 8: An example of the training example from TableLLM. The reasoning part is in italics.

they demonstrate competitive performance compared to the fine-tuned models. For instance, on the InfoTabs dataset, the untuned Phi model yields 62.3 compared to the best performance of 67.0. The original OLMo model achieves 50.5 on the Beer dataset, outperforming the fine-tuned models that are based on OLMo. This demonstrates that through pre-training and general instruction tuning, these models have acquired innate table understanding ability, which echos with the findings by Li et al. (2023); Deng et al. (2024)

Table QA tasks. TableLLM’s training data consistently achieves the best (e.g. on HiTab in Table 7) or competitive performance on table QA tasks across all three base models in Table 10. In contrast, though the recipe for TableLlama’s training data contains table QA tasks, models trained with the training data from TableLlama underperform those from TableLLM. We attribute the effectiveness of TableLLM’s training data on the table QA task to that when constructing the data, Zhang et al. (2024b) leverage LLMs such as GPT-3.5 to enhance the reasoning process. For example, in the training instance in Table 8, in addition to answering the question, the adjusted gold answer incorporates the reasoning of “counting the number of entries”. Such training data teach the base models of the underlying reasoning process to reach to the final answer, therefore benefiting its table QA ability.

Table fact verification tasks. When fine-tuned on the Mistral model, TableGPT’s training data, while slightly underperforming the base model, exhibits the least performance decay. In addition, in Table 10, fine-tuning on TableGPT’s training data achieves the best out-of-domain table fact verification performance for both the OLMo and Phi models. Interestingly, despite TableBench and TableLlama’s training data including table fact verification examples, models trained with these datasets still underperform the base model on the InfoTabs dataset, achieving scores of 27.7 and 27.5 compared to the base model’s 42.8 (Table 7). This pattern is consistent across all three base models. Notably, TableGPT’s training data do not explicitly include table fact verification examples, yet they yield the most significant improvements. We hypothesize that the key to success lies in the reasoning process rather than the superficial similarity of task formats. Although TableLlama’s training data include tasks like TabFact, which involve table fact verification, the model may rely on domain-specific patterns rather than the authentic reasoning process to output labels such as “entail” or “refute”. In addition, we highlight that the original TableLlama model, as the previous SOTA model on TabFact, yields 2.85 on InfoTabs as reported by Zheng et al. (2024). Our fine-tuned Mistral model, though outperforming the original TableLlama on the TabFact dataset,

Base Model	*Real*								*Synthesized*							
	Table QA					**Fact Veri.**		**Tab2T**	**Schema Reasoning**							**Misc.**
	FeT	HiT	TabM	TAT	Wiki	TabF	Inf	ToT	Beer	DeepM	DI	ED	C	CF	CTA	$\text{TabB}_{\text{eval}}$
	BLEU	Acc	Acc	Acc	Acc	Acc	Acc	BLEU	F1	Recall	Acc	F1	F1	Acc	F1	R-L
M	10.2	44.1	75.0	**25.0**	32.3	11.9	15.4	6.7	45.0	**78.6**	33.1	**43.1**	25.6	15.0	66.9	3.7
O	9.7	35.5	65.5	17.7	26.7	40.6	16.9	**8.9**	16.5	42.9	33.0	37.6	13.0	18.7	43.6	6.3
P	**18.2**	**45.3**	**81.2**	24.1	**37.7**	**69.6**	**44.6**	8.1	**80.2**	50.0	**34.0**	41.3	**27.9**	**49.5**	**70.1**	**27.2**

Table 9: Out-of-domain evaluation for different table tasks. Here the models are all trained on the TableLLM training data. In terms of the base models, "M", "O", and "P" represent Mistral v0.3 7B Instruct, OLMo 7B Instruct, and Phi 3 Small Instruct (7B), respectively. We make the number bold if it is the best among the three.

underperforms the untuned Mistral model on InfoTabs. *Such results highlight the limitations of the SOTA-chasing works* as even TableLlama and our Mistral-TableLlama achieve competitive performance on TabFact, these models still do not generalize to datasets in the same task category.

Transferability across table tasks. As aforementioned, though TableGPT's training data do not explicitly include table fact verification tasks, tasks such as error detection and schema matching appear to positively contribute to table fact verification performance. Moreover, unlike the original TableLlama model, we do not instruction-tune the Mistral model on the extensive amount of table operation data as described in Section 4.1. Nevertheless, fine-tuning the model on tasks like table QA and table fact verification still results in a significant performance boost on DeepM, a schema-matching dataset, with a score of 70.0 compared to the base model's 42.9. Our finding aligns with Zhang et al. (2024a)'s observation that training solely on HiTab leads to better performance on certain table operation datasets, suggesting that transferability exists across table tasks.

5.3 Results for Same Training Data, Different Base Models

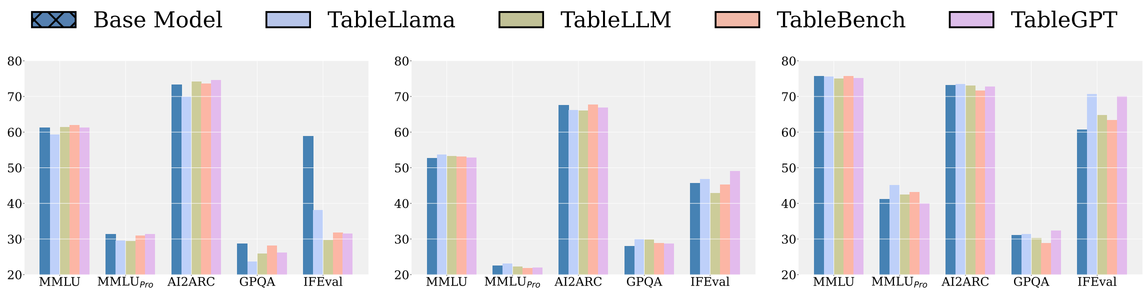
Table 9 presents the performance of models fine-tuned from three base models, all on the TableLLM training data.

The effects of the training data depend on the base model. In Table 9, when all fine-tuned on TableLLM's training data, there is a significant performance gap among the three models. For instance, on WikiTQ, fine-tuning the OLMo model yields 26.7, fine-tuning the Mistral model yields 32.3, while fine-tuning the Phi model yields 37.7. We attribute the performance difference to the varying innate capabilities of each model. In Table 11 in Appendix D, the original Phi model outperforms both the Mistral and OLMo model on four out of five general benchmarks, which aligns with its superior performance on most table tasks after fine-tuning. Moreover, as switching to stronger base models can lead to better performance even with the same training data, it is possible that existing models may not exhaust the potential of their corresponding training data in their originally reported results.

Strong base model leads to better performance. In Table 10, the best performance for a single dataset is typically achieved by fine-tuning the base model, which outperforms the other two models when untuned. For instance, TabMWP's overall best performance is achieved by fine-tuning the Phi model with the TableBench training data, and the original Phi model achieves 76.1, outperforming the original Mistral's 66.9 and the original OLMo's 54.4. TATQA's overall best performance is achieved by fine-tuning the Mistral model with TableBench training data, and the original Mistral model achieves 18.0, outperforming the original OLMo's 14.3 and the original Phi's 13.0. This suggests that *practitioners can select the base model by comparing their performance on downstream tasks prior to fine-tuning*, which can save the effort of training all candidate base models before deciding which one to use.

6 General Tasks Evaluation

In this section, we evaluate these models on general benchmarks to understand how table instruction tuning impacts the models' general capabilities. Ideally, a model should maintain its general capabilities as much as possible after the instruction tuning process, because a model that loses significant knowledge during tuning may struggle to serve end-users in real-world applications.



(a) Mistral v0.3 7B Instruct. (b) OLMo 7B Instruct. (c) Phi 3 Small Instruct (7B).

Figure 1: Performance of fine-tuned models trained on different data (e.g. TableLlama) on general benchmarks. The green and red hatched bars represent performance gains or losses relative to the base model, respectively. On IFEval, unlike other models, the Mistral model shows a significant performance drop, underscoring the impact of innate model capabilities on preserving general performance after domain-specific fine-tuning.

6.1 Datasets

MMLU (Hendrycks et al., 2021) examines the general ability of the model on 57 tasks including elementary mathematics, US history, computer science, etc. We adopt the 5-shot setup. **MMLU$_{\text{Pro}}$** (Wang et al., 2024) is an enhanced benchmark evaluating the general ability of the model, which contains up to ten options and eliminates the trivial questions in MMLU. We adopt the 5-shot setup. **AI2ARC** (Clark et al., 2018) is a reasoning benchmark containing natural, grade-school questions. We adopt the 0-shot setup and report the accuracy score on the challenging set. **GPQA** (Rein et al., 2023) is a reasoning benchmark containing questions in biology, physics, and chemistry written by domain experts. We adopt a 0-shot setup and report the accuracy score on its main set. **IFEval** (Zhou et al., 2023) is a dataset evaluating the general instruction following ability of the model containing instructions such as "return the answer in JSON format". We report the instance-level strict accuracy defined by Zhou et al. (2023). We include a more detailed setup in Appendix C for our evaluation process and provide examples from these datasets in Appendix E.

6.2 Results and Analysis

Figure 1 presents the results of our models on the general benchmarks. Table 11 in Appendix D presents the complete performance of the base model, our fine-tuned models, and the corresponding difference that we plot in Figure 1. We find that on MMLU, MMLU$_{\text{Pro}}$, AI2ARC, and GPQA, *our fine-tuned models do not compromise too much of the base models' general capabilities.* On AI2ARC, the score for Mistral-TableGPT is even slightly higher than the base model. Such performance improvement is likely due to the fact that many table tasks involve reasoning over tables, which may enhance the model's general reasoning ability. On IFEval, models fine-tuned from the Mistral model suffer a significant performance drop of over 20 points compared to the original model. However, models fine-tuned from the Phi model even improve the base model's performance. We attribute such discrepancy to the difference in the base model's innate characteristics. For instance, certain base models may be more robust in terms of acquiring new capabilities while maintaining their original capabilities, or the examples of the downstream tasks happen to align well with the examples the model has seen during its general training process. Our finding suggests that domain-specific tuning does not necessarily lead to performance decay on general benchmarks, and it heavily depends on the base model's innate characteristics. We provide additional discussion on the effects of model scales for both out-of-domain table tasks evaluation and general tasks evaluation in Appendix D.

7 Conclusion

To conduct an apples-to-apples comparison, we train three 7B Instruct models based on the table instruction tuning data proposed in the prior works. As a side product, we achieve the new SOTA performance on HiTab. We are the first to decouple the factors of training data versus base models and provide analysis on each side. In addition, we conduct evaluations on the general benchmarks to investigate how domain-specific fine-tuning may influence the model's general capabilities. We hope our work provides future directions for research on structured data.

Limitations

We believe our work presents a comprehensive evaluation over a diverse set of table benchmarks and the general benchmarks. In addition, we want to stress the massive training effort we have invested in, as the training data provided in the existing works can be as large as 100K. As a side product, we have achieved the new SOTA performance on HiTab dataset, and provide the first open-source model replication of existing closed-source table LLMs such as Table-GPT. However, there exists other datasets proposed by the researchers which can be further used to evaluate these models' capabilities, and by no means we can exhaust all of them in this paper. We encourage future efforts in comprehensively evaluating these table LLMs' capabilities, and we believe our work has laid a solid foundation to decoupling the contributions of training data and base models, and further enhancing our understanding of table instruction tuning.

Ethical Considerations

In this work, we isolate the contributions of training data proposed by the existing table LLMs by training the same base models and comparing their performance. The base models we have used in this work include Mistral v0.3 7B Instruct model (Jiang et al., 2023), OLMo 7B Instruct (Groeneveld et al., 2024), and Phi 3 Small Instruct (7B) (Abdin et al., 2024). We conduct additional studies on Phi 3 Mini Instruct (4B) in Appendix D. Foundational models like Mistral v0.3 7B Instruct model are susceptible to jail-breaking instructions (Wei et al., 2024) and may lead to harmful behaviors. Our objective in this work is to understand the limitations of the existing table instruction tuning, and we urge practitioners to stick to the good purpose when developing or using our models. Our replicated models can serve as baseline models for future research on structured data, and we provide a holistic evaluation of these models on both table tasks and how they compromise their general capabilities. Our results lead to various findings on what training data helps the models most on these table tasks, and how to construct LLMs specialized in tables efficiently.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. The fact extraction and VERification over unstructured and structured information (FEVEROUS) shared task. In *Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER)*, pages 1–13, Dominican Republic. Association for Computational Linguistics.

Shuaichen Chang and Eric Fosler-Lussier. 2023. How to prompt llms for text-to-sql: A study in zero-shot, single-domain, and cross-domain settings. *arXiv preprint arXiv:2305.11853*.

Hsin-Hsi Chen, Shih-Chung Tsai, and Jin-He Tsai. 2000. Mining tables from large scale HTML texts. In *COLING 2000 Volume 1: The 18th International Conference on Computational Linguistics*.

Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Wang, and William W Cohen. 2020a. Open question answering over tables and text. *arXiv preprint arXiv:2010.10439*.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2019. Tabfact: A large-scale dataset for table-based fact verification. *arXiv preprint arXiv:1909.02164*.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. 2020b. HybridQA: A dataset of multi-hop question answering over tabular and textual data. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1026–1036, Online. Association for Computational Linguistics.

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. 2024. Longlora: Efficient fine-tuning of long-context large language models. In *The International Conference on Learning Representations (ICLR)*.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. 2021. FinQA: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3697–3711, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhoujun Cheng, Haoyu Dong, Zhiruo Wang, Ran Jia, Jiaqi Guo, Yan Gao, Shi Han, Jian-Guang Lou, and Dongmei Zhang. 2022. HiTab: A hierarchical table dataset for question answering and natural language generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*

(Volume 1: Long Papers), pages 1094–1110, Dublin, Ireland. Association for Computational Linguistics.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Naihao Deng, Zhenjie Sun, Ruiqi He, Aman Sikka, Yulong Chen, Lin Ma, Yue Zhang, and Rada Mihalcea. 2024. Tables as texts or images: Evaluating the table reasoning ability of LLMs and MLLMs. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 407–426, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Mihail Eric and Christopher D Manning. 2017. Key-value retrieval networks for task-oriented dialogue. *arXiv preprint arXiv:1705.05414*.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*.

Vivek Gupta, Maitrey Mehta, Pegah Nokhiz, and Vivek Srikumar. 2020. INFOTABS: Inference on tables as semi-structured data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2309–2324, Online. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Sujay Kumar Jauhar, Peter Turney, and Eduard Hovy. 2016. Tabmcq: A dataset of general knowledge tables and multiple-choice questions. *arXiv preprint arXiv:1602.03960*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Pei Ke, Bosi Wen, Andrew Feng, Xiao Liu, Xuanyu Lei, Jiale Cheng, Shengyuan Wang, Aohan Zeng, Yuxiao Dong, Hongning Wang, Jie Tang, and Minlie Huang. 2024. CritiqueLLM: Towards an informative critique generation model for evaluation of large language model generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13034–13054, Bangkok, Thailand. Association for Computational Linguistics.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1203–1213, Austin, Texas. Association for Computational Linguistics.

Peng Li, Yeye He, Dror Yashar, Weiwei Cui, Song Ge, Haidong Zhang, Danielle Rifinski Fainman, Dongmei Zhang, and Surajit Chaudhuri. 2023. Table-gpt: Table-tuned gpt for diverse table tasks. *arXiv preprint arXiv:2310.09263*.

Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. 2018. Table-to-text generation by structure-aware seq2seq learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. 2022. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. *arXiv preprint arXiv:2209.14610*.

Nafise Sadat Moosavi, Andreas Rücklé, Dan Roth, and Iryna Gurevych. 2021. Scigen: a dataset for reasoning-aware text generation from scientific tables. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Hailey Schoelkopf, Riley Kong, Xiangru Tang, Mutethia Mutuma, Ben Rosand, Isabel Trindade, Renusree Bandaru, Jacob Cunningham, Caiming Xiong, Dragomir Radev, and Dragomir Radev. 2022. FeTaQA: Free-form table question answering. *Transactions of the Association for Computational Linguistics*, 10:35–49.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of the 2020*

Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186, Online. Association for Computational Linguistics.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480, Beijing, China. Association for Computational Linguistics.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2023. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*.

Aofeng Su, Aowen Wang, Chao Ye, Chen Zhou, Ga Zhang, Guangcheng Zhu, Haobo Wang, Haokai Xu, Hao Chen, Haoze Li, et al. 2024. Tablegpt2: A large multimodal model with tabular data integration. *arXiv preprint arXiv:2411.02059*.

Ashwin Tengli, Yiming Yang, and Nian Li Ma. 2004. Learning table extraction from examples. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 987–993, Geneva, Switzerland. COLING.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Preprint*, arXiv:2406.01574.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2024. Jailbroken: How does llm safety training fail? *Advances in Neural Information Processing Systems*, 36.

Xianjie Wu, Jian Yang, Linzheng Chai, Ge Zhang, Jiaheng Liu, Xinrun Du, Di Liang, Daixin Shu, Xianfu Cheng, Tianzhen Sun, Guanglin Niu, Tongliang Li, and Zhoujun Li. 2024. Tablebench: A comprehensive and complex benchmark for table question answering. *Preprint*, arXiv:2408.09174.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I. Wang, Victor Zhong, Bailin Wang, Chengzu Li, Connor Boyle, Ansong Ni, Ziyu Yao, Dragomir Radev, Caiming Xiong, Lingpeng Kong, Rui Zhang, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. UnifiedSKG: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 602–631, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Jingfeng Yang, Aditya Gupta, Shyam Upadhyay, Luheng He, Rahul Goel, and Shachi Paul. 2022. TableFormer: Robust transformer modeling for table-text encoding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 528–537, Dublin, Ireland. Association for Computational Linguistics.

Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. TaBERT: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8413–8426, Online. Association for Computational Linguistics.

Liangyu Zha, Junlin Zhou, Liyao Li, Rui Wang, Qingyi Huang, Saisai Yang, Jing Yuan, Changbao Su, Xiang Li, Aofeng Su, et al. 2023. Tablegpt: Towards unifying tables, nature language and commands into one gpt. *arXiv preprint arXiv:2307.08674*.

Tianshu Zhang, Xiang Yue, Yifei Li, and Huan Sun. 2024a. 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)*, pages 6024–6044, Mexico City, Mexico. Association for Computational Linguistics.

Xiaokang Zhang, Jing Zhang, Zeyao Ma, Yang Li, Bohan Zhang, Guanlin Li, Zijun Yao, Kangli Xu, Jinchang Zhou, Daniel Zhang-Li, et al. 2024b. Tablellm: Enabling tabular data manipulation by llms in real office usage scenarios. *arXiv preprint arXiv:2403.19318*.

Mingyu Zheng, Xinwei Feng, Qingyi Si, Qiaoqiao She, Zheng Lin, Wenbin Jiang, and Weiping Wang. 2024. Multimodal table understanding. *arXiv preprint arXiv:2406.08100*.

Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning. *arXiv preprint arXiv:1709.00103*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2024. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. 2021. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3277–3287, Online. Association for Computational Linguistics.

A Experimental Setup

We run our experiments on 1 server node with 8 A100, each with 48 GB GPU memory. We set the batch size to 16 in our training process.

B Out-of-Domain Evaluation Setup

For FeTaQA, we use the BLEU4 score following Nan et al. (2022). For ToTTo, we follow Xie et al. (2022) to report the BLEU4 scores over multiple references. We adopt the evaluation script from the original HiTab, TabMWP, TATQA, and WikiTQ repository on GitHub. For these table QA tasks, we notice that since the fine-tuned models may not follow instructions such as “generate in the JSON format”, we do not pose any constraints to these models in terms of the generation format. Instead, we use Haiku 3.5[2] to extract the answer entity from the model generation. For TabFact and InfoTabs, we report the accuracy by checking if only the gold answer appears in the prediction.

In terms of the test set format, we use the exact same test set for FeTaQA, HiTab, TATQA, and ToTTo as Zhang et al. (2024a) with the Markdown table format. For TabMWP, WikiTQ, and InfoTabs, etc., we follow the original data format. Specifically, TabMWP uses ‘|’ to separate columns, and WikiTQ and InfoTabs use HTML format to represent tables.

C General Evaluation Setup

For MMLU, MMLU_{Pro}, AI2ARC, and GPQA, as they are all multi-choice question-answering datasets, our objective is to select the most appropriate completion among a set of given options based on the provided context. Following Touvron et al. (2023), we select the completion with the highest likelihood given the provided context. As we evaluate the model based on their selection of the letter choice of “A”, “B”, etc., we do not normalize the likelihood by the number of characters in the completion.

D More Results

D.1 Out-of-domain Table Tasks Evaluation

Comparison across 7B models. Table 10 presents performance scores for our fine-tuned models from Section 4 and their corresponding

[2] https://www.anthropic.com/claude/haiku

Train Data	*Real*								*Synthesized*							
	Table QA					**Fact Veri.**		**Tab2Text**	**Schema Reasoning**							**Misc.**
	FeT	HiT	TabM	TAT	Wiki	TabF	Inf	ToT	Beer	DeepM	DI	ED	C	CF	CTA	$\text{TabB}_{\text{eval}}$
	BLEU	Acc	Acc	Acc	Acc	Acc	Acc	BLEU	F1	Recall	Acc	F1	F1	Acc	F1	ROUGE-L
Mistral v0.3 7B Instruct																
👑 N/A	**20.0**	35.5	66.9	18.0	27.9	**62.3**	**42.8**	11.5	**97.2**	42.9	27.9	24.1	**30.2**	19.1	63.8	18.9
TableLlama	38.7	70.6	71.2	5.6	23.8	86.8	27.7	**28.5**	25.8	70.0	13.4	25.1	17.4	0.5	34.9	**19.6**
👑 TableLLM	10.2	**44.1**	**75.0**	25.0	32.3	11.9	15.4	6.7	45.0	**78.6**	**33.1**	**43.1**	25.6	15.0	66.9	3.7
TableBenchLLM	7.9	**44.1**	70.6	**25.7**	**37.4**	36.5	27.5	3.5	88.5	50.0	32.0	20.3	27.4	13.3	**72.2**	27.2
TableGPT	19.5	35.8	62.2	14.1	25.5	61.4	35.8	4.5	100.0	98.0	46.4	46.0	23.8	**25.3**	68.3	13.1
OLMo 7B Instruct																
N/A	6.0	27.3	54.4	14.3	19.4	38.2	21.4	5.1	**50.5**	35.7	28.9	14.1	15.0	16.2	54.5	7.6
TableLlama	36.8	67.9	**72.9**	9.9	6.7	83.8	15.0	**20.8**	0.0	7.1	21.2	14.6	14.8	10.7	23.5	**17.1**
👑 TableLLM	**9.7**	**35.5**	65.5	**17.7**	26.7	40.6	16.9	8.9	16.5	**42.9**	33.0	**37.6**	13.0	18.7	43.6	6.3
TableBenchLLM	3.8	28.3	62.6	15.6	**34.0**	30.9	6.5	7.5	43.4	16.6	**36.6**	28.6	18.1	21.2	46.5	19.3
👑 TableGPT	9.3	27.2	65.6	14.6	16.4	**44.9**	**33.0**	11.4	96.2	100.0	45.4	35.3	**19.9**	**29.3**	**62.5**	13.7
Phi 3 Small Instruct (7B)																
N/A	5.0	39.6	76.1	13.0	29.7	65.3	62.3	1.4	95.0	42.9	31.9	49.7	30.6	43.4	71.5	8.3
TableLlama	38.1	63.6	74.8	18.3	**46.3**	86.2	54.3	**29.6**	**95.6**	35.7	4.3	19.4	27.9	36.5	43.9	22.4
👑 TableLLM	18.2	**45.3**	81.2	**24.1**	37.7	69.6	44.6	8.1	80.2	**50.0**	34.0	41.3	27.9	**49.5**	70.1	**27.2**
👑 TableBenchLLM	10.0	3.5	**83.0**	20.5	34.6	68.0	65.3	0.9	95.0	28.6	**35.9**	**53.8**	**31.1**	46.2	**76.7**	27.8
TableGPT	**24.8**	45.1	76.8	15.6	30.0	**71.0**	**67.0**	14.0	98.9	98.8	49.4	55.4	24.8	45.2	68.3	26.1

Table 10: Out-of-domain evaluation for different table tasks. The number is in gray if the model's training data contains the training set corresponding to the dataset. We make the number bold if it is the best among the same model, and the number red if it is the best across all the models. Mistral v0.3 7B Instruct, OLMo 7B Instruct, and Phi 3 Small Instruct (7B) indicate the base model on which we apply the training data, respectively. "👑" indicates the model has the most number of top performance across all the datasets with respect to the same base model.

base models. We find that when training with different base models in Table 10, TableLLM's training data consistently yield the best performance on the most out-of-domain table datasets.

Comparison across different model sizes. Figures 2 and 3 provide performance comparison between Phi 3 Mini Instruct (4B) versus Phi 3 Small Instruct (7B). We find that the larger sized model often leads to better performance for both the original model and the model after training on the same set of data.

D.2 General Tasks Evaluation

Comparison across 7B models. Table 11 presents the performance of the base model, our fine-tuned models, and the corresponding performance difference (also plotted in Figure 1). We find that the original model's capability typically decides the fine-tuned models' capability. With proper training, the original models' capability can be largely preserved even after domain-specific fine-tuning.

Comparison across different model sizes. Figure 4 provides the performance comparison between the Phi 3 Mini Instruct (4B) versus the Phi 3 Small Instruct (7B) model on the five general benchmarks. On most datasets, the 7B model outperforms the 4B model. However, on AI2ARC, the 4B model performs better, and on GPQA, the two models perform comparably. We note that on AI2ARC, we adopt a zero-shot setup, where we do not provide any examples to the models. The 7B model in this case may not prefer to answer their question at the very beginning, leading to an incorrect probability distribution over the four choices. For GPQA, as the task itself is challenging, both the 4B and 7B models cannot answer most of them, leading to a comparable performance.

E Dataset Examples

E.1 FeTaQA

Input:

```
[TLE] The Wikipedia page title of this
table is Gerhard Bigalk. The Wikipedia
section title of this table is Ships
attacked. [TAB] | Date | Name |
Nationality | Tonnage (GRT) | Fate | [
SEP] | 14 June 1941 | St. Lindsay |
United Kingdom | 5,370 | Sunk | [SEP] |
```

Method	MMLU	MMLU_{Pro}	AI2ARC	GPQA	IFEval
	Acc	Acc	Acc	Acc	Acc
M	61.2	31.4	73.3	28.6	58.8
M-TableLlama	59.4	29.5	69.6	23.7	38.0
Δ	↓ 1.9	↓ 1.9	↓ 3.4	↓ 4.9	↓ 20.7
M-TableLLM	61.4	29.3	74.2	25.9	29.6
Δ	↑ 0.2	↓ 2.0	↑ 0.9	↓ 2.7	↓ 29.1
M-TableBenchLLM	62.0	31.0	73.6	28.1	31.8
Δ	↑ 0.7	↓ 0.4	↑ 0.3	↓ 0.5	↓ 27.0
M-TableGPT	61.3	31.3	74.6	26.1	31.4
Δ	↑ 0.1	↓ 0.1	↑ 1.3	↓ 2.4	↓ 27.3
O	52.6	22.5	67.6	27.9	45.6
O-TableLlama	53.7	23.1	66.2	29.7	46.8
Δ	↑ 1.1	↑ 0.6	↓ 1.4	↑ 2.0	↑ 1.2
O-TableLLM	53.3	22.3	66.0	29.0	42.8
Δ	↑ 0.7	↓ 0.3	↓ 1.6	↑ 1.9	↓ 2.8
O-TableBenchLLM	53.1	21.9	67.7	28.6	45.2
Δ	↑ 0.5	↓ 0.7	↑ 0.1	↑ 0.9	↓ 0.4
O-TableGPT	52.9	21.9	66.8	28.8	48.9
Δ	↑ 0.3	↓ 0.6	↓ 0.8	↑ 0.8	↑ 3.4
P	75.7	41.2	73.1	31.0	60.7
P-TableLlama	75.5	45.1	73.5	31.5	70.1
Δ	↓ 0.2	↑ 3.9	↑ 0.3	↑ 0.4	↑ 9.9
P-TableLLM	75.0	42.6	73.1	30.4	64.8
Δ	↓ 0.7	↑ 1.3	↑ 0.0	↓ 0.8	↑ 4.1
P-TableBenchLLM	75.7	43.3	60.8	28.8	63.3
Δ	↑ 0.0	↑ 2.0	↓ 1.5	↓ 2.1	↑ 2.6
P-TableGPT	75.1	40.1	72.6	32.4	70.0
Δ	↓ 0.5	↓ 1.2	↓ 0.3	↑ 1.4	↑ 9.4

Table 11: Evaluation of the models on general benchmarks. "M-", "O-", and "P-" represent Mistral v0.3 7B Instruct, OLMo 7B Instruct, Phi 3 Small Instruct (7B), respectively. "Δ" denotes the performance difference between the instruction-tuned model and its base model.

21 December 1941 | HMS Audacity | Royal Navy | 11,000 | Sunk | [SEP] | 2 February 1942 | Corilla | Netherlands | 8,096 | Damaged | [SEP] | 4 February 1942 | Silveray | United Kingdom | 4,535 | Sunk | [SEP] | 7 February 1942 | Empire Sun | United Kingdom | 6,952 | Sunk | [SEP] | 16 May 1942 | Nicarao | United States | 1,445 | Sunk | [SEP] | 19 May 1942 | Isabela | United States | 3,110 | Sunk |\n\nThe highlighted cells of the table are: [HIGHLIGHTED_BEGIN] [11,000], [Sunk], [8,096], [Damaged] [HIGHLIGHTED_END] What happened to the two heaviest ships Gerhard Bigalk attacked?

Instruction:

This is a free-form table question answering task. The goal for this task is to answer the given question based on the given table and the highlighted cells.

Output:

Gerhard Bigalk damaged one ship of 8,096 GRT, and sunk one warship of 11,000 tons.

E.2 TabFact

Input:

[TLE] The table caption is about tony lema. [TAB] | tournament | wins | top - 5 | top - 10 | top - 25 | events | cuts made [SEP] | masters tournament | 0 | 1 | 2 | 4 | 4 | 4 | [SEP] | us open | 0 | 2 | 3 | 4 | 6 | 5 | [SEP] | the open championship | 1 | 2 | 2 | 2 | 3 | 3 | [SEP] | pga championship | 0 | 0 | 1 | 2 | 5 | 4 | [SEP] | totals | 1 | 5 | 8 | 12 | 18 | 16 |\n\nThe statement is: < tony lema be in the top 5 for the master tournament , the us open , and the open championship>. Is it entailed or refuted by the table above?

Instruction:

This is a table fact verification task. The goal of this task is to distinguish whether the given statement is entailed or refuted by the given table.

Output:

entailed

E.3 ToTTo

Input:

<page_title> List of Governors of South Carolina </page_title> <section_title> Governors under the Constitution of 1868 </section_title> <table> <cell> 76 < col_header> # </col_header> <col_header> 74 </col_header> <col_header> 75 </ col_header> </cell> <cell> Daniel Henry Chamberlain <col_header> Governor </ col_header> <row_header> 76 </row_header > </cell> <cell> December 1, 1874 < col_header> Took Office </col_header> < row_header> 76 </row_header> </cell> </ table>

Instruction:

This is a highlighted cells description task. The goal of this task is to generate the language description given table cells.

Output:

Daniel Henry Chamberlain was the 76th Governor of South Carolina from 1874.

E.4 Beer

Input:

```
Beer A is:\n|name|factory|\n|---|---|\n|Sierra Amber Ale|Silver Peak Restaurant \& Brewery|\n\nBeer B is:\n|name|factory|\n|---|---|\n|Sierra Andina Alpamayo Amber Ale|Sierra Andina|
\# Task Description: Please determine whether Beer A and Beer B refer to the same entity or not.
```

Instruction:

```
You are a helpful assistant that specializes in tables.\n Your final answer should be \'Yes\' or \'No\'. Return the final result as JSON in the format \{"answer": "<Yes or No>"\}. Let's think step by step and show your reasoning before showing the final result.
```

Output:

```
\{"answer": "No"\}
```

E.5 TabB$_{eval}$

Input:

```
Read the table below in JSON format:\n[TABLE] \n\{"columns": ["index", "organization", "year", "rank", "out of"], "data": [["bribe payers index", "transparency international", 2011, 19, 28], ["corruption perceptions index", "transparency international", 2012, 37, 176], ["democracy index", "economist intelligence unit", 2010, 36, 167], ["ease of doing business index", "world bank", 2012, 16, 185], ["economic freedom index", "fraser institute", 2010, 15, 144], ["economic freedom index", "the heritage foundation", 2013, 20, 177], ["global competitiveness report", "world economic forum", 20122013, 13, 144], ["global peace index", "institute for economics and peace", 2011, 27, 153], ["globalization index", "at kearney / foreign policy magazine", 2006, 35, 62], ["press freedom index", "reporters without borders", 2013, 47, 179], ["property rights index", "property rights alliance", 2008, 28, 115]]\}\n\nLet\'s get start!\nQuestion: What is the average rank of the indices published by Transparency International?
```

Instruction:

```
You are a helpful assistant that specializes in tables.\nYou are a table analyst. Your task is to answer questions based on the table content.\n\n\nThe answer should follow the format below:\n[Answer Format]\nFinal Answer: AnswerName1, AnswerName2...\n\nEnsure the final answer format is the last output line and can only be in the "Final Answer: AnswerName1, AnswerName2..." form, no other form. Ensure the "AnswerName" is a number or entity name, as short as possible, without any explanation.\n\n\nGive the final answer to the question directly without any explanation.
```

Output:

```
28
```

E.6 MMLU

Input:

```
{5-shot examples}
Find the degree for the given field extension Q(sqrt(2), sqrt(3), sqrt(18)) over Q.
\nA. 0\nB. 4\nC. 2\nD. 6\nAnswer:
```

Instruction:

```
The following are multiple choice questions (with answers) about abstract algebra.\n\n
```

Output:

```
B
```

E.7 IFEval

Input:

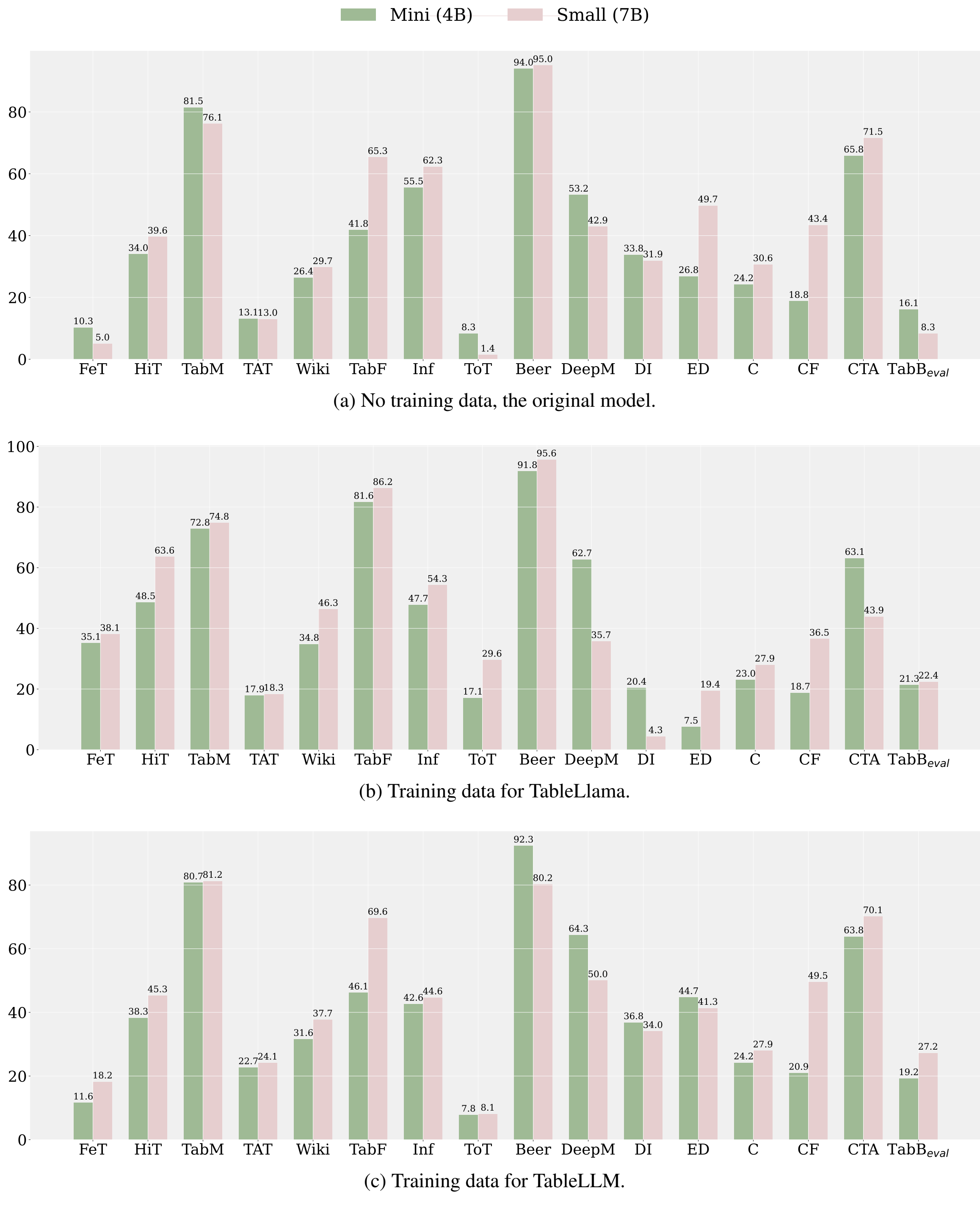
```
Can you help me make an advertisement for a new product? It's a diaper that's designed to be more comfortable for babies and I want the entire output in JSON format.
```

Instruction:

```
You are a helpful assistant.
```

Output:

```
[JSON formatted answer]
```

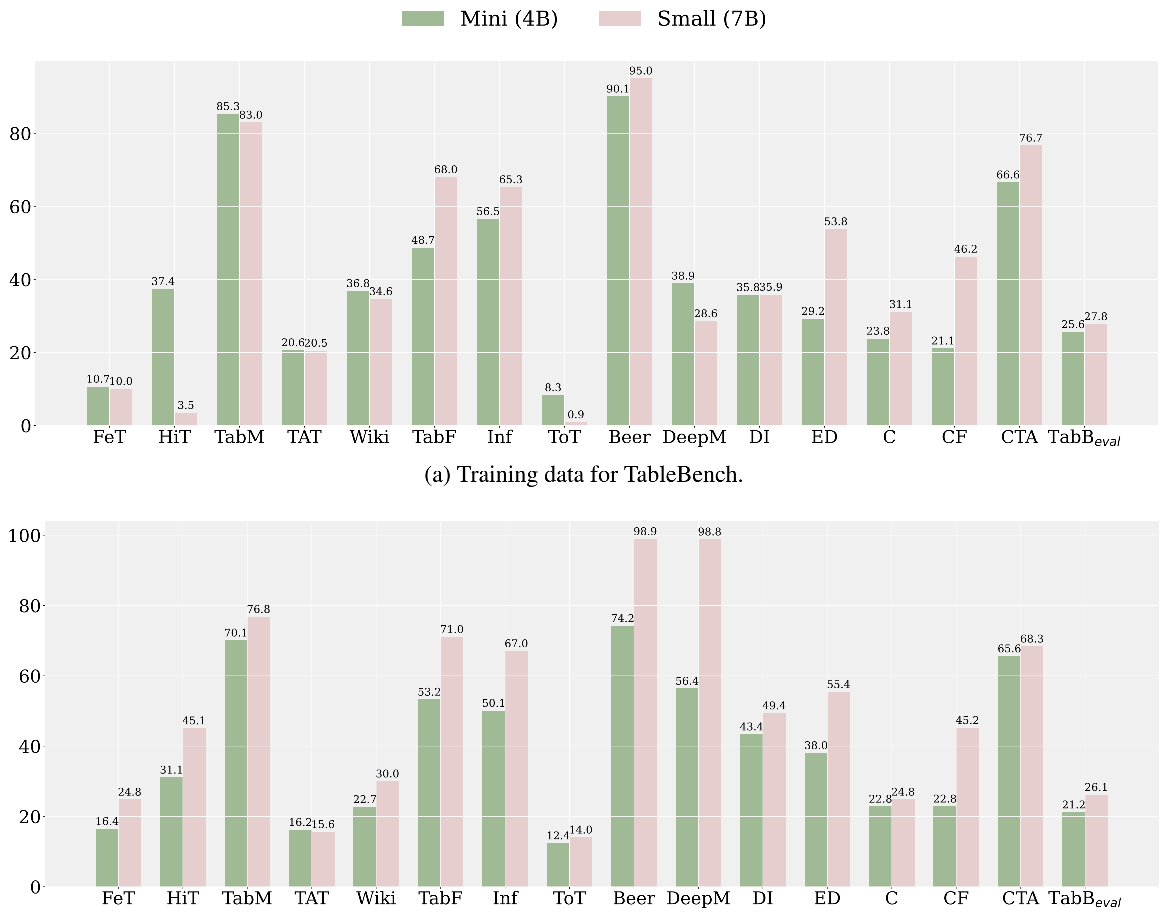


(a) No training data, the original model.

(b) Training data for TableLlama.

(c) Training data for TableLLM.

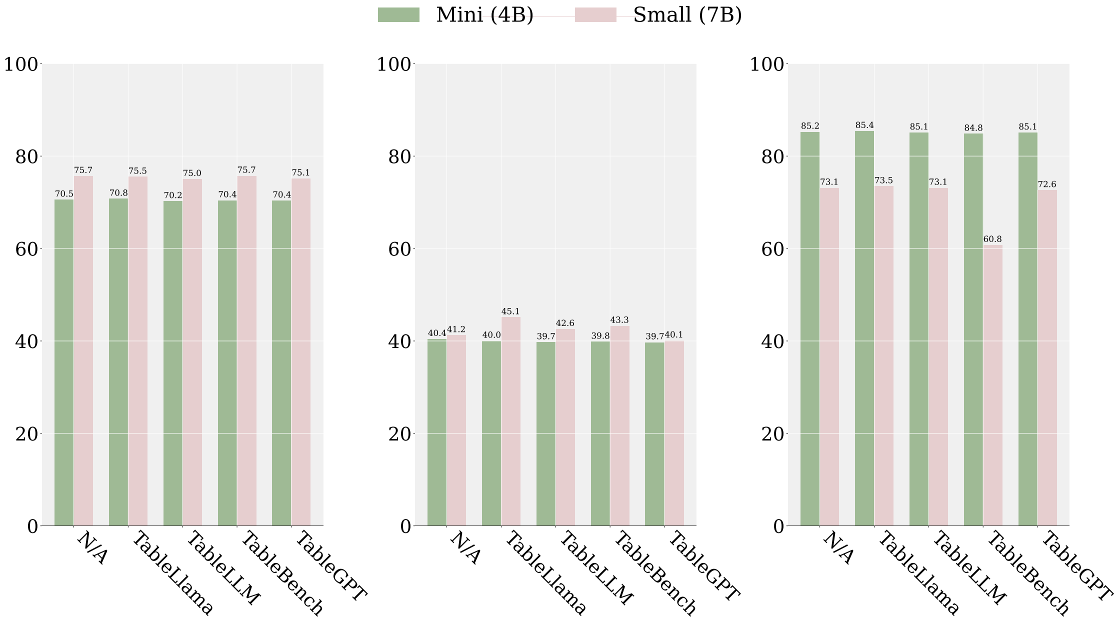
Figure 2: Performance of Phi 3 Mini Instruct (4B) versus Phi 3 Small Instruct (7B) model on different table tasks with different training data. In most cases, the 7B model outperforms the 4B model.



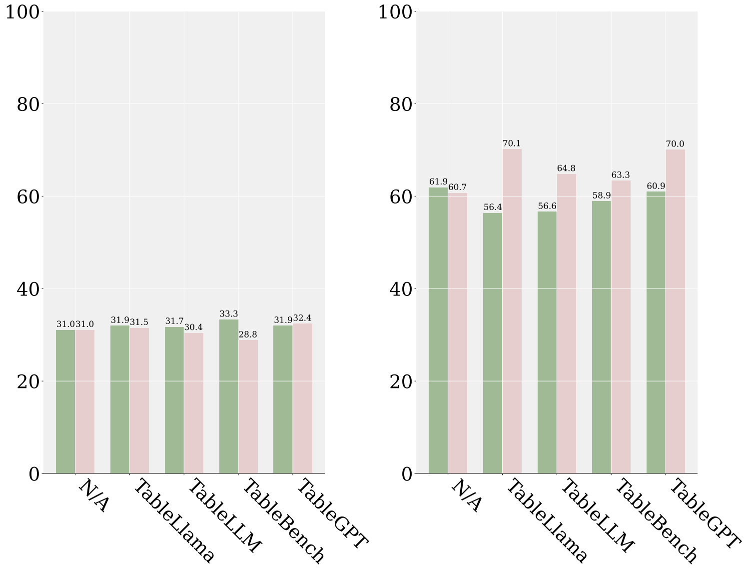
(a) Training data for TableBench.

(b) Training data for TableGPT.

Figure 3: Performance of Phi 3 Mini Instruct (4B) versus Phi 3 Small Instruct (7B) model on different table tasks with different training data. In most cases, the 7B model outperforms the 4B model.



(a) MMLU. (b) $\mathrm{MMLU_{Pro}}$. (c) AI2ARC.



(d) GPQA. (e) IFEval.

Figure 4: Performance difference between Phi 3 Mini Instruct (4B) versus Phi 3 Small Instruct (7B) model. On MMLU, $\mathrm{MMLU_{Pro}}$, IFEval, the Small (7B) version yields better performance both before and after fine-tuning. On GPQA, the two models perform comparably. On AI2ARC, the Mini (4B) version yields better performance.