Effective Distillation of Table-based Reasoning Ability from LLMs

Bohao Yang[1], Chen Tang[1,2], Kun Zhao[3], Chenghao Xiao[4], Chenghua Lin[1]
[1]Department of Computer Science, The University of Manchester, UK
[2]Department of Computer Science, The University of Surrey, UK
[3]Department of Electrical and Computer Engineering, University of Pittsburgh, US
[4]Department of Computer Science, The University of Durham, UK
j98519by@student.manchester.ac.uk, chen.tang@surrey.ac.uk,
kun.zhao@pitt.edu, chenghao.xiao@durham.ac.uk
chenghua.lin@manchester.ac.uk

Abstract

Large Language Models (LLMs) have demonstrated remarkable performance across a wide range of natural language processing tasks. However, their enormous parameter size and extremely high requirements for compute power pose challenges for their practical deployment. Recent research has revealed that specific capabilities of LLMs, such as numerical reasoning, can be transferred to smaller models through distillation. Some studies explore the potential of leveraging LLMs to perform table-based reasoning. However, there has been no prior work focusing on table reasoning skills in smaller models specifically tailored for scientific table-to-text generation tasks. In this paper, we propose a novel table-based reasoning distillation approach, with the aim of distilling LLMs into tailored smaller models. Our experimental results have shown that a 220 million parameter model (Flan-T5-base) fine-tuned using distilled data, not only achieves a significant improvement compared to traditionally fine-tuned baselines, but also surpasses specific LLMs on a scientific table-to-text generation dataset. Our code is available at **https://github.com/Bernard-Yang/DistillTableCoT**.

Keywords: table-based reasoning, distillation, table-to-text generation

1. Introduction

Tables, as a ubiquitous and pivotal means of knowledge storage, have been receiving increasing attention in contemporary research. Tabular data, when combined with textual data, provides a valuable and complementary source of information. The intersection of tabular and textual information constitutes a well-established problem within the domain of Natural Language Processing (NLP), with impacts spanning a diverse spectrum of downstream tasks, including table question answering (Pasupat and Liang, 2015; Cho et al., 2019; Nan et al., 2022), and table fact checking (Chen et al., 2020c; Gupta et al., 2020; Aly et al., 2021; Lu et al., 2023).

Conventional approaches to table-based reasoning (Pasupat and Liang, 2015; Zhong et al., 2017; Yu et al., 2018) have predominantly relied on the synthesis of executable languages such as SQL or SPARQL to facilitate information retrieval from tables. However, these symbolic languages often entail rigid assumptions regarding table structures, rendering them incapable of capturing the semantics embedded in textual segments within the table. A holistic comprehension of web tables necessitates the understanding of structured reasoning alongside textual reasoning. To this end, the emergence of table-based pre-trained models (Herzig et al., 2020; Liu et al., 2021; Jiang et al., 2022; Cai et al., 2022) has underscored the efficacy of pre-training models on both textual and tabular data for augmenting reasoning capabilities. This improvement stems from the extensive knowledge obtained from the large-scale crawling or synthesising of tabular and textual data.

In recent years, the advent of Large Language Models (LLMs) (Brown et al., 2020; Chowdhery et al., 2022; Touvron et al., 2023) has revolutionised the landscape of NLP, ushering in a new era marked by their remarkable performance demonstrated across a multitude of controllable text generation tasks (Tang et al., 2022; Yang et al., 2023; Zhao et al., 2023a; Tang et al., 2023b). Large Language Models (LLMs) implicitly capture the intricate interrelationships among tokens within input sequences, enabling them to adeptly comprehend the heterogeneous features present, regardless of their structural format, such as graph representations, tabular data, or sequential patterns(Huang et al., 2022; Goldsack et al., 2023; Tang et al., 2023a). These models leverage vast corpora of textual data and undergo extensive pre-training, exhibiting an exceptional capacity to tackle intricate mathematical and commonsense reasoning tasks, often within the context of few-shot and zero-shot learning scenarios (Wei et al., 2022; Wang et al., 2022; Drozdov et al., 2022; Loakman et al., 2023; Zhou et al., 2023).

LREC-COLING 2024, pages 5538–5550
20-25 May, 2024.

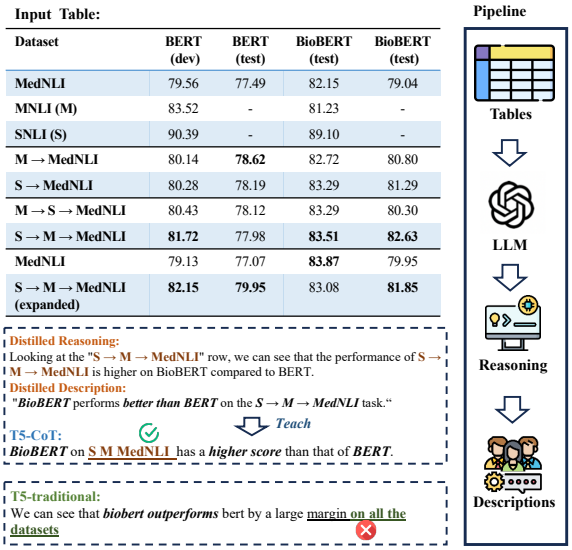

Figure 1: The overview of the distillation pipeline and example data. The pipeline includes using LLMs to generate table-based reasoning and descriptions given the input table.

Drawing inspiration from these groundbreaking developments, a range of studies (Chen, 2023; Ye et al., 2023; Cheng et al., 2023; Gemmell and Dalton, 2023; Lu et al., 2023) have emerged to highlight the competitive performance of LLMs in comparison to state-of-the-art fine-tuned models in the domain of table reasoning tasks (e.g., table question answering and table fact-checking). For instance, Zhao et al. (2023b) delved into the potential of employing LLMs augmented with Chain-of-Thought (CoT) techniques in the Logic-NLG dataset (Chen et al., 2020c) for table-to-text generation tasks. Despite significant advancements, prior research has not focused on the challenging domain of more complex reasoning-aware scientific table-to-text generation task using LLMs. Moreover, the substantial parameter count and demanding computational requirements present obstacles to their feasible implementation. Therefore, distilling LLMs' intrinsic table-based reasoning capabilities into more lightweight alternatives is a more efficient and resource-friendly approach.

In this paper, we investigate the capabilities of LLMs in the task of reasoning-aware scientific table-to-text generation, and propose a two-step distillation approach to transfer the table-based reasoning ability of LLMs into smaller models. The nature of the complex scientific table-to-text generation task requires the LLMs to comprehensively grasp the provided tables and engage in arithmetic reasoning encompassing both tabular and textual data, rather than merely converting table contents into superficial descriptions. Our distillation pipeline is shown in Figure 1, which includes using LLMs to generate table-based reasoning content and descriptions given the input table. We conduct our experiments on the SciGen dataset (Moosavi et al., 2021b), the first scientific table-to-text dataset and is more challenging than other standard table-to-text benchmarks, such as Wiseman et al. (2017), Parikh et al. (2020), and Chen et al. (2020a), as it contains more numerical reasoning. We also provide an example in Figure 1, in which the description generated by T5-CoT is better than that of T5-traditional, as T5-CoT is fine-tune with the reasoning and descriptions distilled from LLMs. This is because the example reasoning describes the "**S** → **M** → **Med**" row, which enables the model to focus on that specific row of the table in further fine-tuning the student models.

Our contributions can be summarised as follows:

- We explore the potential of tackling the task of reasoning-aware scientific table-to-text generation using LLMs.
- We propose a two-stage distillation framework containing data generation and fine-tuning stages. In the data generation stage, we utilise LLMs to generate table-based reasoning and factually consistent statements, which could describe the table correctly based on the input table, employing a one-shot Chain-of-Thought (CoT) methodology. Subsequently, in the fine-tuning phase, we employ the distilled CoT data generated by LLMs to imbue smaller models with table reasoning proficiency.
- We present a range of experimental results that underscore that fine-tuning smaller models with table-based reasoning data distilled from LLMs leads to significant performance enhancements compared to baseline models in the context of scientific table-to-text generation tasks.
- We demonstrate that, distillation empowers student models with as few as 220 million parameters (e.g., only 0.1% the size of teacher model) to outperform the 175 billion-parameter teacher model in certain metrics.

2. Related Work

2.1. Table-based Reasoning

Table-based reasoning tasks require the ability to reason over both natural language and structured tables. Traditional table-based reasoning involves employing semantic parsing to execute

commands on tables, with benchmarks including WikiTableQuestions (Pasupat and Liang, 2015), WikiSQL (Zhong et al., 2017), and Spider (Yu et al., 2018). These models are designed to produce SQL for interacting with tables. However, these languages impose strict criteria on tables and make it so that these methods cannot understand the semantics of text segments. Some works proposed to learn joint representations by pre-training on table and text data (Herzig et al., 2020; Liu et al., 2021; Zhao et al., 2022). Through pre-training the model on extensive synthetic data, they are able to achieve desirable performance on table related tasks. Recent works (Chen, 2023; Ye et al., 2023; Nan et al., 2023) have shown the ability of LLMs in table reasoning tasks through in-context learning. Lu et al. (2023) use LLMs to perform reasoning in the task of scientific table fact-checking. This task requires compositional reasoning using scientific tables as evidence. BINDER (Cheng et al., 2023) uses Codex to synthesise SQL queries to execute logical forms against tables in a question answering task.

2.2. Chain-of-thought Reasoning

Chain of thought (CoT) prompting encourages LLMs to break down a reasoning task into a series of intermediate steps, therefore enhancing reasoning abilities across various tasks (Wei et al., 2022; Shao et al., 2024). With a few CoT reasoning examples, LLMs can achieve state-of-the-art performance on complex arithmetic reasoning tasks. Self-consistency CoT (Wang et al., 2023) involves sampling multiple CoTs and selecting the most consistent one by beam searching. Kojima et al. (2022) propose zero-shot CoT by first generating CoT templates and producing the final answer with LLMs in a zero-shot setting.

2.3. Knowledge Distillation

Distillation has demonstrated its effectiveness in transferring valuable capabilities from a larger model to a smaller one (Hinton et al., 2015; Sanh et al., 2019; Zeng et al., 2022). Recent works have shown that synthetic data generated by the teacher model can effectively transfer the specialised abilities, such as numerical reasoning, to the student model. Chung et al. (2022) use manually generated CoT data to fine-tune a FLAN-based version of PaLM (Chowdhery et al., 2022). Fu et al. (2023) employ enriched chain-of-thought data to specialise a smaller model. Ho et al. (2023) proposes diverse CoT approach by sampling different reasoning outputs from a large model to then fine-tune a smaller model. Magister et al. (2023) use a two-step pipeline for transferring the reasoning capabilities of large models to smaller models. Hsieh et al. (2023) extract rationales from LLMs and integrated such data in the smaller model instruction tuning framework. Zhu et al. (2023) use LLMs to distill the programs, injecting reasoning ability into small models. We extend the above ideas into the table-based reasoning task, specifically in the scientific table-to-text generation domain, in which the generated CoT data leads to improved table reasoning performance.

3. Methodology

Our proposed framework is illustrated in Figure 2, which consists of two steps: synthesising data from LLMs and fine-tuning student models with the distilled data. The primary purpose of the first stage is to generate table-based reasoning and descriptions with LLMs given the input tables through CoT. In the second stage, the table-based reasoning ability is transferred into smaller models by fine-tuning with the distilled data from the LLMs.

3.1. Task Definition

We define the task as follows: The input serialised tabular data is denoted as T. In addition, the table-based reasoning data distilled from LLMs is denoted as $R = r_1, r_2, ..., r_n$, where r_i is the token of reasoning. The primary goal of this task is to generate a description $Y = y_1, y_2, ..., y_m$, where y_i is the token of the description and the model functions by simulating the conditional probability distribution $P(Y|T, R)$. The generated description should be factually consistent with the given table, and contain reasoning over the table.

3.2. Table-based Reasoning Generation

The data synthesis process of our proposed method is illustrated in the upper part of the right-hand side of Figure 2, which is based on in-context learning (Brown et al., 2020), an emergent ability of LLMs (Wei et al., 2022). Different from traditional fine-tuning, in-context learning enables the LLMs to make predictions based on the input context where only a few examples are demonstrated, without the need for parameter updating.

We utilise a large teacher LLM, `gpt-3.5-turbo`, to generate table-based reasoning through CoT. We formulate the data generation process as follows: given a input serialised table T, we prompt the LLMs with the one-shot CoT demonstration example to generate a reasoning R and a description Y which is factually consistent with the input table. Specifically, the demonstration examples $C = (T, R, Y)$ is a table, reasoning, and description triplet, where

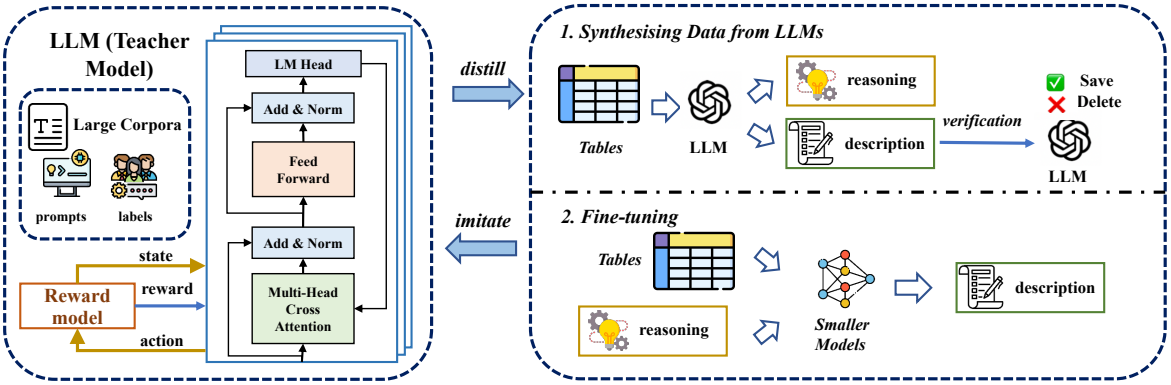

Figure 2: The overview of our framework. **For synthesising data from LLMs**, we provide table examples to LLMs, and use it to generate reasonings and descriptions. Then, the generated descriptions are verified by LLMs and the false reasoning and description pairs are removed. **For fine-tuning smaller models**, we fine-tune small models with generated reasoning and description, which inject the reasoning ability into smaller models.

the R and Y are hand-crafted. Finally, we can generate data as follows:

$$R_i, Y_i = \text{LLMs}(C, T_i) \quad (1)$$

where we prepend the demonstrated example C as the prefix to the input table T_i. Then the LLM will follow the instruction and learn the pattern from the example to generate corresponding reasoning R_i and description Y_i.

Diverse Reasoning. The table-to-text task enables the model to produce varied descriptions by focusing on different table regions or performing various reasoning operations, provided that the generated descriptions are factually consistent with the table (Zhao et al., 2023b). To maximise the reasoning ability distilled from LLMs, we employ the diverse reasoning approach (Ho et al., 2023; Zhu et al., 2023; Zhao et al., 2023b) to generate two different reasoning examples and descriptions for a given scientific table. We do not generate more reasoning-description pairs for each table because the maximal context limit of the LLMs and the average length of the tables and descriptions in the SciGen dataset is larger than in other table-to-text datasets. Specifically, the data generation process is shown as follows: given a context C and table T_i, the LLMs are required to generate two pairs of reasoning and description.

$$\{(R_1, Y_1), (R_2, Y_2)\} = \text{LLMs}(C, T_i) \quad (2)$$

Data Filtering. The synthesised table-based CoT data may contain incorrect samples due to the hallucination problem of generative models (Zhu et al., 2023). Therefore, we need to filter the wrongly generated CoT data. For filtering, we follow Madaan et al. (2023) and employ the Self-Refine method. To be specific, when generating a new set of data (R_i, Y_i) given T_i, we ask the LLMs to verify whether the generated description Y_i is consistent with the input table T_i. We can filter out incorrect samples to refine our generated CoT data. The verification and filtering is crucial as the high quality training data should improve performance. Finally, we get 16,858 validated examples as the training data.

3.3. Fine-tuning Small Models

Once we obtain the generated table-based reasoning data, we use them to fine-tune smaller models and inject the reasoning ability into them. As for the choice of smaller models, we select T5 (Raffel et al., 2019) and Flan-T5 (Chung et al., 2022). This is because recent works (Fu et al., 2023; Zhu et al., 2023; Magister et al., 2023) have revealed that these models can attain a remarkable numerical reasoning ability when trained with CoT data in the task of complex mathematical problem solving. We fine-tune the smaller model with the generated table-based reasoning data. Specifically, we concatenate the table T with table-based reasoning R, which are split by an added special token "<*CoT*>". The resulting input sequence takes the following form: "T <*CoT*> R". We provide an example in Figure 3. Therefore, the description Y is generated based on both the input serialised table T and table-based reasoning R with the following loss function:

$$\mathcal{L} = -\frac{1}{N}\sum_{n=1}^{N} \log P(Y \mid T, R) \quad (3)$$

where N denotes the size of the training data, and $\mathcal{L}$ is the cross entropy loss.

4. Experiments

4.1. Dataset

We conduct scientific table-to-text generation on the **SciGen** dataset (Moosavi et al., 2021a). The statistics of the data are shown in Table 1. It consists of three different settings: few-shot, medium and large. The train/val/test sets of medium setting are split into sizes of 13,607/3,452/1,038. The large setting is split into 39,969/12,129/1,038. We choose the medium and large settings to conduct the experiments. This is because the few-shot setting only contains 200 examples of training data and is insufficient for fine-tuning.

4.2. Baselines

We follow Moosavi et al. (2021a) and select T5 (Raffel et al., 2019) and BART (Lewis et al., 2020) as the student model baselines. For the BART baseline, we use BART-large with 0.40B parameters. For the T5 model, we use T5-base and T5-large with 0.22B, and 0.77B parameters, respectively. For the teacher models, we choose `text-davinci-002` and `gpt-3.5-turbo` as the baseline. For the one-shot prompt setting, we follow previous works (Chen, 2023; Zhao et al., 2023b), which prepend one demonstration example to the input table. We compare with two variants of the teacher models, called *1-shot direct* and *1-shot CoT*. For the prompt formulation of 1-shot direct, we follow the setting of Moosavi et al. (2021a) to linearise the table and concatenate it with the gold description as a demonstration. As for the prompt of 1-shot CoT, we prepend the input table to two handcrafted table-based reasonings and descriptions.

Setting	Text	Train	Val	Test
Few-shot	116	200	100	1,038
Medium	124	13,607	3,452	1,038
Large	133	39,969	12,129	1,038

Table 1: SciGen dataset statistics. Text indicates the average length in words of descriptions.

4.3. Experimental Settings

To use the above text-to-text generation baselines, we follow the setting in Moosavi et al. (2021a) and convert tables into the text sequences. To preserve and help the model better learn the table structure, we add four special tokens to specify the beginning of rows, cells, table captions, and CoT reasoning with tokens "<*R*>", "<*C*>", "<*CAP*>", "<*CoT*>", respectively. Figure 3 shows an original table from a scientific paper (Nam et al., 2019) and its corresponding linearised input representation. The generated reasoning and description from LLMs are also provided.

4.4. Automatic Evaluation Metric

We utilise a wide range of automatic evaluation metrics from various levels to assess the performance of the model.

Surface-level. Following Moosavi et al. (2021a), we choose **METEOR** (Banerjee and Lavie, 2005), **BERTScore** (Zhang et al., 2020), and **BLEURT** (Sellam et al., 2020) to measure the surface similarity of the generated statements to the gold references.

METEOR aligns the output text with the reference text and computes sentence-level similarity scores based on the alignments.

BERTScore employs BERT embeddings, which aligns words in both the generated and reference sentences using cosine similarity. It calculates precision, recall, and F1 scores.

BLEURT is a learned evaluation metric based on BERT. It is first pre-trained on synthetic examples and then fine-tuned on human judgments for the task of machine translation.

However, Moosavi et al. (2021a) stated that these metrics are not sufficient as the value range is quite low (except for BERTScore). In addition, in some cases, the incorrect description scores higher than the correct ones.

Faithfulness-level. Recent works (Moosavi et al., 2021a; Liu et al., 2022a) have pointed out that the above surface-level metrics cannot measure the factual correctness of the generated descriptions given the corresponding tables. The SciGen task requires the model to generate statements which contain numerical reasoning over table values. In addition, the generated statements might cover a different table region from the gold reference. Therefore, we add two faithfulness-level metrics (to assess whether the generated sentence is grounded in the input table), **TAPAS-Acc** and **TAPEX-Acc** (Liu et al., 2022a) to evaluate the factual consistency and fidelity, which have been widely used for table-to-text evaluation.

TAPAS-Acc fine-tunes TAPAS (Herzig et al., 2020) on the TabFact dataset (Chen et al., 2020b) and achieves 81% test accuracy.

TAPEX-Acc use TAPEX (Liu et al., 2022b) which is fine-tuned on the TabFact dataset and achieves 84% test accuracy. Previous works (Liu et al., 2022a; Zhao et al., 2023b) stated that TAPAS-Acc is overly positive about the predictions, while TAPEX-Acc is more reliable for the evaluation of the faithfulness of generated sentences.

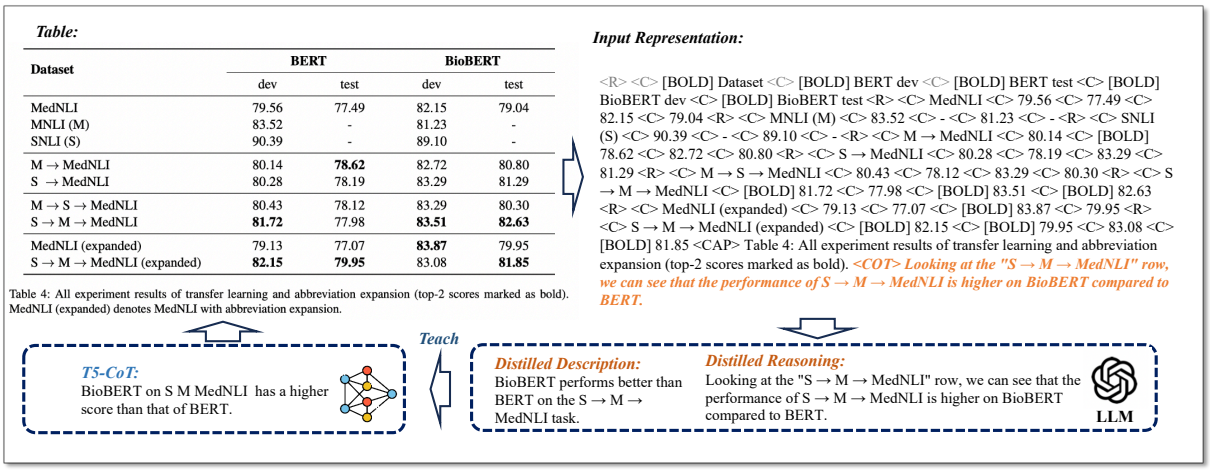

Figure 3: Sample table from Nam et al. (2019) with its corresponding input representation. The reasoning and description are generated from LLMs for further fine-tuning smaller models.

Models	#Params	Faithfulness-level		Surface-level		
		TAPAS-Acc	TAPEX-Acc	Meteor	BERTScore	BLEURT
Teacher Model						
`text-davinci-002 (1-shot direct)`	175B	66.43	64.84	0.08	**0.82**	-0.97
`gpt-3.5-turbo (1-shot direct)`	175B	72.34	70.48	0.09	**0.85**	-0.91
`text-davinci-002 (1-shot CoT)`	175B	75.35	77.89	0.09	0.82	-0.94
`gpt-3.5-turbo (1-shot CoT)`	175B	82.53	84.99	0.09	0.83	-0.96
Medium Setting						
BART-large	0.40B	57.45	58.41	**0.23**	0.84	**-0.72**
T5-base	0.22B	53.27	52.45	0.15	0.82	-0.89
T5-large	0.77B	56.32	54.78	0.17	0.83	-0.77
Flan-T5-base	0.22B	54.78	56.25	0.16	0.84	-0.82
Flan-T5-large	0.77B	58.91	57.29	0.18	0.84	-0.80
Large Setting						
BART-large	0.40B	59.69	61.38	0.15	0.82	-0.89
T5-base	0.22B	55.32	53.76	0.15	0.82	-0.85
T5-large	0.77B	58.21	56.32	0.18	0.83	-0.79
Flan-T5-base	0.22B	56.41	55.37	0.16	0.82	-0.86
Flan-T5-large	0.77B	59.81	58.34	0.17	0.83	-0.83
CoT fine tuning						
T5-base-CoT	0.22B	78.16	82.30	0.08	0.83	-0.89
T5-large-CoT	0.77B	**80.62**	81.97	0.07	0.82	-0.89
Flan-T5-base-CoT	0.22B	78.72	**82.75**	0.08	0.82	-0.89
Flan-T5-large-CoT	0.77B	79.05	82.53	0.06	0.83	-0.89

Table 2: Performance on the SciGen test set. Medium and large settings denote the setting of the datasets used for training. For the teacher model, *direct* refers to direct prompt without CoT. *CoT fine tuning* refers to fine-tuning smaller models with generated CoT data from teacher models.

Both above reference-free metrics score the generated descriptions as 0 for refuted and 1 for entailed given the corresponding tables.

5. Results

In this section, we evaluate both the performance of teacher LLMs and the fine-tuned smaller models on the scientific table-to-text task. We conduct automatic evaluation on both Surface-level and Faithfulness-level metrics. The overall results are shown in Table 2. The comparison of Faithfulness-level metrics between teacher models and student models the large are presented in the Figure 5 and Figure 6.

5.1. Performance of LLMs

Our experiments include two in-context learning methods, *Direct Prompt* and *CoT Prompt*. We select `text-davinci-002` and `gpt-3.5-turbo` to conduct experiments on the SciGen dataset. As shown in Table 2, on surface-level metrics,

Default	TAPAS-Acc	TAPEX-Acc	CoT (Ours)	TAPAS-Acc	TAPEX-Acc
T5-base	55.32	53.76	T5-base	78.16	82.30
Flan-T5-base	56.41	55.37	Flan-T5-base	78.72	82.75
T5-large	58.21	56.32	T5-large	80.62	81.97
Flan-T5-large	59.81	58.34	Flan-T5-large	79.05	82.53

Table 3: Smaller model performance on the test set of the SciGen dataset. Models fine-tuned with CoT data generally perform better than the traditional fine-tuned ones (with a minimum of 20% improvement).

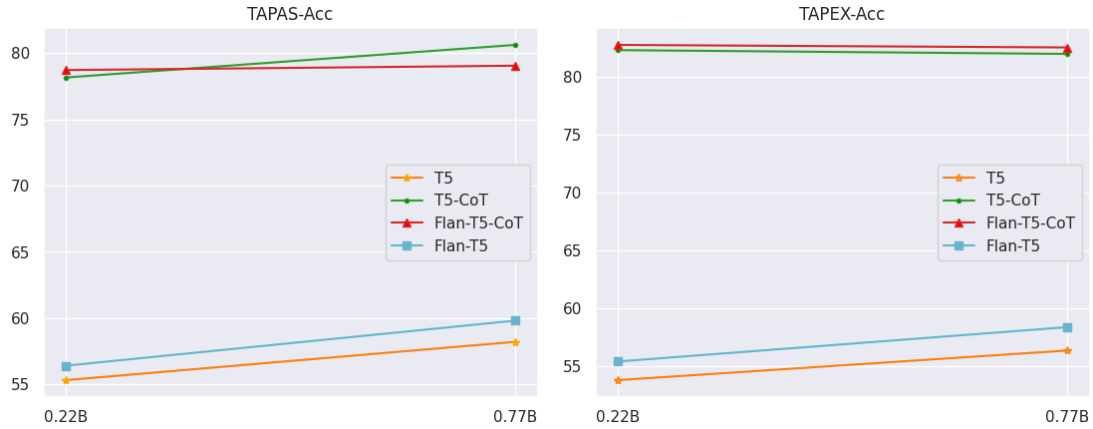

Figure 4: Ablation study of smaller models on the SciGen dataset. Compared with models using standard fine-tuning, T5 and Flan-T5 fine-tuned with CoT data achieve significant improvements on both TAPAS-Acc and TAPEX-Acc.

both *Direct Prompt* and *CoT Prompt* cannot achieve the best performance, except for `gpt-3.5-turbo (1-shot direct)` achieving the best performance on BERTScore. However, the surface-level metrics are unable to accurately measure the faithfulness and accuracy of the models' generated outputs. In terms of the faithfulness-level metrics, `text-davinci-002 (1-shot direct)` can achieve over 64% accuracy and `gpt-3.5-turbo (1-shot direct)` can achieve over 70% accuracy on both TAPAS-Acc and TAPEX-Acc, which outperform the traditional fine-tuned baseline models (i.e. BART and T5). When combined the direct prompt with CoT reasoning, the accuracy of both `text-davinci-002 (1-shot CoT)` and `gpt-3.5-turbo (1-shot CoT)` increases by around 10% on both metrics.

5.2. Performance of Fine-tuned Smaller Model

Regarding the surface-level metrics, the smaller models, whether fine-tuned with CoT data or not, consistently exhibit a narrow range of low values, with absolute values falling within the 0-1 Likert scale range. The experimental results are consistent with the statements in SciGen's paper (Moosavi et al., 2021a) that surface-level metrics are not sufficient to reflect models' abilities on this complex task.

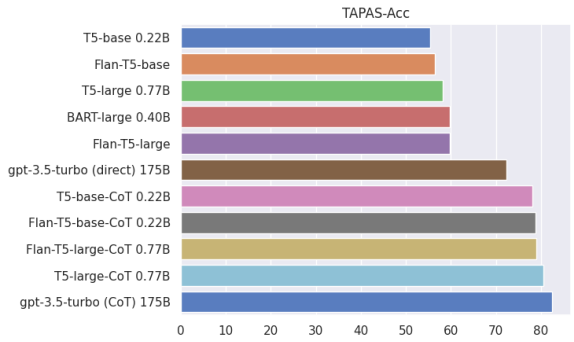

Figure 5: The TAPAS-Acc of the teacher models (LLMs) and small models on the SciGen dataset. All the small models fine-tuned with CoT data can surpass LLMs with direct prompting.

Small models with traditional fine-tuning do not perform well on faithfulness-level metrics. In terms of the smaller models fine-tuned without CoT data, BART-large fine-tuned on the medium dataset achieves the best on surface-level metrics. However, in terms of the faithfulness-level, all the BART and T5 baselines only achieve an accuracy slightly higher than ran-

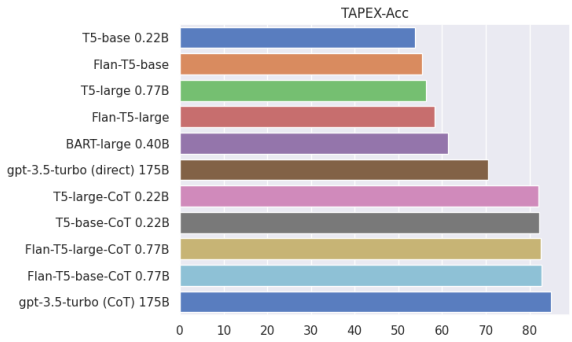

Figure 6: The TAPEX-Acc of teacher models and small models on the SciGen dataset. The trend is similar to TAPAS-Acc, with the performance of small models fine-tuned with CoT data only underperforming when compared to LLMs with CoT prompting.

dom chance. We further investigate the impact of dataset size, ranging from the *Medium Setting* to the *Large Setting*. Although the size of the Large Setting dataset is three times that of the Medium Setting, performance improvements are not as significant (i.e., only around 2% increase on the faithfulness-level metrics). However, for the surface-level metrics, models that are trained with the Medium datasets achieve better overall performance, especially in METEOR and BLEURT.

Small models fine-tuned with CoT data achieve a significant performance improvement. On the other hand, the T5 and Flan-T5 models with CoT fine-tuning can achieve the best overall performance on the faithfulness-level metrics among all the small models. All the performances of CoT fine-tuning models are on par with the teacher model (i.e., `gpt-3.5-turbo (1-shot CoT)` on the faithfulness-level metrics. For instance, T5-large-CoT and Flan-T5-base-CoT achieve the highest TAPAS-Acc (80.62%) and TAPEX-Acc (82.75%), and only underperform the teacher model with the best performance by a margin of 2%. These results indicate that fine-tuning with CoT data distilled from LLMs can transfer the table-based reasoning ability into smaller models.

Larger model size does not guarantee the performance improvement when fine-tuned without CoT data. Furthermore, our experiments also investigate the impact of the model size for CoT fine-tuning, ranging from the base to the large variant. While it is intuitive to expect performance improvements with larger models, the experimental results on TAPEX-Acc metric reveal that models with larger parameter counts, such as T5-large and Flan-T5-large, do not consistently outperform their smaller counterparts, T5-base and Flan-T5-base. However, regarding TAPAS-Acc, the performance improvement is consistent, with the model size increasing from base (0.22B) to large (0.77B).

5.3. Comparison between Teacher and Student Models

We also compare the performance on faithfulness-level metrics (TAPAS-Acc and TAPEX-Acc) of both the teacher model (LLMs) and student models in Figure 5 and Figure 6. For the teacher model, `gpt-3.5-turbo (1-shot direct)` outperforms all smaller baseline models (smaller models fine-tuned without CoT data) and `text-davinci-002 (1-shot direct)`. In addition, `gpt-3.5-turbo (1-shot CoT)` achieves the best performances on both TAPAS-Acc and TAPEX-Acc metrics among both teacher and student models. As for smaller models, both T5 and Flan-T5 can only achieve around 55% accuracy on both faithfulness-level metrics without being fine-tuned with CoT data. However, these smaller models can be injected with reasoning ability after fine-tuning with CoT data, achieving approximately 80% accuracy on both metrics.

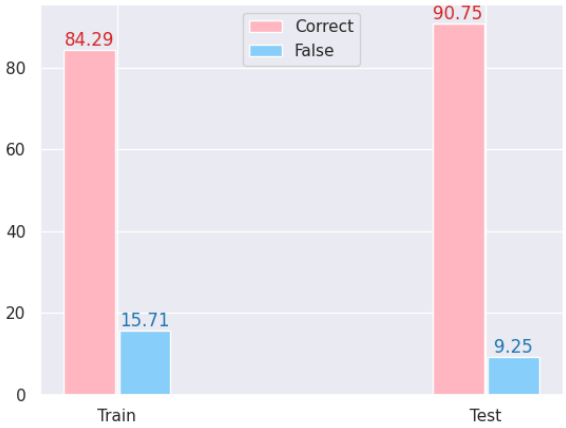

Figure 7: Evaluation of generated data of train and test sets of SciGen dataset. Correct refers to the data with statements verified correctly by LLMs.

5.4. Ablation Study

The ablation study of fine-tuning with CoT data are shown in both Table 3 and Figure 4. For both T5 and Flan-T5 models, we can observe the significant increases after fine-tuning with with CoT data in both TAPAS-Acc and TAPEX-Acc on the SciGen table-to-text generation task. For TAPAS-Acc metric, T5 and Flan-T5 base (0.22B) and large (0.77B) models can only achieve over 55% accuracy. However, when fine-tuning with table-based CoT data from LLMs, there is a significant accuracy increase (over 20%) observed. For instance, the 55% accuracy of T5-large with standard fine-

tuning can be improved to 80% after being fine-tuned with CoT data. As for TAPEX-Acc metric, a similar trend can be observed, where the overall improvement in accuracy is over 25%. For example, the most significant improvement can be observed in the T5-base model, which is from 53% (traditional fine-tune) to 82% (CoT fine-tune).

5.5. Generated Data Analysis

The LLMs we used in this paper contributed towards the synthesis of high-quality table-based CoT data. However, during the generation process, there are certain falsely generated data due to the hallucinatory nature of LLMs. Therefore, we conduct a comprehensive analysis of the samples generated by LLMs. The evaluation results are shown in Figure 7, `gpt-3.5-turbo` achieves an accuracy of 85% on the training set, where the generated descriptions are verified as correct. As for the test set of SciGen, the accuracy is over 90%, and with less than 10% of the samples regarded as incorrect. Regarding the table-to-text generation task, both the generated reasoning and descriptions reveal high-quality coherence and consistency given the input table.

6. Conclusion

In this paper, we introduce a two-stage distillation framework that distills table-based CoT data from LLMs. Our experiments illustrate that this method is able to effectively transfer table reasoning abilities to smaller models in the scientific table-to-text generation task. The performance improvement can even outperform certain teacher LLMs (e.g., `gpt-3.5-turbo`). Our proposed method achieves comprehensive superiority in this specific task while requiring less data and smaller models.

Bibliographical References

Rami Aly, Zhijiang Guo, M. Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. Feverous: Fact extraction and verification over unstructured and structured information. *ArXiv*, abs/2106.05707.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *IEEvaluation@ACL*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *ArXiv*, abs/2005.14165.

Zefeng Cai, Xiangyu Li, Binyuan Hui, Min Yang, Bowen Li, Binhua Li, Zhen Cao, Weijie Li, Fei Huang, Luo Si, and Yongbin Li. 2022. Star: Sql guided pre-training for context-dependent text-to-sql parsing. *ArXiv*, abs/2210.11888.

Wenhu Chen. 2023. Large Language Models are few(1)-shot Table Reasoners. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1120–1130, Dubrovnik, Croatia. Association for Computational Linguistics.

Wenhu Chen, Jianshu Chen, Yu Su, Zhiyu Chen, and William Yang Wang. 2020a. Logical Natural Language Generation from Open-Domain Tables. ArXiv:2004.10404 [cs].

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020b. TABFACT: A LARGE-SCALE DATASET FOR TABLE- BASED FACT VERIFICATION.

Zhiyu Chen, Wenhu Chen, Hanwen Zha, Xiyou Zhou, Yunkai Zhang, Sairam Sundaresan, and William Yang Wang. 2020c. Logic2Text: High-Fidelity Natural Language Generation from Logical Forms. *arXiv:2004.14579 [cs]*. ArXiv: 2004.14579.

Zhoujun Cheng, Tianbao Xie, Peng Shi, Chengzu Li, Rahul Nadkarni, Yushi Hu, Caiming Xiong, Dragomir Radev, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. 2023. Binding Language Models in Symbolic Languages. ArXiv:2210.02875 [cs].

Minseok Cho, Gyeongbok Lee, and Seung won Hwang. 2019. Explanatory and actionable debugging for machine learning: A tableqa demonstration. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam

Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam M. Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Benton C. Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier García, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Díaz, Orhan Firat, Michele Catasta, Jason Wei, Kathleen S. Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways. *ArXiv*, abs/2204.02311.

Hyung Won Chung, Le Hou, S. Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Wei Yu, Vincent Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed Huai hsin Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *ArXiv*, abs/2210.11416.

Andrew Drozdov, Jiawei Zhou, Radu Florian, Andrew McCallum, Tahira Naseem, Yoon Kim, and Ramon Fernandez Astudillo. 2022. Inducing and Using Alignments for Transition-based AMR Parsing. ArXiv:2205.01464 [cs].

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023. Specializing Smaller Language Models towards Multi-Step Reasoning. ArXiv:2301.12726 [cs].

Carlos Gemmell and Jeffrey Stephen Dalton. 2023. Generate, transform, answer: Question specific tool synthesis for tabular data. *ArXiv*, abs/2303.10138.

Tomas Goldsack, Zhihao Zhang, Chen Tang, Carolina Scarton, and Chenghua Lin. 2023. Enhancing biomedical lay summarisation with external knowledge graphs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8016–8032, Singapore. Association for Computational Linguistics.

Vivek Gupta, Maitrey Mehta, Pegah Nokhiz, and Vivek Srikumar. 2020. Infotabs: Inference on tables as semi-structured data. In *Annual Meeting of the Association for Computational Linguistics*.

Jonathan Herzig, Paweł Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Martin Eisenschlos. 2020. TAPAS: Weakly Supervised Table Parsing via Pre-training. *arXiv:2004.02349 [cs]*. ArXiv: 2004.02349.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large Language Models Are Reasoning Teachers. ArXiv:2212.10071 [cs].

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander J. Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *ArXiv*, abs/2305.02301.

Henglin Huang, Chen Tang, Tyler Loakman, Frank Guerin, and Chenghua Lin. 2022. Improving Chinese story generation via awareness of syntactic dependencies and semantics. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 178–185, Online only. Association for Computational Linguistics.

Zhengbao Jiang, Yi Mao, Pengcheng He, Graham Neubig, and Weizhu Chen. 2022. Omnitab: Pretraining with natural and synthetic data for few-shot table-based question answering. In *North American Chapter of the Association for Computational Linguistics*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *ArXiv*, abs/2205.11916.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*,

pages 7871–7880, Online. Association for Computational Linguistics.

Ao Liu, Haoyu Dong, Naoaki Okazaki, Shi Han, and Dongmei Zhang. 2022a. PLOG: Table-to-Logic Pretraining for Logical Table-to-Text Generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5531–5546, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2022b. TAPEX: TABLE PRE-TRAINING VIA LEARNING A NEURAL SQL EXECUTOR.

Tianyu Liu, Xin Zheng, Baobao Chang, and Zhifang Sui. 2021. Towards Faithfulness in Open Domain Table-to-text Generation from an Entity-centric View. *arXiv:2102.08585 [cs]*. ArXiv: 2102.08585.

Tyler Loakman, Chen Tang, and Chenghua Lin. 2023. TwistList: Resources and baselines for tongue twister generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 579–589, Toronto, Canada. Association for Computational Linguistics.

Xinyuan Lu, Liangming Pan, Qian Liu, Preslav Nakov, and Min-Yen Kan. 2023. SCITAB: A Challenging Benchmark for Compositional Reasoning and Claim Verification on Scientific Tables. ArXiv:2305.13186 [cs].

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching Small Language Models to Reason. ArXiv:2212.08410 [cs].

N. Moosavi, Andreas Rücklé, D. Roth, and Iryna Gurevych. 2021a. SciGen: a Dataset for Reasoning-Aware Text Generation from Scientific Tables.

Nafise Sadat Moosavi, Andreas Rücklé, Dan Roth, and Iryna Gurevych. 2021b. 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)*.

Jiin Nam, Seunghyun Yoon, and Kyomin Jung. 2019. Surf at mediqa 2019: Improving performance of natural language inference in the clinical domain by adopting pre-trained language model. *arXiv preprint arXiv:1906.07854*.

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.

Linyong Nan, Yilun Zhao, Weijin Zou, Narutatsu Ri, Jaesung Tae, Ellen Zhang, Arman Cohan, and Dragomir Radev. 2023. Enhancing few-shot text-to-sql capabilities of large language models: A study on prompt design strategies. *arXiv preprint arXiv:2305.12586*.

Ankur P. Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A Controlled Table-To-Text Generation Dataset. *arXiv:2004.14373 [cs]*. ArXiv: 2004.14373.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Annual Meeting of the Association for Computational Linguistics*.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *ArXiv*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *ACL*.

Yujie Shao, Xinrong Yao, Xingwei Qu, Chenghua Lin, Shi Wang, Stephen Huang, Ge Zhang, and Fu Jie. 2024. CMDAG: A chinese metaphor dataset with annotated grounds as cot for boosting metaphor generation. In *Proceedings of the Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)*.

Chen Tang, Chenghua Lin, Henglin Huang, Frank Guerin, and Zhihao Zhang. 2022. EtriCA: Event-triggered context-aware story generation aug-

mented by cross attention. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5504–5518, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chen Tang, Shun Wang, Tomas Goldsack, and Chenghua Lin. 2023a. Improving biomedical abstractive summarisation with knowledge aggregation from citation papers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 606–618, Singapore. Association for Computational Linguistics.

Chen Tang, Hongbo Zhang, Tyler Loakman, Chenghua Lin, and Frank Guerin. 2023b. Enhancing dialogue generation via dynamic graph knowledge aggregation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4604–4616, Toronto, Canada. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. ArXiv:2203.11171 [cs].

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Huai hsin Chi, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *ArXiv*, abs/2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903.

Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in Data-to-Document Generation. *arXiv:1707.08052 [cs]*. ArXiv: 1707.08052.

Bohao Yang, Chen Tang, and Chenghua Lin. 2023. Improving medical dialogue generation with abstract meaning representations. *arXiv preprint arXiv:2309.10608*.

Yunhu Ye, Binyuan Hui, Min Yang, Binhua Li, Fei Huang, and Yongbin Li. 2023. Large Language Models are Versatile Decomposers: Decompose Evidence and Questions for Table-based Reasoning. ArXiv:2301.13808 [cs].

Tao Yu, Rui Zhang, Kai-Chou Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Z Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir R. Radev. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. *ArXiv*, abs/1809.08887.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, P. Zhang, Yuxiao Dong, and Jie Tang. 2022. Glm-130b: An open bilingual pre-trained model. *ArXiv*, abs/2210.02414.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *ArXiv*, abs/1904.09675.

Kun Zhao, Bohao Yang, Chenghua Lin, Wenge Rong, Aline Villavicencio, and Xiaohui Cui. 2023a. Evaluating open-domain dialogues in latent space with next sentence prediction and mutual information. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 562–574, Toronto, Canada.

Yilun Zhao, Linyong Nan, Zhenting Qi, Rui Zhang, and Dragomir R. Radev. 2022. Reastap: Injecting table reasoning skills during pre-training via synthetic reasoning examples. In *Conference on Empirical Methods in Natural Language Processing*.

Yilun Zhao, Haowei Zhang, Shengyun Si, Linyong Nan, Xiangru Tang, and Arman Cohan. 2023b. Large Language Models are Effective Table-to-Text Generators, Evaluators, and Feedback Providers. ArXiv:2305.14987 [cs].

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

Denny Zhou, Nathanael Scharli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. 2023. LEAST-TO-MOST PROMPTING ENABLES COMPLEX REASONING IN LARGE LANGUAGE MODELS.

Xuekai Zhu, Biqing Qi, Kaiyan Zhang, Xingwei Long, and Bowen Zhou. 2023. PaD: Program-aided Distillation Specializes Large Models in Reasoning. ArXiv:2305.13888 [cs].