Does Table Source Matter? Benchmarking and Improving Multimodal Scientific Table Understanding and Reasoning

Anonymous ACL submission

Abstract

Recent large language models (LLMs) have advanced table understanding capabilities but rely on converting tables into text sequences. While multimodal large language models (MLLMs) enable direct visual processing, they face limitations in handling scientific tables due to fixed input image resolutions and insufficient numerical reasoning capabilities. To address these challenges, we present MMSci, a comprehensive dataset for scientific table understanding and reasoning. MMSci consists of three key components: (1) MMSci-Pre, a domain-specific dataset of 52K scientific table structure recognition samples, (2) MMSci-Ins, an instruction tuning dataset with 12K samples across three table-based tasks, and (3) MMSci-Eval, a benchmark with 3,114 testing samples specifically designed to evaluate numerical reasoning capabilities. Based on MMSci, we develop a table-based MLLM framework with dynamic input image resolutions. Extensive experiments demonstrate that our domain-specific approach with 52K scientific table images achieves superior performance compared to 150K general-domain tables, highlighting the importance of data quality over quantity. Our proposed framework shows significant improvements in both general table understanding and numerical reasoning capabilities, with strong generalisation to held-out datasets. Our code and data are publicly available at https://anonymous.4open.science/r/MMSci_Table-F278/.

1 Introduction

Tables serve as a fundamental tool for organising structured information across diverse domains. Recent studies have shown the potential of leveraging large language models (LLMs) to automatically understand and process tabular data, which has emerged as a critical research direction with applications such as Table Question Answering (TQA) (Pasupat and Liang, 2015), Table Fact Verification (TFV) (Chen et al., 2020a), and Table-to-Text Generation (T2T) (Moosavi et al., 2021).

However, current table-oriented LLMs (Zhang et al., 2023; Li et al., 2023b) face inherent limitations as they require converting tables into sequential text formats (i.e., HTML strings), potentially losing crucial structural and positional information. While table-based multimodal large language models (MLLMs) have addressed this by enabling direct processing of table images, several critical limitations persist: (1) fixed input image resolutions that constrain practical applicability, (2) limited capability in processing scientific tables that contain significant numerical values, and (3) insufficient numerical reasoning abilities for scientific domain tasks. These limitations are particularly significant in scientific domains, where tables frequently incorporate complex numerical relationships. While current MLLMs have demonstrated efficacy with general-domain tables (e.g., tables from Wikipedia or reports), they exhibit substantial performance degradation when confronted with scientific tables that mostly contains numerical values. Processing such tables effectively requires not only visual understanding to capture structural elements but also advanced numerical reasoning capabilities to interpret and analyse complex numerical relationships, statistical significance, and experimental results. Current MLLMs, however, lack the specific architectural designs and domain-specific training data to handle these sophisticated scientific table understanding requirements.

To address these challenges, we introduce MMSci, a comprehensive dataset for scientific table understanding and reasoning. We first conduct a systematic analysis of table source effectiveness through MMSci-Pre, a carefully curated dataset containing 52K structure recognition samples derived from scientific papers. Our experimental results demonstrate that MLLMs trained on these domain-specific table images significantly outper-

form those trained on 150K general-domain tables, establishing the importance of data quality over quantity in table understanding tasks.

Building upon this foundation, we then create MMSci-Ins, an instruction tuning dataset comprising 12K samples with explicit intermediate reasoning steps across three fundamental tasks: TQA, TFV, and T2T. Each sample includes detailed step-by-step reasoning processes to develop models' table-based numerical reasoning and scientific analysis capabilities. To overcome the limitations of fixed-resolution approaches in existing table MLLMs (Lee et al., 2023; Alonso et al., 2024; Zheng et al., 2024), we implement our MLLM with dynamic input image resolution capabilities on two distinct model architectures (Qwen2-VL-7B-Instruct and LLaVA-NeXT-7B). Experimental results demonstrate consistent performance improvements across both general table understanding and specialised numerical reasoning tasks.

To enable comprehensive evaluation, we establish MMSci-Eval, a benchmark with 3,114 testing samples requiring numerical reasoning capabilities. The benchmark provides rigorous assessment of models' performance across TQA, TFV, and T2T tasks. Our extensive experiments demonstrate that our 52K scientific table images prove more effective than 150K general-domain table images for both general understanding and numerical reasoning tasks. This efficiency highlights the value of domain-specific, high-quality data in developing robust table understanding capabilities.

Our contributions are summarised as follows:

- We introduce MMSci, a comprehensive dataset consisting of three components: (1) MMSci-Pre, consists of 52K table image-to-HTML table structure recognition samples; (2) MMSci-Ins, an instruction tuning dataset of 12K samples with reasoning steps; and (3) MMSci-Eval, a benchmark with 3,114 samples for numerical reasoning capabilities assessment across TQA, TFV, and T2T tasks.

- We develop a comprehensive table-based MLLM framework that achieves strong performance on three table-based numerical reasoning tasks while demonstrating robust generalisation to held-out datasets.

- We implement dynamic input resolution capabilities across different model architectures, validating the effectiveness of our approach through consistent performance gains on both Qwen2-VL-7B-Instruct and LLaVA-NeXT-7B.

2 Related Work

2.1 Table Understanding Models

Early table-based models based on general language models with large-scale table corpus (Liu et al., 2022; Chen et al., 2023) only support limited types of tables and tasks. Table understanding capabilities have been enhanced through prompt engineering (Chen, 2023; Sui et al., 2023), instruction tuning (Zhang et al., 2023; Li et al., 2023b; Yang et al., 2024b) and external tools (Lu et al., 2023a; Li et al., 2023a) with the development of LLMs. However, these approaches require converting tables into text formats, limiting their applications.

Recently, MLLMs have emerged as a promising direction for table understanding. TableGPT2 (Su et al., 2024a) features a novel table encoder to handle table cell-level information. Pix2Struct (Lee et al., 2023) introduces a unified image-to-text model pretrained on web page screenshots with HTML supervision. PixT3 (Alonso et al., 2024) takes table-to-text tasks as table visual recognition tasks and generates texts. Table-LLaVA (Zheng et al., 2024) introduces a novel multimodal table understanding approach that directly processes table images. However, these approaches do not focus on datasets requiring sophisticated numerical reasoning capabilities.

2.2 Table-based Reasoning and Datasets

Table-based reasoning requires reasoning over both free-form natural language queries and structured tables. Early works either rely on executable languages (*e.g.*, SQL) (Yin et al., 2016; Yu et al., 2018) to capture logical structure in statements. TAPAS (Herzig et al., 2020), and DATER (Ye et al., 2023) encode sentence-table pairs and transform table-based reasoning into question-answering or inference tasks. Existing datasets primarily focus on specific domains like Wikipedia and finance. HybridQA (Chen et al., 2020b) derived from Wikipedia emphasises span lookup, while TAT-QA (Zhu et al., 2021), FinQA (Chen et al., 2021), and DocMath-Eval (Zhao et al., 2024) address numerical reasoning in the financial domain. SciGen (Moosavi et al., 2021) introduces a scientific table-to-text generation dataset that requires

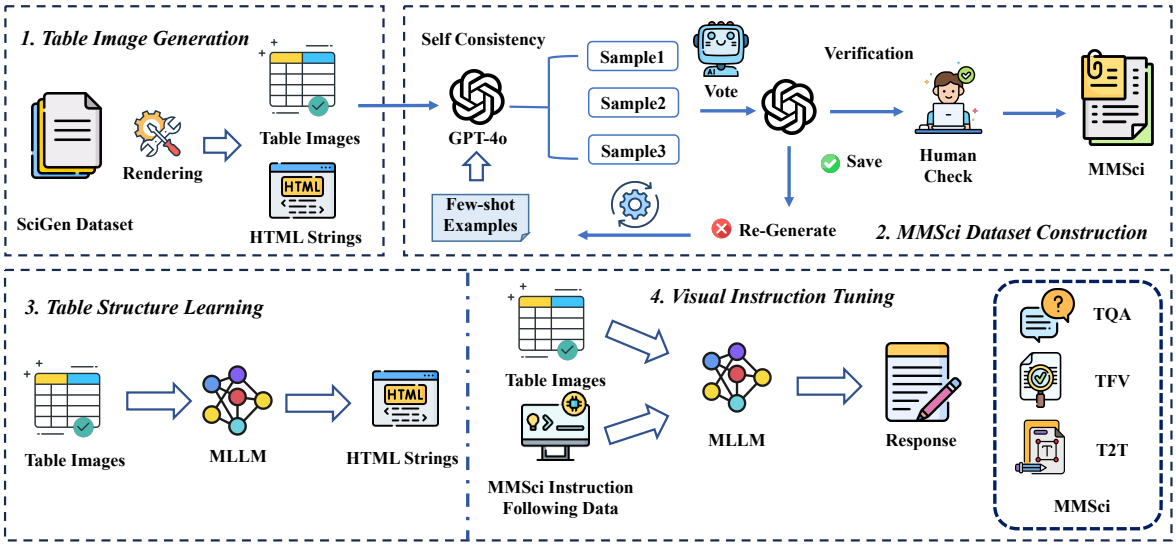

Figure 1: Overview of the proposed framework, which consists of four key stages: (1) Table Image Generation; (2) MMSci Dataset Construction; (3) Table Structure Learning; and (4) Visual Instruction Tuning.

arithmetic reasoning, but focuses mainly on generation rather than comprehensive reasoning evaluation. However, these datasets have relatively limited reasoning types, significantly differing from real-world scientific table understanding that require numerical computation reasoning. To address this gap, we propose MMSci dataset that combines multiple reasoning types to enhance model performance on complex scientific table understanding tasks.

3 Construction of MMSci Dataset

As shown in Figure 1, our framework consists of four key stages: (1) Table Image Generation: converting SciGen dataset tables into high-quality HTML and image formats; (2) MMSci Dataset Construction: employing GPT-4o (OpenAI, 2024) with self-consistency voting (Wang et al., 2023) and human verification to generate high-quality samples; (3) Table Structure Learning: training MLLMs to generate HTML representations from table images; and (4) Visual Instruction Tuning: fine-tuning MLLMs on instruction-following data across TQA, TFV, and T2T tasks.

3.1 Data Collection

To construct MMSci dataset, we focus on scientific tables that contain significant numerical values and complex reasoning requirements. We collect raw tabular data from the SciGen dataset (Moosavi et al., 2021), which provides pairs of scientific tables and their corresponding descriptions across computer science research domains. A large number of these descriptions require arithmetic reasoning over table values, indicating the natural presence of numerical reasoning in scientific table descriptions. We transform the original textual tables into high-quality HTML format and then render them into table images while preserving their structural integrity. This process ensures that the visual representation maintains the complex layouts and relationships present in the original scientific tables. Finally, we collect 52K image-to-HTML pairs based on tables from the training set and development set of the SciGen dataset.

3.2 Table Structure Learning

Existing table-based MLLMs (Lee et al., 2023; Alonso et al., 2024; Zheng et al., 2024) demonstrate that generating textual table representations given the table image is crucial for aligning table structure and text information within the table image. Therefore, we create 52K instruction-following image-to-HTML samples based on the tables from the training set of SciGen dataset via the Imgkit[1] python package. The resulting pre-training data contains 52K table image-to-HTML examples, which is denoted as MMSci-Pre dataset.

3.3 Numerical Reasoning Augmentation

For the construction of instruction dataset MMSci-Ins and MMSci-Eval, we select 12,000 tables from the training set and 1,038 from the testing set of

[1] https://pypi.org/project/imgkit/

SciGen dataset to create MMSci-Ins and MMSci-Eval datasets, respectively. For each table, we employ GPT-4o to generate task-specific content by feeding the table image and its corresponding descriptions in SciGen dataset. For TQA tasks, we generate questions paired with corresponding reasoning steps and answers. For TFV tasks, we create claims along with supporting reasoning steps and verification results. The labels of claim including three types: supported, refuted, and not enough information, which is consistent with the existing TFV datasets (Lu et al., 2023c). Additionally, we augment the existing table-to-text pairs in SciGen dataset with detailed reasoning steps for T2T tasks.

To ensure the quality and consistency of the generated content, we employ the self-consistency Chain-of-Thought (CoT) reasoning mechanism (Wang et al., 2023). For each task, we generate multiple reasoning paths and employ a voting mechanism to determine the final output. To validate the quality of these generated samples, we employ a two-stage verification process: first using GPT-4o to assess the consistency between reasoning steps and outputs, then manually assess 40% of the generated samples to further ensure high quality. The identified false samples are regenerated using GPT-4o to maintain dataset quality. Through this rigorous construction and verification process, we create the MMSci-Ins dataset comprising 12K instruction-tuning samples based on the training set of SciGen dataset, and the MMSci-Eval benchmark with 3,114 testing examples based on the testing set of SciGen dataset. Each sample includes detailed step-by-step reasoning processes, enabling models to learn both the final outputs and the logical progression needed to arrive at those conclusions. The dataset maintains a balanced distribution across the three tasks, where each table is paired with one TQA sample, one TFV sample, and one T2T sample, ensuring comprehensive coverage of different reasoning requirements in scientific table understanding. Detailed dataset quality control could be found in the Appendix A.2.

4 Experiments

4.1 Model Training

To demonstrate the effectiveness of MMSci dataset, we train two series of MLLM following the architecture of Qwen2-VL-7B-Instruct (Wang et al., 2024) and LLaVA-NeXT-7B (Li et al., 2024).

Model Architectures. Both models follow a three-component design: **Qwen2-VL-7B-Instruct** consists of a Vision Transformer (ViT) (Dosovitskiy, 2020) as the vision tower, a MLP as the vision-language connector, and a Qwen2-7B-Instruct (Yang et al., 2024a) as the language model. **LLaVA-NeXT-7B** uses a pre-trained CLIP model (Radford et al., 2021) as the visual encoder, a MLP connector, and a Vicuna-7B model (Chiang et al., 2023) as the backbone. In both architectures, the vision encoder processes images into visual features, which are projected into the LLM's word embedding space via the MLP connector.

We divide the training into two stages:

Table Structure Learning. We use both MMSci-Pre and MMTab-Pre (Zheng et al., 2024) corpus (202K table image-to-HTML pairs in total) to align visual features with textual representations in different experimental settings as shown in Table 1. Models learn to generate HTML table representations, developing table structure perception capabilities. For LLaVA-NeXT-7B, only the MLP connector parameters are updated during this stage.

Visual Instruction Tuning. We use 12K instruction-following samples from MMSci-Ins to fine-tune the MLLMs while keeping visual encoders frozen. Only the MLP projection layer and LLM weights are updated, focusing on developing instruction-following numerical reasoning capabilities across TQA, TFV, and T2T tasks.

Notably, both models support dynamic input resolutions, addressing a key limitation of existing table MLLMs (Lee et al., 2023; Alonso et al., 2024; Zheng et al., 2024) that require fixed-size input image resolutions (e.g., 336×336). Qwen2-VL achieves this through 2D-RoPE (Su et al., 2024b) to capture two-dimensional positional information of images while LLaVA-NeXT employs a simpler approach of splitting images into grids and encoding them independently.

4.2 Experimental Settings

Baselines. We select several state-of-the-art MLLMs as our baselines, including GPT-4V (OpenAI, 2023), InternVL-2-76B (Chen et al., 2024), LLaVA-NeXT series (72B/34B/13B/7B) (Li et al., 2024), Qwen-2-VL-Instruct series (72B/7B) (Wang et al., 2024), Table-LLaVA series (13B/7B) (Zheng et al., 2024), Pixtral-12B (Agrawal et al., 2024), Llama-3.2-11B-Vision-Ins. (Meta, 2024), MiniCPM-V-2.6-8B (Yao et al., 2024), and InternVL-2-8B (Chen et al., 2024).

Datasets and Metrics. The held-in evaluation

Models	MMSci-Eval			Held-out	
	TQA	TFV	T2T	TABMWP	TAT-QA
	Acc.	Acc.	BLEU	Acc.	Acc.
Baseline					
GPT-4V (OpenAI, 2023)	**53.13**	**78.01**	**4.80**	**60.00**	**32.50**
InternVL-2-76B (Chen et al., 2024)	40.31	62.46	1.79	46.28	6.73
LLaVA-NeXT-72B (Li et al., 2024)	11.75	49.28	1.79	10.69	3.29
Qwen-2-VL-72B-Ins. (Wang et al., 2024)	39.11	64.06	2.83	41.42	17.65
LLaVA-NeXT-34B (Li et al., 2024)	9.73	42.19	2.33	6.96	1.29
LLaVA-NeXT-13B (Li et al., 2024)	2.31	1.83	1.79	1.67	0.43
Table-LLaVA-13B (Zheng et al., 2024)	8.57	51.15	0.03	59.77	15.67
Pixtral-12B (Agrawal et al., 2024)	0.96	5.49	4.12	4.64	7.46
Llama-3.2-11B-Vision-Ins. (Meta, 2024)	1.15	5.85	3.04	7.39	0.37
LLaVA-NeXT-7B (Li et al., 2024)	0.19	0.86	2.99	1.73	0.72
Qwen-2-VL-7B-Ins. (Wang et al., 2024)	25.62	52.79	3.04	34.43	16.19
InternVL-2-8B (Chen et al., 2024)	25.72	44.99	2.64	18.42	7.12
MiniCPM-V-2.6-8B (Yao et al., 2024)	26.58	33.23	0.07	24.30	11.94
Table-LLaVA-7B (Zheng et al., 2024)	7.99	39.30	0.03	57.78	12.82
Ours (LLaVA-NeXT-7B)					
MMSci-Pre (52k) + MMSci-Ins	17.72	57.12	2.93	49.47	10.46
MMTab-Pre (150k) + MMSci-Ins	15.79	56.16	2.88	47.55	8.03
MM-Pre (202k) + MMSci-Ins	23.02	58.57	2.36	49.72	12.27
w/o MM-Pre (202k)	15.22	51.73	2.86	46.24	7.63
Ours (Qwen2-VL-7B-Ins.)					
MMSci-Pre (52k) + MMSci-Ins	41.13	72.92	3.24	49.50	19.68
MMTab-Pre (150k) + MMSci-Ins	40.75	72.73	3.16	49.08	19.30
MM-Pre (202k) + MMSci-Ins	42.10	73.98	3.29	49.96	20.85
w/o MM-Pre (202k)	41.71	70.90	3.29	48.02	20.07

Table 1: Performance comparison on MMSci-Eval and held-out tabular numerical reasoning datasets. MM-Pre (202k) indicates the combination of MMTab-Pre (150k) and MMSci-Pre (52k). w/o MM-Pre represents only training with MMSci-Ins dataset. Best results are in **bold**, second best are underlined.

sets in Table 1 include TQA, TFV and T2T tasks of MMSci-Eval. The held-out evaluation sets in Table 2 are from MMTab-Eval benchmark (Zheng et al., 2024). TQA contains TABMWP (Lu et al., 2023b), WTQ (Pasupat and Liang, 2015), HiTab (Cheng et al., 2022), TAT-QA (Zhu et al., 2021), and FeTaQA (Nan et al., 2022), where TABMWP and TAT-QA specifically focus on tabular numerical reasoning. TFV contains TabFact (Chen et al., 2020a) and InfoTabs (Gupta et al., 2020), while Table-to-Text (T2T) generation uses HiTab_T2T (Cheng et al., 2022), Rotowire (Wiseman et al., 2017), and WikiBIO (Lebret et al., 2016). While these datasets contain tables from Wikipedia, financial reports, and government documents, our MMSci-Eval datasets primarily feature scientific tables with numerical values from research papers. We use accuracy and BLEU (Papineni et al., 2002) for TQA, TFV, and T2T benchmarks.

5 Results and Analysis

5.1 Performance on Numerical Reasoning Datasets

The experimental results demonstrate the effectiveness of our proposed approach across various multimodal table understanding tasks. As shown in Table 1, we compare our method with state-of-the-art baselines on both MMSci benchmarks (TQA, TFV, T2T) and held-out tabular numerical reasoning datasets (TABMWP, TAT-QA). Among the baseline models, GPT-4V (OpenAI, 2023) achieves superior performance across all tasks, establishing strong benchmarks with 53.13% accuracy on TQA, 78.01% on TFV, and notably strong generalisation ability on held-out numerical reasoning datasets. Large-scale open-sourced models like InternVL-2-76B (Chen et al., 2024) and Qwen-2-VL-72B (Wang et al., 2024) also demonstrate competitive performance but show relatively weaker generalisation to held-out numerical reasoning datasets.

As for our approaches, with LLaVA-NeXT-7B as the foundation model, we observe that training with MMSci-Pre (52k) dataset demonstrates higher performance (17.72% on TQA, 57.12% on TFV) compared to training with MMTab-Pre (150k) dataset (15.79% on TQA, 56.16% on TFV). The combination of both table structure learning dataset (MM-Pre 202k) further improves performance to 23.02% on TQA and 58.57% on TFV. Notably, our approach shows strong generalisation ability on held-out datasets, achieving 49.72% on TABMWP with the experiment setting of MM-Pre (202k) + MMSci-Ins.

With Qwen2-VL-7B-Instruct as the foundation model, we observe significantly stronger performance across all settings. Training with MMSci-Pre (52k) + MMSci-Ins achieves comparable or

Method	TQA						TFV			T2T			
	TABMWP	WTQ	HiTab	TAT-QA	FeTaQA	Avg. TQA	TabFact	InfoTabs	Avg. TFV	HiTab_T2T	Rotowire	WikiBIO	Avg. T2T
	Acc.	Acc.	Acc.	Acc.	BLEU	Acc.	Acc.	Acc.	Acc.	BLEU	BLEU	BLEU	BLEU
Baseline													
GPT-4V (OpenAI, 2023)	**60.50**	**48.00**	**27.50**	**32.50**	11.04	**35.91**	45.50	65.60	55.55	2.98	4.23	1.94	3.05
Qwen2-VL-7B-Ins. (Wang et al., 2024)	34.44	12.55	3.36	16.19	11.75	15.66	20.28	34.19	27.23	1.90	2.30	2.94	2.38
LLaVA-NeXT-7B (Li et al., 2024)	1.73	0.00	0.00	0.00	1.17	0.58	1.24	1.78	1.51	0.45	1.04	0.67	0.72
Table-LLaVA-7B (Zheng et al., 2024)	57.78	18.43	10.09	12.82	25.60	24.94	59.85	65.26	62.56	9.74	**10.46**	**9.68**	**9.96**
Table-LLaVA-13B (Zheng et al., 2024)	59.77	20.41	10.85	15.67	**28.03**	26.95	**65.00**	**66.91**	**65.96**	**10.40**	8.83	9.67	9.63
Ours (LLaVA-NeXT-7B)													
MMSci-Pre (52k) + MMSci-Ins	8.76	3.22	0.63	0.39	5.99	3.80	35.78	25.37	30.57	1.57	1.10	1.78	1.48
MMTab-Pre (150k) + MMSci-Ins	9.00	2.62	0.63	0.26	7.23	3.95	36.22	26.91	31.56	1.64	0.84	1.57	1.35
MM-Pre (202k) + MMSci-Ins	10.66	4.83	0.82	0.65	9.39	5.27	39.63	27.63	33.63	1.13	0.83	1.90	1.29
w/o MM-Pre (202k)	9.69	2.74	0.19	0.39	6.84	3.97	31.72	23.80	27.76	1.69	0.79	1.53	1.34
Ours (Qwen2-VL-7B-Ins.)													
MMSci-Pre (52k) + MMSci-Ins	49.51	18.74	4.95	19.69	12.89	21.15	37.93	45.33	41.63	0.75	2.81	2.69	2.08
MMTab-Pre (150k) + MMSci-Ins	49.09	18.95	4.63	19.30	9.77	20.35	40.00	46.56	43.28	0.91	1.26	2.89	1.69
MM-Pre (202k) + MMSci-Ins	46.97	19.73	4.38	20.85	12.34	20.85	39.99	45.96	42.97	0.96	1.32	2.60	1.63
w/o MM-Pre (202k)	48.02	18.67	5.33	20.08	12.58	20.94	33.53	44.93	39.23	0.71	2.76	2.70	2.06

Table 2: Performance comparison on MMTab held-out datasets. Best results are in bold, second best are underlined.

Models	MMSci-Eval			Held-out	
	TQA	TFV	T2T	TABMWP	TAT-QA
Ours (LLaVA-NeXT-7B)					
MMSci-Pre (52k) + MMSci-Ins	17.72	57.12	2.93	49.47	10.46
w/o Reasoning	10.75	42.73	2.16	42.50	7.68
MMTab-Pre (150k) + MMSci-Ins	15.79	56.16	2.88	43.55	8.03
w/o Reasoning	9.58	50.31	1.93	42.50	7.42
MM-Pre (202k) + MMSci-Ins	23.02	58.57	2.36	49.72	12.27
w/o Reasoning	12.73	45.21	2.16	46.50	19.68
w/o MM-Pre (202k)	15.22	51.73	2.86	46.24	7.63
w/o Reasoning	9.43	42.31	2.36	45.50	8.39
Ours (Qwen2-VL-7B-Ins.)					
MMSci-Pre (52k) + MMSci-Ins	41.13	72.92	3.24	49.50	19.68
w/o Reasoning	35.06	66.47	3.14	44.08	16.72
MMTab-Pre (150k) + MMSci-Ins	40.75	72.73	3.16	49.08	19.30
w/o Reasoning	34.48	66.28	2.27	43.97	16.07
MM-Pre (202k) + MMSci-Ins	42.10	73.98	3.29	49.96	20.85
w/o Reasoning	35.45	67.43	1.97	46.34	17.68
w/o MM-Pre (202k)	41.71	70.90	3.29	48.02	20.07
w/o Reasoning	34.44	62.90	3.18	44.60	14.68

Table 3: Ablation study results for reasoning steps on MMSci-Eval and held-out datasets.

better performance (41.13% on TQA, 72.92% on TFV) compared to training with MMTab-Pre (150k) + MMSci-Ins (40.75% on TQA, 72.73% on TFV), despite using only one-third of the table structure learning data. The experiment setting of training with MM-Pre (202k) + MMSci-Ins achieves the best performance with 42.10% accuracy on TQA and 73.98% on TFV, while also demonstrating strong generalisation ability on held-out numerical reasoning datasets (49.96% on TABMWP and 20.85% on TAT-QA).

These results demonstrate that our proposed MMSci-Pre dataset with 52K scientific domain-specific data is more effective than MMTab-Pre with 150K general-domain data, highlighting the importance of data quality over quantity. Furthermore, Qwen2-VL-7B-Instruct consistently outperforms LLaVA-NeXT-7B across all experimental settings, suggesting its stronger capability in table understanding and numerical reasoning tasks. Besides, our approach shows strong generalisation to held-out tabular numerical reasoning datasets, demonstrating enhanced general ability in multi-modal table understanding and reasoning.

5.2 Performance on Held-out MMTab Benchmarks

The experimental results in Table 2 also demonstrate the effectiveness and generalisation ability of our proposed approach across various held-out MMTab benchmark. GPT-4V (OpenAI, 2023) show strong performance across all tasks, achieving 35.91% average accuracy on TQA, 55.55% on TFV, and 3.05 BLEU on T2T. Table-LLaVA models, which are specifically trained on MMTab-Ins dataset, demonstrate competitive performance. Table-LLaVA-13B achieves strong results on TFV (65.96% average accuracy) and T2T (9.63 BLEU) while Table-LLaVA-7B shows robust performance on TABMWP (57.78%).

As for our approaches, with LLaVA-NeXT-7B as the foundation model, we observe that training with MMSci-Pre (52k) and MMSci-Ins, despite not being trained on MMTab-Ins dataset (Zheng et al., 2024), demonstrates promising generalisation ability. The MMSci-Pre (52k) + MMSci-Ins combination achieves 3.80% average accuracy on TQA and 30.57% on TFV with only scientific domain data. The combination of both table structure learning datasets (MM-Pre 202k) further improves performance across all metrics, reaching 5.27% on TQA and 33.63% on TFV. As for Qwen2-VL-7B-Instruct as the foundation model, we observe significantly stronger generalisation capability. MMSci-Pre (52k) + MMSci-Ins combination achieves 21.15% average accuracy on TQA and 41.63% on TFV, demonstrating strong zero-

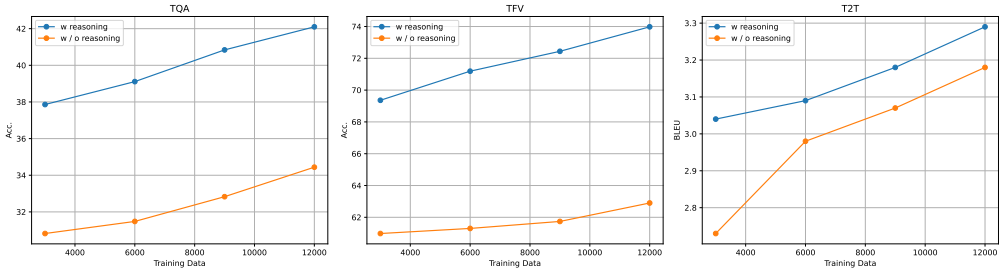

Figure 2: Performance scaling with increasing instruction tuning data size on three MMSci tasks.

shot transfer to MMTab benchmark despite using only scientific domain data (MMSci dataset). This performance is particularly impressive when compared to MMTab-Pre (150k) + MMSci-Ins combination, which uses three times more image-to-HTML data. Even without any table structure learning (w/o MM-Pre), our approach achieves competitive results, highlighting the effectiveness of our MMSci-Ins instruction tuning dataset.

These results empirically demonstrate our MMSci-Pre dataset with 52K scientific domain-specific data achieves comparable or better performance than MMTab-Pre with 150K general-domain data in MMTab held-out benchmark, highlighting the importance of scientific domain-specific tables. Even without MMTab table structure learning data, our approach demonstrates strong generalisation ability, particularly evident in the performance of MMSci-Pre (52k) + MMSci-Ins and w/o MM-Pre experiment settings.

5.3 Ablation Study on Reasoning Steps

We evaluate the effectiveness of reasoning steps across different experiment configurations. As shown in Table 3, Qwen2-VL-7B-Instruct demonstrates superior performance across all configurations. Without reasoning steps, the model training with MMSci-Pre (52k) + MMSci-Ins achieves better results than that with MMTab-Pre (150k) + MMSci-Ins, highlighting the importance of domain-specific table structure learning over data quantity. Adding reasoning steps consistently improves performance across all metrics, with the model reaching its peak performance under the MM-Pre (202k) + + MMSci-Ins experiment configuration. Similar trends are observed in LLaVA-NeXT-7B, though with lower overall performance. These patterns extend to held-out tabular numerical reasoning datasets, where both models show strong generalisation capabilities with reasoning steps, especially on numerical reasoning tasks like TABMWP and TAT-QA. The results demonstrate that a smaller amount of scientific domain-specific table structure learning data, combined with explicit reasoning steps, can be more effective than larger-scale general domain table structure learning.

5.4 Impact of Training Data Size

As shown in Figure 2, we compare performance of MLLMs training with MM-Pre (202k) + MMSci-Ins across three MMSci tasks (TQA, TFV, T2T) with instruction tuning data (MMSci-Ins dataset) size increasing from 3K to 12K samples. The findings demonstrate consistent advantages of incorporating reasoning steps across all data scales. Models trained with reasoning steps maintain substantial performance advantages across all tasks (7-8% for TQA, 8-10% for TFV, 0.3-0.4 BLEU for T2T). While both variants benefit from increased training data, models with reasoning steps show stronger scaling behavior, particularly in TQA and TFV tasks. The persistent performance gap across all data sizes suggests that reasoning steps provide fundamental improvements in model learning that cannot be simply achieved through increased training data alone.

5.5 Representational Alignment Analysis

In this section, we conduct an in-depth analysis to assess the language-vision alignment from the perspective of the representation space. This analysis aims to provide further insights into the observed variations in model performance, particularly in the context of scientific multimodal table understanding and reasoning tasks.

Preliminaries. We formalise MLLMs within the framework of an *unembedding-embedding* architecture. In this framework, the unembedding stage is responsible for learning transformations between observations (e.g., text, vision) and latent spaces

Models	Cycle KNN	Mutual KNN	Lcs KNN	CKA	CKNNA	SVCCA	Edit KNN
Unembedding stage: ImageNet(Concepts)							
Random	0.02761	0.01257	0.52355	0.08614	0.00714	0.12425	0.00019
Qwen2-VL-7B-Ins.	**0.68110**	0.03486	1.28153	0.08856	**0.03067**	**0.14318**	**0.00112**
Llama3.2-11B-Vision-Ins.	0.08608	**0.04205**	**1.52788**	0.06079	0.01403	0.11651	0.00061
LLaVA-NeXT-7B	0.57173	0.02077	0.81645	0.08024	0.01577	0.13240	0.00037
Phi3.5-Vision-Ins.	0.02761	0.01257	0.52355	0.08614	0.00714	0.12118	0.00019
InternVL2-8B	0.08175	0.01637	0.72495	**0.09185**	0.00062	0.12148	0.00044
Unembedding stage: Wikipedia Caption (short descriptive sentences)							
Qwen2-VL-7B-Ins.	0.49414	**0.06855**	**2.05078**	**0.08876**	**0.04093**	0.20229	**0.00175**
Llama3.2-11B-Vision-Ins.	0.31347	0.03623	1.29980	0.00968	0.00779	0.22120	0.00050
LLaVA-NeXT-7B	**0.57813**	0.03935	1.36523	0.07933	0.03998	**0.23114**	0.00082
Phi3.5-Vision-Ins.	0.04980	0.03027	1.14843	0.01669	0.03890	0.18183	0.00066
InternVL2-8B	0.36914	0.04132	1.55761	0.04732	0.01658	0.21739	0.00093
Embedding stage: MMSci T2T tasks (table to text description).							
Qwen2-VL-7B-Ins.	**0.38631**	**0.06726**	2.03660	**0.19318**	0.05514	**0.38461**	**0.00183**
Llama3.2-11B-Vision-Ins.	0.31310	0.02200	0.84007	1.73979e-8	0.03208	0.08180	0.00026
LLaVA-NeXT-7B	0.38246	0.04514	1.49325	0.15203	**0.06673**	0.28857	0.00109
Phi3.5-Vision-Ins.	0.38053	0.06712	**2.12909**	0.16121	0.03688	0.26982	0.00127
InternVL2-8B	0.36512	0.04651	1.56647	0.04230	0.02675	0.11876	0.00096

Table 4: Kernel alignment analysis. The representation for each sample is the averaged token embeddings. The best two values are shown in **bold** and underlined.

through encoders, while the embedding stage captures the complex interactions among latent variables within the latent space of LLMs' hidden layers. Each stage serves distinct functions and yields representations with different properties (Park et al., 2024). Consequently, by focusing on each stage independently, we can have a systematical evaluation of model behaviours in representation spaces. To assess the representational alignment between vision-language modalities at each stage, we next measure the geometrical similarity between them via the *kernel*.

Kernels, characterising the distance metrics between points in a representation space, are commonly used to assess vector space (Huh et al., 2024). Typically, the more similarity between two kernels derived from different spaces (text or vision) indicates a higher degree of alignment between those modality spaces. This similarity can be quantified via *kernel-alignment metrics*, such as Centered Kernel Distance (CKA) (Kornblith et al., 2019). For more information about kernel-alignment metrics used in the experiment, we refer to Huh et al. (2024) for a deep understanding.

Quantitative evaluation. For the unembedding stage, we specifically choose two language-vision datasets: ImageNet (Deng et al., 2009) and Wikipedia Caption (WIT) (Srinivasan et al., 2021). We randomly select 2048 samples from each dataset. These datasets offer varying levels of fine granularity in language-vision alignment, enabling a comprehensive assessment of representational performance. As illustrated in Table 4, we can observe that the Qwen2-VL-7B-Instruct can generally outperform other baselines on both datasets, indicating it has better fine-grained alignment between language and vision. In the embedding stage, we evaluate alignment on the MMSci T2T task. Since some models do not support single-modality input, we utilise a reference language model (e.g., open-llama- 7B (Geng and Liu, 2023)) as the text encoder and MLLMs as the image encoder with prompt *"please describe the table"*. Alignment is measured based on the output embedding from the last hidden layer. As shown in Table 4, Qwen2-VL-7B-Instruct outperforms the other models, demonstrating its superior language-vision alignment capability. This segment of the experiment demonstrates that the Qwen2-VL-7B-Instruct model exhibits superior language-vision alignment within the representation space. This finding is consistent with the cross-modal consistency analysis presented in Appendix C.1, where we evaluate different table information modalities as inputs to MLLMs and assess their cross-modal consistency (i.e., the proportion of identical predictions) on TQA and TFV tasks.

6 Conclusion

In this paper, we introduce a comprehensive framework for multimodal scientific table understanding and reasoning with dynamic input image resolutions. Experimental results validate our framework's effectiveness across different model architectures, showing consistent improvements in both general table understanding and numerical reasoning capabilities, with strong generalisation to held-out datasets.

Limitations

While this work advances scientific multimodal table understanding and reasoning, several limitations remain for future research. First, our framework primarily focuses on scientific tables containing numerical values, while other types of scientific tables (e.g., qualitative comparison tables, methodology tables) are not extensively covered. Second, though our framework demonstrates strong performance on numerical reasoning tasks, the current approach may still struggle with complex statistical analyses and domain-specific mathematical notations that are common in scientific literature. Third, while our models support dynamic input resolutions, processing extremely large tables with dense information remains challenging due to computational constraints and potential information loss during visual encoding.

Ethical Statement

The MMSci datasets are constructed from publicly available scientific papers and their associated tables, primarily sourced from open-access repositories and academic databases with appropriate licenses. All table images are generated through automated scripts from the original scientific papers, maintaining their integrity while ensuring proper attribution. The instruction tuning samples are created based on the original scientific context, preserving the academic nature of the source material. Our framework is designed to assist in scientific research by improving the accessibility and understanding of tabular data in academic literature. We anticipate that this work will contribute positively to the research community by facilitating more efficient analysis of scientific publications. The code and datasets are made publicly available for research purposes, promoting transparency and reproducibility in the field of multimodal scientific table understanding.

References

Pravesh Agrawal, Szymon Antoniak, Emma Bou Hanna, Baptiste Bout, Devendra Chaplot, Jessica Chudnovsky, Diogo Costa, Baudouin De Monicault, Saurabh Garg, Theophile Gervet, et al. 2024. Pixtral 12b. *arXiv preprint arXiv:2410.07073*.

Iñigo Alonso, Eneko Agirre, and Mirella Lapata. 2024. PixT3: Pixel-based table-to-text generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6721–6736, Bangkok, Thailand. Association for Computational Linguistics.

Pei Chen, Soumajyoti Sarkar, Leonard Lausen, Balasubramaniam Srinivasan, Sheng Zha, Ruihong Huang, and George Karypis. 2023. Hytrel: Hypergraph-enhanced tabular data representation learning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

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, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020a. Tabfact: A large-scale dataset for table-based fact verification.

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.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

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.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference*

on Computer Vision and Pattern Recognition, pages 248–255.

Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Xinyang Geng and Hao Liu. 2023. Openllama: An open reproduction of llama.

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.

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

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. 2024. Position: The platonic representation hypothesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 20617–20642. PMLR.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR.

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.

Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. 2023. Pix2Struct: Screenshot Parsing as Pretraining for Visual Language Understanding. ArXiv:2210.03347 [cs].

Bo Li, Kaichen Zhang, Hao Zhang, Dong Guo, Renrui Zhang, Feng Li, Yuanhan Zhang, Ziwei Liu, and Chunyuan Li. 2024. Llava-next: Stronger llms supercharge multimodal capabilities in the wild.

Hongxin Li, Jingran Su, Yuntao Chen, Qing Li, and Zhaoxiang Zhang. 2023a. Sheetcopilot: Bringing software productivity to the next level through large language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

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

Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2022. TAPEX: Table pre-training via learning a neural SQL executor. In *International Conference on Learning Representations*.

Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. 2023a. Chameleon: Plug-and-play compositional reasoning with large language models. In *The 37th Conference on Neural Information Processing Systems (NeurIPS)*.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. 2023b. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. In *International Conference on Learning Representations (ICLR)*.

Xinyuan Lu, Liangming Pan, Qian Liu, Preslav Nakov, and Min-Yen Kan. 2023c. SCITAB: A challenging benchmark for compositional reasoning and claim verification on scientific tables. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7787–7813, Singapore. Association for Computational Linguistics.

Meta. 2024. Llama 3.2: Pushing the boundaries of vision and language for edge and mobile devices. Accessed: 2024-12-06.

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, and Dragomir Radev. 2022. Fetaqa: Free-form table question answering. *Transactions of the Association for Computational Linguistics*, 10:35–49.

OpenAI. 2023. Gpt-4v. `https://openai.com/index/gpt-4v-system-card/`. Accessed: 2023-02-09, 2023-02-11, 2023-02-12.

OpenAI. 2024. Hello gpt-4o. `https://openai.com/index/hello-gpt-4o/`. Accessed: 2024-02-09, 2024-02-11, 2024-02-12.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics*.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2024. The linear representation hypothesis and the geometry of large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 39643–39666. PMLR.

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.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision.

Krishna Srinivasan, Karthik Raman, Jiecao Chen, Michael Bendersky, and Marc Najork. 2021. Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning. *ArXiv preprint, abs/2103.01913*.

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

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024b. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2023. Gpt4table: Can large language models understand structured table data? a benchmark and empirical study.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *ArXiv preprint*, abs/2409.12191.

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].

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024a. Qwen2 technical report.

Bohao Yang, Chen Tang, Kun Zhao, Chenghao Xiao, and Chenghua Lin. 2024b. Effective distillation of table-based reasoning ability from llms. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5538–5550.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. 2024. Minicpm-v: A gpt-4v level mllm on your phone. *ArXiv preprint*, abs/2408.01800.

Yunhu Ye, Binyuan Hui, Min Yang, Binhua Li, Fei Huang, and Yongbin Li. 2023. Large language models are versatile decomposers: Decomposing evidence and questions for table-based reasoning. In *Proceedings of the 46th International ACM Conference on Research and Development in Information Retrieval (SIGIR)*, pages 174–184.

Pengcheng Yin, Zhengdong Lu, Hang Li, and Ben Kao. 2016. Neural enquirer: Learning to query tables in natural language. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2308–2314.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene 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. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3911–3921.

Tianshu Zhang, Xiang Yue, Yifei Li, and Huan Sun. 2023. Tablellama: Towards open large generalist models for tables.

Yilun Zhao, Yitao Long, Hongjun Liu, Ryo Kamoi, Linyong Nan, Lyuhao Chen, Yixin Liu, Xiangru Tang, Rui Zhang, and Arman Cohan. 2024. DocMath-eval: Evaluating math reasoning capabilities of LLMs in understanding long and specialized documents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16103–16120, Bangkok, Thailand. Association for Computational Linguistics.

Mingyu Zheng, Xinwei Feng, Qingyi Si, Qiaoqiao She, Zheng Lin, Wenbin Jiang, and Weiping Wang. 2024. Multimodal Table Understanding. In *Proceedings*

of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 9102–9124, Bangkok, Thailand. Association for Computational Linguistics.

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 Details about MMSci

A.1 Datasets Statistics

Table 5 presents the distribution of reasoning types in our MMSci-Eval dataset. The most common type is addition (21.1%), followed by subtraction (15.3%) and max/min operations (15.7%). Division and comparison operations also appear frequently (14.2% and 13.7% respectively). More complex operations like ranking (9.6%) and look-up (8.9%) occur less frequently, while domain knowledge calculations are rare (1.5%).

The average number of reasoning steps varies significantly across types, with subtraction requiring the most steps (4.1) and look-up operations requiring the fewest (1.5). This variation reflects the inherent complexity of different mathematical operations and their application to tabular data. Notably, even seemingly simple operations like addition require multiple steps (2.8) on average, indicating the complexity of reasoning with tabular scientific data.

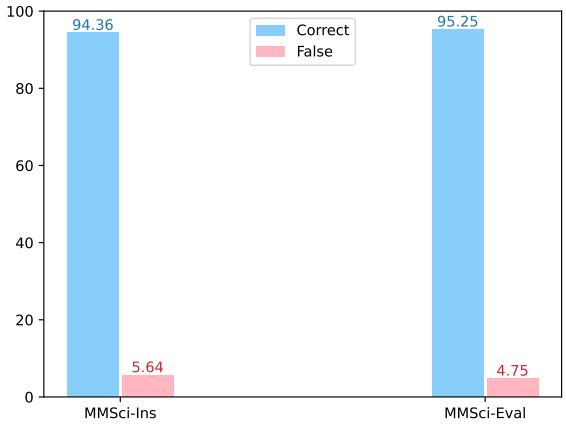

Figure 3: Evaluation of generated data of MMSci-Ins and MMSci-Eval dataset. Correct refers to the data verified correctly by human annotators.

A.2 Dataset Quality Control

To ensure data quality, we conduct a rigorous human verification process for both MMSci-Ins and MMSci-Eval datasets. For MMSci-Ins, we manually verify 40% of the generated samples, achieving a high accuracy rate of 94.36%. For MMSci-Eval, given its critical role as a benchmark, we carefully examine all 3,114 generated samples and achieve an accuracy of 95.25%. For any identified incorrect samples, we employ GPT-4o to regenerate them following the same self-consistency voting mechanism, followed by another round of both automatic

Reasoning Type	Description	Avg. Reasoning Step	Prop.%
Add	Calculate the sum between numbers	2.8	21.1
Comparison	Comparison of values	2.1	13.7
Domain Knowledge Calculation	Calculations need domain knowledge	2.2	1.5
Divide	Perform division between numbers	3.4	14.2
Look Up	Search for cells in tables	1.5	8.9
Max/Min	Retrieve the maximum or minimum number	3.2	15.7
Ranking	Arranges items in a specific order	2.4	9.6
Subtract	Perform subtraction between numbers	4.1	15.3

Table 5: The reasoning types, descriptions, average reasoning step, and proportion in our dataset.

and manual verification to ensure quality. This iterative process ensures the reliability and correctness of our datasets for both training and evaluation purposes.

A.3 Prompt for Generating Data

The prompt for MMSci-Ins and MMSci-Pre data generation is shown in Table 7.

B Experimental Settings

Implementation Details. Both models follow a three-component design. Qwen2-VL-7B-Instruct consists of a Vision Transformer (ViT) (Dosovitskiy, 2020) as the vision tower, a MLP as the vision-language connector, and Qwen2 (Wang et al., 2024) as the language model. LLaVA-NeXT-7B uses a pre-trained CLIP model (Radford et al., 2021) as the visual encoder, a MLP connector, and Vicuna-7B (Chiang et al., 2023) as the backbone. In both architectures, the vision encoder processes images into visual features, which are projected into the LLM's word embedding space via the MLP connector.

Training Details. All experiments are conducted on 4×A100 80GB GPUs using LoRA with rank 64 and sequence length 4096. For table structure learning, LLaVA-NeXT-7B requires 15 hours for MMTab-Pre (150k), 3 hours for MMSci-Pre (52k), and 20 hours for combined training (one epoch). Qwen2-VL-7B takes 15 hours, 8 hours, and 19 hours respectively. The instruction tuning stage requires approximately 1 hour for 4 epochs with 12k samples for both models.

C More Experimental Results and Analysis

C.1 Vision-Language Consistency Analysis

We evaluate the cross-modal consistency of different MLLMs by comparing their performance when processing table information through different modalities. For each model, we test with both table images (image modality) and their textual representations (text modality), measuring both task performance (Acc.) and cross-modal consistency (Consis.).

Qwen2-VL-7B-Instruct demonstrates superior cross-modal alignment, achieving the highest consistency scores on both TQA (60.40%) and TFV (72.48%) tasks. Notably, it maintains strong performance across both modalities, with image-based accuracy (TQA: 39.11%, TFV: 52.79%) consistently outperforming text-based results (TQA: 21.65%, TFV: 50.10%). This suggests robust integration of visual and textual understanding capabilities.

Other models show varying degrees of modality gap. MiniCPM-V-2.6-8B and InternVL-2-8B achieve moderate consistency (48.78% and 50.89% on TQA), while models like LLaVA-NeXT-7B and Pixtral-12B show significant disparities between modalities, resulting in lower consistency scores. These results highlight the challenge of maintaining consistent reasoning capabilities across different input modalities in table understanding tasks.

D Dataset Examples and Case Study

D.1 Dataset Examples

The training examples of MMSci-Pre Dataset are shown in Figure 4. The examples of MMSci-Eval are shown Figure 6.

D.2 Case Study

The case study of TQA and TFV tasks are shown in Figure 6 and Figure 7, respectively

Model	Modal	TQA Acc.	TQA Consis.	TFV Acc.	TFV Consis.
Qwen2-VL-7B-Ins. (Wang et al., 2024)	Text	**21.65**	**60.40**	**50.10**	**72.48**
	Image	**39.11**		**52.79**	
LLaVA-NeXT-7B (Li et al., 2024)	Text	3.17	14.81	2.03	23.65
	Image	0.19		0.86	
MiniCPM-V-2.6-8B (Yao et al., 2024)	Text	21.11	48.78	30.82	38.53
	Image	26.58		33.23	
InternVL-2-8B (Chen et al., 2024)	Text	19.84	50.89	42.87	36.42
	Image	25.72		44.99	
Pixtral-12B (Agrawal et al., 2024)	Text	1.44	16.52	4.43	29.88
	Image	0.96		5.49	
Llama-3.2-11B-Vision-Ins. (Meta, 2024)	Text	3.24	15.71	6.96	20.40
	Image	1.15		5.85	

Table 6: Vision-language consistency evaluation across different MLLMs. Consistency scores measure the percentage of identical responses between modalities, indicating the model's cross-modal alignment.

Task

Table Image-to-HTML

Input:
Please review this table image and return a text representation of the table in HTML format.

Table Image

Model	Target Corpus	Transfer BERT	Transfer Filler	Transfer Role	Baseline Acc. (%)	Fine-tuned Acc. (%)	Gain (%)
BERT	QNLI	True	–	–	91.60	91.27	– 0.33
BERT	QQP	True	–	–	91.45	91.12	– 0.33
BERT	RTE	True	–	–	71.12	73.65	+ 2.53
BERT	SNLI	True	–	–	90.45	90.69	+ 0.24
BERT	SST	True	–	–	93.23	92.78	– 0.45
HUBERT (Transformer)	QNLI	True	True	False	90.56	91.16	+ 0.60
HUBERT (Transformer)	QQP	False	False	True	90.81	91.42	+ 0.61
HUBERT (Transformer)	RTE	True	True	True	61.73	74.01	+ 12.28
HUBERT (Transformer)	SNLI	True	False	True	90.66	91.36	+ 0.70
HUBERT (Transformer)	SST	True	False	True	91.28	92.43	+ 1.15

Table 3: Transfer learning results for GLUE tasks. The source corpus is MNLI. Baseline accuracy is when Transfer BERT, Filler, and Role are all False, equivalent to no transfer. Fine-tuned accuracy is the best accuracy among all possible transfer options.

Output:

<table border=\"1\" cellspacing=\"0\">\n<tr> <th class=\"highlighted\" colspan=1 rowspan=1 > Model </th><th class=\"highlighted\" colspan=1 rowspan=1 > Target Corpus </th><th class=\"highlighted\" colspan=1 rowspan=1 > Transfer BERT </th><th class=\"highlighted\" colspan=1 rowspan=1 > Transfer Filler </th><th class=\"highlighted\" colspan=1 rowspan=1 > Transfer Role </th><th class=\"highlighted\" colspan=1 rowspan=1 > Baseline Acc. (%) </th><th class=\"highlighted\" colspan=1 rowspan=1 > Fine-tuned Acc. (%) </th><th class=\"highlighted\" colspan=1 rowspan=1 > Gain (%) </th></tr>\n<tr> <td colspan=1 rowspan=1 > BERT </td><td colspan=1 rowspan=1 > QNLI </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > 91.60 </td><td colspan=1 rowspan=1 > 91.27 </td><td colspan=1 rowspan=1 > – 0.33 </td></tr>\n<tr> <td colspan=1 rowspan=1 > BERT </td><td colspan=1 rowspan=1 > QQP </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > 91.45 </td><td colspan=1 rowspan=1 > 91.12 </td><td colspan=1 rowspan=1 > – 0.33 </td></tr>\n<tr> <td colspan=1 rowspan=1 > BERT </td><td colspan=1 rowspan=1 > RTE </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > 71.12 </td><td colspan=1 rowspan=1 > 73.65 </td><td colspan=1 rowspan=1 > + 2.53 </td></tr>\n<tr> <td colspan=1 rowspan=1 > BERT </td><td colspan=1 rowspan=1 > SNLI </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > 90.45 </td><td colspan=1 rowspan=1 > 90.69 </td><td colspan=1 rowspan=1 > + 0.24 </td></tr>\n<tr> <td colspan=1 rowspan=1 > BERT </td><td colspan=1 rowspan=1 > SST </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > – </td><td colspan=1 rowspan=1 > 93.23 </td><td colspan=1 rowspan=1 > 92.78 </td><td colspan=1 rowspan=1 > – 0.45 </td></tr>\n<tr> <td colspan=1 rowspan=1 > HUBERT (Transformer) </td><td colspan=1 rowspan=1 > QNLI </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > False </td><td colspan=1 rowspan=1 > 90.56 </td><td colspan=1 rowspan=1 > 91.16 </td><td class=\"highlighted\" colspan=1 rowspan=1 > + 0.60 </td></tr>\n<tr> <td colspan=1 rowspan=1 > HUBERT (Transformer) </td><td colspan=1 rowspan=1 > QQP </td><td colspan=1 rowspan=1 > False </td><td colspan=1 rowspan=1 > False </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > 90.81 </td><td colspan=1 rowspan=1 > 91.42 </td><td class=\"highlighted\" colspan=1 rowspan=1 > + 0.61 </td></tr>\n<tr> <td colspan=1 rowspan=1 > HUBERT (Transformer) </td><td colspan=1 rowspan=1 > RTE </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > 61.73 </td><td colspan=1 rowspan=1 > 74.01 </td><td class=\"highlighted\" colspan=1 rowspan=1 > + 12.28 </td></tr>\n<tr> <td colspan=1 rowspan=1 > HUBERT (Transformer) </td><td colspan=1 rowspan=1 > SNLI </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > False </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > 90.66 </td><td colspan=1 rowspan=1 > 91.36 </td><td class=\"highlighted\" colspan=1 rowspan=1 > + 0.70 </td></tr>\n<tr> <td colspan=1 rowspan=1 > HUBERT (Transformer) </td><td colspan=1 rowspan=1 > SST </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > False </td><td colspan=1 rowspan=1 > True </td><td colspan=1 rowspan=1 > 91.28 </td><td colspan=1 rowspan=1 > 92.43 </td><td class=\"highlighted\" colspan=1 rowspan=1 > + 1.15 </td></tr>\n</table>"

Figure 4: MMSci-Pre Dataset example

The prompt for Generating data
You are given a table image and a description: {description}. 1.For the Table to Text (T2T) task, come up with a one to two sentence length succinct multi-hop reasoning step of the description. Write your results as 'T2T Reasoning:' and then the succinct reasoning step. 2.For the Table Question Answering (TQA) task, come up with a question and answer with multi-hop reasoning step. The question and answer must be based on the table image and description. Write your results as 'TQA Question:' and then the question and 'TQA Reasoning:' and then the reasoning step and 'TQA Answer:' and then the answer. When generating 'TQA Question:', make sure it is a single question that requires reasoning based on the table. When generating 'TQA Answer:', provide the final answer in the JSON structure, using the format "answer": "<YOUR ANSWER>" Make sure the answer only contains one entity, such as 'So, the answer is "answer": "23".' 3.For the Table Fact Checking (TFV) task, come up with a statement and answer with multi-hop reasoning step. The statement and answer must be based on the table image and description. The table 'supports' or 'refutes' the statement. The statement should be considered 'not enough info' if it may or may not be true. Write your results as 'TFV Statement:' and then the statement and 'TFV Reasoning:' and then the reasoning step and 'TFV Answer:' and then the answer. Make sure the answer only contains one entity, such as 'Thus, the answer is "answer": "supports".' When generating 'TFV Answer:', provide the final answer in the JSON structure, using the format "answer": "<YOUR ANSWER>" Fill the result into JSON format without any other words: "T2T Reasoning": "<YOUR T2T REASONING>", "TQA Question": "<YOUR TQA QUESTION>", "TQA Reasoning": "<YOUR TQA REASONING>", "TQA Answer": "<YOUR TQA ANSWER>", "TFV Statement": "<YOUR TFV STATEMENT>", "TFV Reasoning": "<YOUR TFV REASONING>", "TFV Answer": "<YOUR TFV ANSWER>" Examples: {TQA Examples} {TFV Examples} {T2T Examples}

Table 7: The prompts for generating the questions, reasoning steps, and answers or claims of MMSci-Ins and MMSci-Eval datasets.

Task | **Table Image** | **Input:** Task, Output format, Task-related input | **Output:** Reasoning, Answer or description

Table Question Answer

Refer to the provided table image and work through the question step by step. Output the final answer as JSON in the format {'answer': '<YOUR ANSWER>'}. Question: 'Which model shows the highest gain in accuracy for the RTE target corpus

By comparing the gain percentages for the RTE target corpus, HUBERT (Transformer) shows a gain of +12.28%, which is higher than BERT's gain of +2.53%.", **So, the answer is {"answer": "HUBERT (Transformer)"}.**

Model	Target Corpus	Transfer BERT	Transfer Filler	Transfer Role	Baseline Acc. (%)	Fine-tuned Acc. (%)	Gain (%)
BERT	QNLI	True	–	–	91.60	91.27	– 0.33
BERT	QQP	True	–	–	91.45	91.12	– 0.33
BERT	RTE	True	–	–	71.12	73.65	+ 2.53
BERT	SNLI	True	–	–	90.45	90.69	+ 0.24
BERT	SST	True	–	–	93.23	92.78	– 0.45
HUBERT (Transformer)	QNLI	True	True	False	90.56	91.16	+ 0.60
HUBERT (Transformer)	QQP	False	False	True	90.81	91.42	+ 0.61
HUBERT (Transformer)	RTE	True	True	True	61.73	74.01	+ 12.28
HUBERT (Transformer)	SNLI	True	False	True	90.66	91.36	+ 0.70
HUBERT (Transformer)	SST	True	False	True	91.28	92.43	+ 1.15

Table Fact Verification

Using the given table image, determine if the claim is 'supports' or 'refutes'. If the table does not provide sufficient information, the answer should be 'not enough info'. Output the final answer as JSON in the format {'answer': '<YOUR ANSWER>'}. BERT shows a gain in accuracy for the QNLI target corpus.

The table shows that BERT has a gain of -0.33% for the QNLI target corpus, indicating an decrease in accuracy. **Thus, the answer is {\"answer\": \"refutes\"}.**

Table to Text Generation

Table 3: Transfer learning results for GLUE tasks. The source corpus is MNLI. Baseline accuracy is when Transfer BERT, Filler, and Role are all False, equivalent to no transfer. Fine-tuned accuracy is the best accuracy among all possible transfer options.

You are provided with a table image with caption. Formulate a statement describing the table.
Table Caption: 'Table 3: Transfer learning results for GLUE tasks. The source corpus is MNLI. Baseline accuracy is when Transfer BERT, Filler, and Role are all False, equivalent to no transfer. Fine-tuned accuracy is the best accuracy among all possible transfer options.

HUBERT (Transformer) shows substantial gains across all target corpora after transfer learning from MNLI, while BERT shows a drop in accuracy for QNLI, QQP, and SST. Table 3 summarizes the results for these transfer learning experiments when the source task is MNLI. Gain shows the difference between Fine-tuned model's accuracy and Baseline's accuracy. For HUBERT (Transformer), we observe substantial gain across all 5 target corpora after transfer. However, for BERT we have a drop for QNLI, QQP, and SST.

Figure 5: MMSci-Ins and MMSci-Eval Dataset example

Task: TQA | **Output: correct, wrong** | **Answer: HUBERT (Transformer)**

Refer to the provided table image and work through the question step by step. Output the final answer as JSON in the format {'answer': '<YOUR ANSWER>'}. Question: 'Which model shows the highest gain in accuracy for the RTE target corpus

GPT-4V

To determine which model shows the highest gain in accuracy for the RTE target corpus, we need to compare the \"Gain (%)\" values for the RTE target corpus in the table. Comparing the gains, the HUBERT (Transformer) model shows the highest gain in accuracy for the RTE target corpus with a gain of +12.28%.\n\nFinal answer in JSON format:\n\n```json\n{\n \"answer\": \"HUBERT (Transformer)\"\n}\n```"

Model	Target Corpus	Transfer BERT	Transfer Filler	Transfer Role	Baseline Acc. (%)	Fine-tuned Acc. (%)	Gain (%)
BERT	QNLI	True	–	–	91.60	91.27	– 0.33
BERT	QQP	True	–	–	91.45	91.12	– 0.33
BERT	RTE	True	–	–	71.12	73.65	+ 2.53
BERT	SNLI	True	–	–	90.45	90.69	+ 0.24
BERT	SST	True	–	–	93.23	92.78	– 0.45
HUBERT (Transformer)	QNLI	True	True	False	90.56	91.16	+ 0.60
HUBERT (Transformer)	QQP	False	False	True	90.81	91.42	+ 0.61
HUBERT (Transformer)	RTE	True	True	True	61.73	74.01	+ 12.28
HUBERT (Transformer)	SNLI	True	False	True	90.66	91.36	+ 0.70
HUBERT (Transformer)	SST	True	False	True	91.28	92.43	+ 1.15

Table 3: Transfer learning results for GLUE tasks. The source corpus is MNLI. Baseline accuracy is when Transfer BERT, Filler, and Role are all False, equivalent to no transfer. Fine-tuned accuracy is the best accuracy among all possible transfer options.

Ours (Qwen2-VL-7B-Ins)

By examining the 'Gain (%)' column for the RTE target corpus, we see that HUBERT (Transformer) with Transfer BERT, Transfer Filler, and Transfer Role all True has the highest gain of +12.28%. So, the answer is {'answer': 'HUBERT (Transformer)'

Intern2-VL-8B

{ \"answer\": HUBERT}

Table-LLaVA-13B

"Based on the information in the table, 12.29% > 11.81%. Therefore, the answer is {\"answer\": \"HUBERT (Transformer)\"}."

LLaVA-NeXT-7B

Fine-tuned accuracy is the highest among all transfer options, with a gain of 15.0% over the baseline.

Figure 6: A case example of TQA task

Task: TFV **Output: correct, wrong** **Answer: Refutes**

Using the given table image, determine if the claim is 'supports' or 'refutes'. If the table does not provide sufficient information, the answer should be 'not enough info'. Output the final answer as JSON in the format {'answer': '<YOUR ANSWER>'}. BERT shows a gain in accuracy for the QNLI target corpus.

Model	Target Corpus	Transfer BERT	Transfer Filler	Transfer Role	Baseline Acc. (%)	Fine-tuned Acc. (%)	Gain (%)
BERT	QNLI	True	–	–	91.60	91.27	– 0.33
BERT	QQP	True	–	–	91.45	91.12	– 0.33
BERT	RTE	True	–	–	71.12	73.65	+ 2.53
BERT	SNLI	True	–	–	90.45	90.69	+ 0.24
BERT	SST	True	–	–	93.23	92.78	– 0.45
HUBERT (Transformer)	QNLI	True	True	False	90.56	91.16	+ 0.60
HUBERT (Transformer)	QQP	False	False	True	90.81	91.42	+ 0.61
HUBERT (Transformer)	RTE	True	True	True	61.73	74.01	+ 12.28
HUBERT (Transformer)	SNLI	True	False	True	90.66	91.36	+ 0.70
HUBERT (Transformer)	SST	True	False	True	91.28	92.43	+ 1.15

Table 3: Transfer learning results for GLUE tasks. The source corpus is MNLI. Baseline accuracy is when Transfer BERT, Filler, and Role are all False, equivalent to no transfer. Fine-tuned accuracy is the best accuracy among all possible transfer options.

GPT-4V

```json\n{\n \"answer\": \"supports\"\n}\n```

Ours (Qwen2-VL-7B-Ins)

The table shows that for the QNLI target corpus, the gain for BERT is -0.33%, indicating a decrease in accuracy. Thus, the answer is {\"answer\": \"refutes\"}.

Intern2-VL-8B

The answer can be represented in JSON as follows:\n\n```json\n{\n \"answer\": \"supports\"\n}\n```

Table-LLaVA-13B

Based on the table information, the given claim is false. Thus, the answer is {\"answer\": \"refutes\"}.

LLaVA-NeXT-7B

The statement is also ambiguous as it refers to \"QNLI target corpus,\" which could refer to a specific dataset or context, but the image does not provide enough information to understand this context.

Figure 7: A case example of TFV task
```