Efficient Table Retrieval and Understanding with Multimodal Large Language Models

Zhuoyan Xu[1*], Haoyang Fang[2], Boran Han[2], Bonan Min[2], Bernie Wang[2], Cuixiong Hu[2], Shuai Zhang[2]
[1]University of Wisconsin-Madison, [2]AWS
{zhuoyanxu1,kevinfang97}@gmail.com

Abstract

Tabular data is frequently captured in image form across a wide range of real-world scenarios such as financial reports, handwritten records, and document scans. These visual representations pose unique challenges for machine understanding, as they combine both structural and visual complexities. While recent advances in Multimodal Large Language Models (MLLMs) show promising results in table understanding, they typically assume the relevant table is readily available. However, a more practical scenario involves identifying and reasoning over relevant tables from large-scale collections to answer user queries. To address this gap, we propose TabRAG, a framework that enables MLLMs to answer queries over large collections of table images. Our approach first retrieves candidate tables using jointly trained visual-text foundation models, then leverages MLLMs to perform fine-grained reranking of these candidates, and finally employs MLLMs to reason over the selected tables for answer generation. Through extensive experiments on a newly constructed dataset comprising 88,161 training and 9,819 testing samples across 8 benchmarks with 48,504 unique tables, we demonstrate that our framework significantly outperforms existing methods by **7.0%** in retrieval recall and **6.1%** in answer accuracy, offering a practical solution for real-world table understanding tasks.

1 Introduction

Multimodal Large Language Models (MLLMs) have achieved significant success in many fields, drawing increasing research attention (OpenAI, 2023; Anthropic, 2024). Most MLLMs thrive in the vision and text domains, especially large language models (LLMs). On the other hand, tabular data remains predominant in many fields, such as finance, healthcare, and census data (Borisov et al., 2022; van Breugel and van der Schaar, 2024). Thus, it is an active research area in developing a framework that enables MLLMs to better handle tabular data. Tabular data encompasses a variety of tasks, one of major problems is table understanding, including table question answering, table fact verification, and so on (Wang et al., 2024; Deng et al., 2020). These tasks often involve one or more tables and a relevant question that requires cross-referencing information across rows, columns, or tables (Fang et al., 2024).

Current research on table understanding involves providing models with a table and a query based on that table. Most related work focuses on tasks involving table-query pairs by generating correct responses to table-related requests (e.g., questions) in an end-to-end manner based on the corresponding table (Zhang et al., 2023; Nan et al., 2022; Cheng et al., 2022; Zheng et al., 2024). However, in real-world scenarios we often do not have the most relevant table by hand. We propose a more practical task where user posing a general query with massive tables stored in a large data store. The query-answer workflow often involves identifying the correct ground truth table from a certain database without manual intervention. One common scenario is that users query information stored in tabular format within a large database. The system must search through a vast collection of tables to identify the most relevant ones, and then the model extracts the necessary information to accurately answer the query. Developing a robust framework for this task could significantly improve practicality and efficiency in handling real-world scenarios involving tabular data.

Several approaches have been proposed to bridge the gap between tabular data and large language models (LLMs) (Hegselmann et al., 2023; Wen et al., 2024; Dinh et al., 2022; Fernandez et al., 2023; Zhang et al., 2023). However, existing tabular LLMs rely solely on the prerequisite that the

[*]Work done as an intern at Amazon Web Services.

Findings of the Association for Computational Linguistics: EACL 2026, pages 4327–4340

given tables are clean text sequences, such as markdown representations or HTML. These clean formats are then fed into the model for downstream tasks. In this work, we consider tables in image format for the following reasons. Table images are often more accessible than textual formats, especially in scanned documents or webpage screenshots, providing convenience to build a data store (Zheng et al., 2024). Directly working with table images bypasses the need for OCR conversion, avoiding OCR-related challenges such as high cost, computational overhead, limited adaptability, and error propagation (Kim et al., 2022). For inference on LLMs, treating tables as images has additional benefits. Textual representations can disrupt table structure, especially tables with images, merged cells, cells with color highlight, and high level references, whereas humans intuitively understand two-dimensional tables visually. Therefore, exploring direct processing of table images using visual features is crucial for improved performance.

To address these challenges, we propose **TabRAG**, a framework that leverages MLLMs throughout the table understanding process. Our approach consists of three components: (1) a retrieval system with jointly trained visual and text encoders that generate embeddings for table images and text queries respectively, enabling efficient semantic search in large image collections, (2) a reranking mechanism where MLLMs perform fine-grained analysis of table-query pairs to identify the most relevant candidates, and (3) a generation module where MLLMs reason over the query together with the selected table images to produce accurate answers. By combining specialized retrieval models with multimodal understanding, our framework effectively bridges the gap between textual queries and tabular information stored in image format.

We summarize our contributions as following three folds:

- We introduce TabRAG, an end-to-end framework that addresses the practical challenge of retrieving and understanding table images from large collections given text queries.

- We develop a specialized retrieval system with jointly trained visual-text foundation models and an MLLM-based reranking mechanism, effectively identifying relevant tables from large image collections.

- Through comprehensive experiments on retrieval effectiveness, reranking accuracy, and answer quality, we demonstrate that our approach significantly outperforms existing retrieval methods and MLLM baselines across various table understanding tasks.

2 Related Work

Multimodal Large Language Models As LLMs revolutionize both natural language processing (NLP) and the broader AI community, research is increasingly focusing on expanding their capabilities beyond text to encompass image (Liu et al., 2023b; Li et al., 2023; Bai et al., 2023), video (Li et al., 2024; Lin et al., 2024), and audio (Latif et al., 2023; Yao et al., 2024) modalities. Such development leads to the emergence of MLLMs. Flamingo (Alayrac et al., 2022) inserts gated cross-attention dense blocks between vision encoder and LLM, aligning vision and language modality. BLIP2 (Li et al., 2023) introduce Q-former, bridging pre-trained image encoders and LLMs to enable visual instruction capability. LLaVA (Liu et al., 2023b,a) uses simple MLP to connect vision embedding space and text token space and show state-of-the-art performance on a variety of tasks. Our work is built on such works and develop a framework for MLLMs on tablular data.

Context Engineering. Context Engineering enhances LLMs by integrating external knowledge sources with input queries, enriching the model with additional context for knowledge-intensive tasks (Lewis et al., 2020; Guu et al., 2020). Context engineering employs retrieval methods to identify relevant documents and integrates them with the input prompt to enhance response generation in LLMs or MLLMs (Asai et al., 2023a,b; Chen et al., 2024). In this work, we develop a context-enhanced framework specifically designed for MLLMs to handle multimodal tabular data.

Table Understanding. Table Understanding is one of the main areas in tabular data (Zhang et al., 2023; Nan et al., 2022; Cheng et al., 2022; Zheng et al., 2024), involving extraction, understanding, interpretation of the information from the table. Table understanding includes question answering (Nan et al., 2022; Lebret et al., 2016; Cheng et al., 2022), fact verification (Gupta et al., 2020), natural language generation and interpretation (Parikh et al., 2020; Chen et al., 2020). Several works have been proposed to solve table understanding problems using LLMs. One common strategy involves converting table data into natural language, facilitating text-based table reasoning (Ye et al., 2023; Yin et al., 2020; Singha et al., 2023; Sui et al., 2024). As MLLMs have advanced, recent studies have shifted towards processing tables as images, leveraging visual perception capabilities (Huang et al., 2022; Faysse et al., 2024; Zheng et al., 2024; Liu et al., 2023a). Following this line of work, our approach treats tables as visual data, incorporating them into both retrieval and generation processes using vision-based techniques.

3 Framework

We formally define our problem of TabRAG. We consider a large table image datastore consisting of a vast collection of images $\mathcal{S} = \{x_1, \ldots, x_N\}$, where N is the total number of tables in datastore.

Let f_θ be the MLLM in consideration, and h_α as vision encoder, and g_β as text encoder. Inspired by Yu et al. (2024), we conduct *retrieve-rerank-generation* pipeline during the inference. Given a user query q and a large store of images $\mathcal{S}$, we have $g_\beta(q)$ as the embedding of query. We have computed the embedding of images in large table collection as $E_\mathcal{S} = \{h_\alpha(x_i)\}_{i=1}^N$. In Retrieval step, we retrieve top n images of the query based on the cosine similarity of $g_\beta(q)$ and each image embedding in $E_\mathcal{S}$, resulting in a subset of images $\mathcal{S}_n$, where $|\mathcal{S}_n| = n$. In reranking step, we then concatenate query q with each image in $\mathcal{S}_n$, resulting in n different pairs, we prepend prompt asking MLLMs whether the image is relevant to the question and return `True` or `False`. Then we compute the probability of model output `True` token and rank them, keeping top-k tables as final contexts. In Generation step, we concatenate top-k table images and query into MLLMs to generate the final answer. We show our general pipeline in Figure 1. We detailed our training pipelines in the following sections.

3.1 Bi-Encoder Retriever

To efficiently match user queries with relevant table images given large data store, we implement a bi-encoder retriever architecture. This approach allows for separate encoding of queries and table images, enabling fast retrieval through similarity search in a shared embedding space.

We fine-tune these encoders using a contrastive learning approach. The training data consists of pairs of table images and corresponding queries, denoted as $\{(q_i, x_i)\}_{i \in I}$, where I represents batches of training samples. The training objective is to maximize the similarity between matched query-image pairs while minimizing similarity with non-matching pairs. This is achieved through a contrastive loss function:

$$\mathcal{L}(\alpha, \beta) = -\log\left(\frac{\exp\{\langle g_\beta(q_i), h_\alpha(x_i)\rangle\}}{\sum_{j \in I}\exp\{\langle g_\beta(q_j), h_\alpha(x_j)\rangle\}}\right)$$

where (α, β) are trainable parameters in retrievers. Here, $\langle\cdot,\cdot\rangle$ denotes the cosine similarity between the encoded representations. The numerator here encourages high similarity for matched pairs, while the denominator, summing over all samples in the batch, acts as a normalization factor and implicitly pushes non-matching pairs apart.

During training, the vision and text encoders are jointly fine-tuned to create a unified embedding space for direct visual-textual comparison. This alignment enables efficient table image indexing, fast similarity search, and effective query-based ranking. Once training is complete, the refined encoders are employed in the retrieval stage. Given a user query, the system encodes it using $h_\alpha(q)$, compares this embedding to pre-computed embeddings of table images in the database, and retrieves the top-k most similar table images based on cosine similarity. We use FAISS index search system (Douze et al., 2024) for fast search and retrieve. This bi-encoder architecture ensures scalable retrieval, quickly narrowing large table collections to relevant candidates and improving both efficiency and accuracy in downstream table understanding tasks.

3.2 Cross-Encoder Reranker and Generation

In table reranking stage, we combine the cross-encoder reranking step and context-rich generation step. We use multimodal LLMs f_θ in this stage and unified reranking and generation tasks into one training stage. Existing MLLMs struggle to effectively process and reason over multiple images, particularly in the context of table understanding. They also lack the ability to assess the relevance of a given image to a specific query, often leading to poor performance when the image is unrelated or uninformative.

The goal of this stage comprises two parts: first, to fine-tune the MLLMs to identify the most relevant table images for a given query; and second, to enable the models to generate accurate answers in context-rich scenarios involving multiple table images.

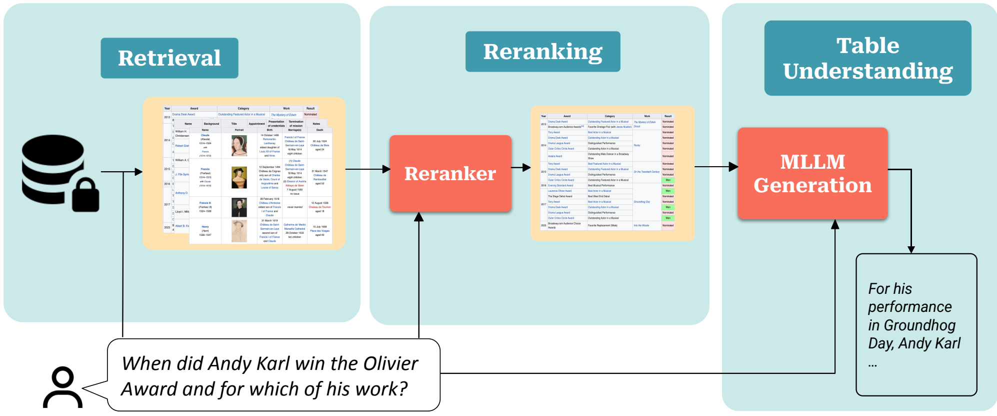


Figure 1: The **TabRAG** framework, which consists of a retriever, a reranker and a MLLM. Once receiving the general query, retriever identifies the relevant tables. Once having the subset of tables, reranker model will rank the relevance of each image with the query, and select the best ones, then MLLM will take the images selected by reranker and query as input, generate the final results.

Task	**Question** x	**Context** c	**Answer** y
Retrieval-augmented QA	Answer the following question from context. `{question}`	Passage 1: `{Image 1}`... Passage 2: `{Image 2}`	phrase/sentence
Context ranking	For the question `{question}`, access whether the passage is relevant to the question.	Passage: `{Image}` (1 Psg.)	True/False
Retrieval-augmented ranking	For the question `{question}`, find all passages from the context that are relevant to the question.	Passage 1: `{Image 1}`... Passage 2: `{Image 2}`	Passage Indexes

Table 1: The instruction template for Cross-Encoder Reranker and Generation.

We follow the instruction tuning template in Yu et al. (2024) and combine different tasks into training stage, as shown in Table 1. **Retrieval-augmented QA**: For each query with the answer, we combine the ground truth table with the top-retrieved tables by retrievers, aiming to enhance the model's capability of robustly generating answers with multiple tables. **Context ranking**: We combine positive pairs of query and table as relevant while random choosing other tables from top-retrieved tables from retrievers, combine it with query as hard negative pairs. We assign positive pair as `True` and negative pair as `False`. We train the MLLMs to determine whether the given table is relevant to the query. **Retrieval-augmented ranking**: We aim to train the LLM with the capability of determining the relevance of multiple contexts simultaneously given a question, which is closer to the test-time behavior of retrieved information with top-k tables, we combine the ground truth table with the top retrieved tables by retriever. The contexts containing the answer are considered relevant, and the MLLMs is trained to explicitly identify all relevant tables for the question. We show the instruction tuning template in Table 1.

We unified the tasks together using standardized QA format (q, c, y), where q is the user query, c is concatenation of table images $(x_1, \ldots, x_k)$, y is answer. We combine the query and images together as prompt, feed into instruction tuning pipelines, training our MLLMs with causal language modeling loss:

$$\mathcal{L}(\theta) = -\sum_{i=1}^{|y|} \log p_\theta \left(y_i \mid \hat{y}_{1:i-1}, q, x_1, \ldots, x_k \right)$$

where θ are trainable parameters in MLLM, $\hat{y}_{1:i-1}$ is the $i-1$ preceding tokens of output y_i.

4 Experiment

In this section, we present our experimental setup and results. We first describe the datasets and models used in our empirical evaluation in Section 4.1. We then report results for the retrieval stage (Section 4.2), reranking stage (Section 4.3), and final answer generation (Section 4.4). Across all stages, our proposed framework consistently outperforms existing baselines and prior methods.

4.1 Dataset and Models

We aim to build up multimodal table datasets for our retrieval and generation purposes. We collected several datasets from public table datasets and then prune and clean the datasets for our purposes. We store all table images in a large datastore and format the query.

We mainly adopt the MMTab dataset in Table-LLaVA (Zheng et al., 2024), which is a collection of 14 public table datasets, covering 8 domains. The original tables in these public datasets are stored in divergent textual formats such as HTML or Markdown. Zheng et al. (2024) convert textual tables into high quality table images. The task-specific input and output texts are transformed into the instruction following format. To minimize errors during answering parsing, they added extra instructions, requiring models to output the final answer in the JSON format. The rendered table images and processed input-output pairs constitute the final multimodal instruction-tuning samples with a unified format of `<table image, input request, output response>`. See details about original MMTab datasets in Appendix A.

For retrieval purposes, our goal is to identify queries that include basic information or metadata related to the table title or topic. The original MMTab dataset contains many general-purpose queries, such as *Count the number of rows or columns in the table* or *Generate a descriptive sentence about the highlighted cells in the provided table*. While these queries are useful for evaluating table or cell-level understanding when the relevant table is provided, they are not suitable for context retrieval settings. In retieval setting, systems are required to locate the correct tables or background knowledge themselves, and such generic queries lack the specificity required to guide effective retrieval.

To fix this problem, we created a careful filtering system using regular expressions to identify and remove generic questions. We identified recurring patterns such as *count the number of rows* or *describe the cells* that are commonly found in general table operations. Following these automated filters, we performed manual checks on the remaining queries. This two-step filtering enabled us create a high-quality dataset that properly challenges a system's ability to retrieve relevant tables. The result was a more targeted collection that better demonstrates how well our retrieval approach works with real information-seeking questions. Through this filtering process, we obtained 88K training samples and 9K testing samples. We conduct filtering separately on training data and testing data to prevent data leakage. We provide detailed statistics about our dataset in Table 6 in Appendix A.

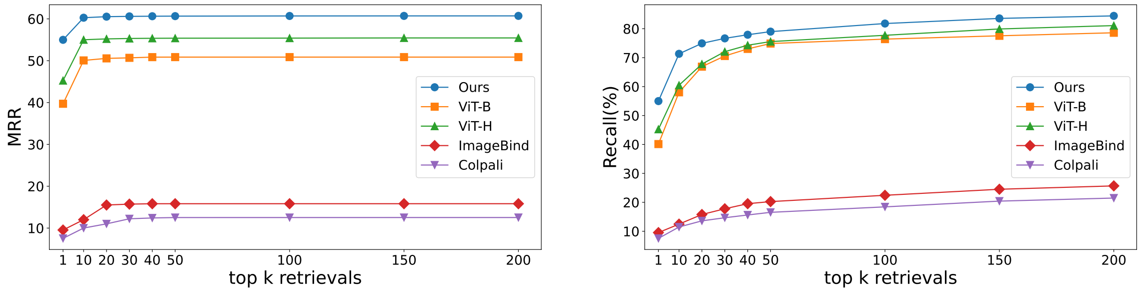
In first stage where we train retrievers, we need to obtain good vision and text encoders to get embedding of table images and user queries. To get good embedding of queries, we need query to be consice to contain key information of the table. The original queries contains formating instructions like *Show your answer in the JSON format answer: [a list of answer strings]*. We further prune the user queries by removing redundant text for better embedding representation.

For retrievers finetuning, we use LayoutLMv3 (Huang et al., 2022) as vision encoder and General Text Embedding (GTE) Models (Zhang et al., 2024) as text encoders. We finetune the encoders using contrastive loss following the CLIP (Radford et al., 2021) model pipeline. For rerankers finetuning, We use Mistral-7b (Jiang et al., 2023) as the LLM backbone and CLIP-ViT-L-14-336px as the visual encoder. We follow the training and instruction finetuning pipeline in Liu et al. (2023b,a); Zheng et al. (2024).

4.2 Retrieval Stage

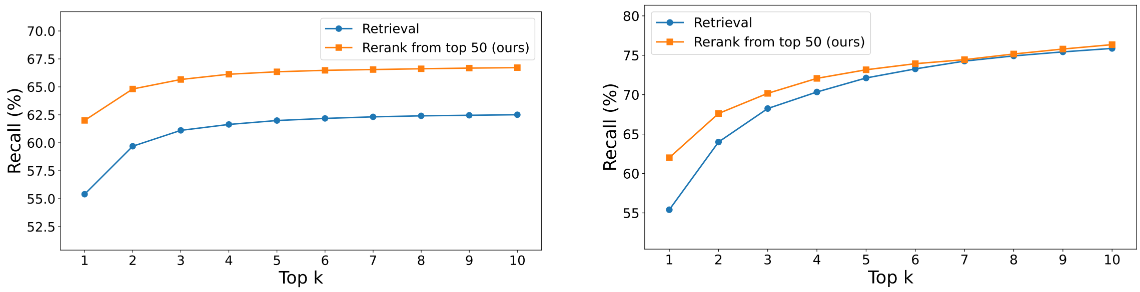
Setup. We trained our own multimodal retriever by combining LayoutLMv3 (Huang et al., 2022) for visual encoding and General Text Embedding (GTE) Models (Zhang et al., 2024) for textual encoding. For comparison, we implemented zero-shot retrieval using other multimodal encoders that align images and text in a shared space, including CLIP (with ViT-H and ViT-B vision encoders) (Radford et al., 2021), ImageBind (Girdhar et al., 2023), and Colpali (Faysse et al., 2024). Our retrievers were fine-tuned on training data and evaluated on test data following Section 3.1. We train on an 8 GPU setup with data parallelism, a learning rate of 2e-5 with cosine decay with 100 warmup steps, and a batch size of 32. We use the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.98$. During evaluation, we use Mean Reciprocal Rank (MRR) and recall as metrics. We assessed performance for top k retrievals, where $k = [1, 10, 20, 30, 40, 50, 100, 150, 200]$.

Results. Figure 2 presents a comprehensive comparison of retrieval performance between our fine-



(a) MRR. (b) Recalls.

Figure 2: Retrieval results of different encoders on RAGTab dataset, different curves represents different models. The graphs illustrate both Mean Reciprocal Rank (MRR) and Recall metrics across various top-k retrievals ranging from 1 to 200. (a) The MRR metric on top k results. (b) The Recall metric on top k results.



(a) MRR. (b) Recalls.

Figure 3: Retrieval results of different encoders on RAGTab dataset, different curves represents only retrievals and retrieval and reranking stage. (a) The MRR metric on top k results. (b) The Recall metric on top k results.

tuned retrievers and other encoder models on the TabRAG dataset described in Table 6. Our fine-tuned retrievers consistently outperform all other models across both MRR and Recall metrics, maintaining a significant lead throughout the entire range of k values. Among the baseline models, CLIP encoders show respectable performance, with the ViT-H vision encoder slightly edging out ViT-B on both metrics. ImageBind and Colpali encoders exhibit notably lower retrieval performance compared to the CLIP variants and our model. Their performance, while improving with increasing k, remains substantially below that of the other approaches. We also noticed the performance gains for all models tend to plateau as k increases, with the most significant improvements occurring in the range of 1 to 50 retrievals.

4.3 Reranking Stage

Setup. We trained our multimodal LLM backbone as our reranker. For reranking training, we designed three tasks including Retrieval-augmented QA (RQA), Context reranking, and Retrieval-augmented reranking (RAR). We first split TabRAG dataset train set into three partition with proportion of 20%, 50%, 30% respectively. Following the procedure in Section 3.2, For RQA and RAR, each query was paired with its positive image and one negative image selected from the top 50 retrieved by our fine-tuned retrievers. For context reranking, we created multiple pairs per query: one positive pair (query with ground truth image) and 20 negative pairs (query with images from top 50 retrieved negatives). See dataset details in Appendix A.1. Our training pipeline involves different steps. Initially, we adopted the pretraining and instruction-finetuning approach in (Liu et al., 2023b), utilizing Mistral-7b and CLIP ViT-L as components in Llava models. We then get our TabRAG-mistral-7b model. See details in Appendix B.1. Subsequently, we further refined TabRAG-mistral-7b using our custom reranking dataset. Additional training details are provided in Appendix B.1.

Results. Figure 3 illustrates a comprehensive comparison between our fine-tuned retrievers and the performance after applying our finetuned Language MLLM rerankers on the TabRAG dataset described in Table 6. The graphs depict both Mean Reciprocal Rank (MRR) and Recall metrics for top k results, where k ranges from 1 to 10. The reranking stage demonstrates consistent and significant improvements in retrieval performance across both

Method	LLM	Question Answering			Fact Verification	Text Generation			
		WTQ	HiTab	FeTaQA	TabFact	ToTTo	HiTab_T2T	Rotowire	WikiBIO
		Acc.	Acc.	BLEU	Acc.	BLEU	BLEU	BLEU	BLEU
LMM									
BLIP2	Flan-T5 7B	2.01	1.52	2.34	18.62	4.3	2.63	1.08	0.72
MiniGPT-4	Vicuna 7B	0.9	0.2	0.39	0	0.2	0.11	1.26	0.33
LLaVA v1.5	Vicuna-1.5 7B	1.24	2.03	8.24	18.9	6.40	2.07	1.92	2.34
Vary-toy	Qwen 1.8B	7.96	3.42	2.44	6.33	0.70	0.27	0.46	0.37
Monkey	Qwen 7B	19.07	6.41	3.41	22.56	3.50	1.12	0.03	2.77
LLaVA v1.6	Vicuna-1.5 7B	1.04	3.57	7.52	19.26	5.87	1.89	2.04	1.84
Table-LLaVA 7B	Vicuna-1.5 7B	18.43	10.09	**25.60**	**59.85**	23.00	9.74	**10.46**	**9.68**
LLM									
Llama2+text	Llama-2 7B	4.26	1.21	5.54	4.21	6.20	1.84	4.67	1.33
TableLlama+text	Llama-2 7B	31.63	64.71	39.05	82.55	20.77	0.19	0.13	0.39
Ours									
Re-Table-7B-retrieval	Mistral 7B	17.19	12.96	14.36	50.64	50.3	14.84	6.11	1.81
Re-Table-7B-rerank	Mistral 7B	**19.19**	**19.49**	23.14	56.67	**52.28**	**16.96**	8.15	4.79

Table 2: Evaluation results TabRAG datasets. '+*text*' represents that the **OCR** textual table representations are provided to LLMs. **Re-Table**-7B-retrieval refers to a model employing retrieval with images obtained through retrievers, **Re-Table**-7B-rerank denotes a model that generates results using retrieval and reranking, with images given by the reranking process.

MRR and recall metrics. For MRR in Figure 3a, our reranking method shows a significant improvement, starting at approximately 62% for Top 1 and reaching about 66.5% for Top 10. In contrast, the baseline retrieval method begins at around 55% for Top 1 and plateaus at about 62.5% for Top 10. The performance gap is particularly pronounced for lower k values, highlighting the effectiveness of our approach in improving ranking quality. Similarly, for Recall in Figure 3b, our reranking method demonstrates superior performance, beginning at about 62% for Top 1 and climbing to approximately 76% for Top 10. The baseline retrieval method starts lower at about 55% for Top 1 and reaches around 75% for Top 10, narrowing the gap at higher k values but still trailing our method. This enhancement is particularly pronounced for smaller values of k, indicating that reranking is most impactful when considering the top few results.

4.4 Generation Results.

Setup. In our evaluation we evaluated our fine-tuned checkpoint under zero-shot settings without additional demonstrations. We leveraged results from (Zheng et al., 2024) for prior LLMs and MLLMs on the MMTab dataset. While baseline models in related work were provided with the query alongside the ground truth table as input, our method differs by using retrieved tables through the TabRAG pipeline instead. For each query, we created different distinct input pairs: the query with the retrieved image and the query with the reranked image. These pairs were then fed into our trained TabRAG-mistral-7b MLLM, allowing us to assess the model's performance across various image selection strategies and compare the effectiveness of our retrieval and reranking mechanisms.

Results. Table 2 presents the comprehensive performance of our pipeline with previous MLLMs and LLMs. We present results for our method using two configurations: one with images retrieved by our retriever model, and another with the top-ranked image after the reranking stage. As shown in Table 2, we see LLaVA-1.6 does not significantly outperform LLaVA-1.5 on the MMTab dataset, suggesting that general improvements in multimodal models may not always translate to better performance on specialized tasks like table understanding. Our TabRAG model consistently performs better with reranked images compared to retrieved images across all tasks. This highlights the importance of the reranking stage in refining the selection of relevant visual information. In several tasks, our TabRAG model with reranking outperforms other LLMs and MLLMs, even when they are provided with the ground truth table. This is particularly notable for tasks like `ToTTo` with BLEU score of 52.28 and `HiTab_T2T` with BLEU score of 16.96. Our model shows significant improvements in text generation tasks such as `WTQ`, `HiTab` and `TabFact` compared to most other models, suggesting it's particularly effective at synthesizing information from tables into coherent text. Overall, the TabRAG model demonstrates robust performance

LMM	Retrieval Method	Question Answering			Fact Verification	Text Generation			
		WTQ	HiTab	FeTaQA	TabFact	ToTTo	HiTab_T2T	Rotowire	WikiBIO
		Acc.	Acc.	BLEU	Acc.	BLEU	BLEU	BLEU	BLEU
Table-LLaVA 7B	random	9.57	5.61	14.08	56.37	3.29	6.55	6.87	2.04
	retrieval	18.76	10.67	16.09	57.27	21.88	3.97	9.10	2.49
	rerank	20.20	11.18	18.18	58.39	22.08	4.86	8.59	8.31
	gold†	18.43	10.09	25.60	59.85	23.00	9.74	10.46	9.68
Re-Table-7B	random	1.27	0.40	1.84	43.96	2.02	2.13	0.17	0.77
	retrieval	17.19	12.96	14.36	50.64	50.3	14.84	6.11	1.81
	rerank	19.19	19.49	23.14	56.67	52.28	16.96	8.15	4.79
	gold	23.00	22.45	43.24	63.85	47.70	20.42	8.69	15.35

Table 3: Ablation results TabRAG datasets using Table-LLaVA 7B. † denotes results from original papers. **Re-Table**-7B-retrieval refers to a model employing retrieval with images obtained through retrievers, **Re-Table**-7B-rerank denotes a model that generates results using retrieval and reranking, with images given by the reranking process.

across diverse tasks, from question answering to text generation, indicating its versatility in handling various table-related challenges.

4.5 Ablation study

While our framework outperform existing work, we try to investigate whether our framework's superior performance stems from the pretrained model or our context enhancement. We perform extensive ablation experiments on the same model released by Table-LLaVA (Zheng et al., 2024) and our trained Re-Table-Mistral-7B, validating the effectiveness of our proposed strategy.

We compare the generation performance of model across various tasks under four distinct conditions when given a user query: (1) Generation using a randomly selected table from the data store. (2) Generation using a table retrieved by our trained bi-encoder retriever in retrieval stage. (3) Generation using tables refined by our fine-tuned MLLM reranker in reranking stage. (4) Generation using the ground truth (golden) table.

Our results in Table 3 demonstrate the efficacy of our trained retrievers and reranking approach across diverse tasks. For the Re-Table-Mistral 7B model, we observed substantial improvements over random table selection. The retrieval stage significantly boosted performance by increasing BLEU scores from 1.84 to 14.36 in `FeTaQA` and from 2.13 to 14.84 in `HiTab_T2T`. The reranking step further enhanced these gains, with notable improvements such as `WTQ` accuracy rising from 17.19% to 19.19%, and `ToTTo` BLEU scores increasing from 50.3 to 52.28. Remarkably, our approach achieved performance remarkably close to ground truth baselines. In `Rotowire`, our reranking method attained 8.15% accuracy compared to 8.69% with gold standard tables. For `HiTab_T2T`, we achieved a BLEU score of 16.96 versus 20.42 with ground truth tables, while in another `ToTTo` evaluation, our method exceeded the gold standard with a BLEU score of 52.28 compared to 47.7. These consistent improvements across question answering, fact verification, and text generation tasks underscore the effectiveness of our proposed strategy.

Our results demonstrate the superior performance of our three-step context enhancement strategy. Our trained retrievers effectively identify relevant subsets of tables, substantially outperforming random table selection. The reranking step further enhances performance. Our approach's performance closely approximates that achieved when using the ground truth table, all within the same experimental framework.

4.6 Computational cost

We provide an analysis of the computational cost of our framework to demonstrate its inference efficiency. Our pipeline is designed to explicitly limit expensive operations: **Stage 1**: Bi-encoder retrieval uses a trained vision and text encoder with pre-built vector index over table-image embeddings. At inference, we only perform a single query encoding and approximate nearest-neighbour search, which is standard and efficient for large-scale retrieval. **Stage 2**: MLLM reranking is run only on the top-k candidates (where k is small). This requires a fixed number of forward passes without autoregressive decoding, which is substantially cheaper than full answer generation. **Stage 3**: Generation calls the MLLM once using the top-ranked tables.

We have measured end-to-end latency and FLOPs on a single NVIDIA A100-40GB GPU with batch size 1. We provide computational details for each module, excluding image embeddings as they are pre-stored in the database. We set $k = 10$:

Stage	Component	Latency (ms)	Memory (GB)	FLOPs
Retrieval	Query encoding (GTE-large)	22	1.7	0.056T
	FAISS search (15k tables)	35	1.8	–
	Subtotal	**57**	**3.5**	**0.056T**
Reranking	MLLM scoring (top-10)	$10 \times 81 = 810$	7.8	$10 \times 4.3\text{T} = 43\text{T}$
	Subtotal	**810**	**7.8**	**43T**
Generation	MLLM generation (top-1 table)	520	7.8	8.6T
	Subtotal	**520**	**7.8**	**8.6T**
Total (Ours)		**1,387**	**7.8**	**51.7T**
Baseline	MLLM with gold table	520	7.8	8.6T

Table 4: Computational cost breakdown for each pipeline stage.

As predicted, generation dominates inference time for one image, but reranking adds non-trivial overhead since we need to rank 10 retrieved table images. However, this latency increase yields substantial accuracy gains (+7.0% retrieval recall, +6.1% answer accuracy). We therefore conclude an accuracy-efficiency trade-off, depending on the number k we choose. We also observe memory efficiency, where peak memory usage (7.8GB) fits comfortably on a single GPU. The FAISS index can be memory-mapped to reduce RAM usage.

5 Conclusion

We present TabRAG, a novel framework for context enhancement on multimodal table understanding. We directly process table images, our three-stage approach - retrieval, reranking, and generation - addresses key challenges in handling tabular data within LLMs. TabRAG demonstrates significant improvements over existing methods, enhancing the identification of relevant tables from large collections and leveraging multimodal LLMs to generate accurate responses to user queries. This work not only advances the field of information retrieval but also opens up new possibilities for more intuitive and efficient solutions tabular understanding in AI systems. Future directions including extend tabular to various document types, from structured reports to freeform handwritten notes, as well as general images spanning photographs, also extend visual data to other modalities such as text in various formats (e.g. markup languages), audio recordings, and video content.

Ethical Considerations

Our paper is mostly empirical in nature and we foresee no immediate negative ethical impact. Our work aims to advance the multimodal reasoning field with a focus on real-world table understanding problems. By building upon open-source datasets, we ensure transparency and accessibility to the underlying data. In the long term, we hope our work may inspire effective algorithm design and better understanding and employment of LLMs.

Limitation

Our work presents several limitations that warrant consideration. We acknowledge that our approach has not been thoroughly tested on real-world scanned table images extracted from documents or scraped from the web, which often contain distortions or complex layouts. This limits our ability to conclusively demonstrate the framework's effectiveness in more challenging scenarios.

References

Mubashara Akhtar, Oana Cocarascu, and Elena Simperl. 2022. PubHealthTab: A public health table-based dataset for evidence-based fact checking. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1–16, Seattle, United States. Association for Computational Linguistics.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, and 1 others. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf.

Akari Asai, Sewon Min, Zexuan Zhong, and Danqi Chen. 2023a. Retrieval-based language models and applications. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*

(Volume 6: Tutorial Abstracts), pages 41–46, Toronto, Canada. Association for Computational Linguistics.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023b. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*.

Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. 2022. Deep neural networks and tabular data: A survey. *IEEE transactions on neural networks and learning systems*.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020. Tabfact: A large-scale dataset for table-based fact verification. *Preprint*, arXiv:1909.02164.

Yixin Chen, Shuai Zhang, Boran Han, Tong He, and Bo Li. 2024. Camml: Context-aware multimodal learner for large models. *arXiv preprint arXiv:2401.03149*.

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.

Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. 2020. Turl: table understanding through representation learning. *Proceedings of the VLDB Endowment*, 14(3):307–319.

Tuan Dinh, Yuchen Zeng, Ruisu Zhang, Ziqian Lin, Michael Gira, Shashank Rajput, Jy-yong Sohn, Dimitris Papailiopoulos, and Kangwook Lee. 2022. Lift: Language-interfaced fine-tuning for non-language machine learning tasks. *Advances in Neural Information Processing Systems*, 35:11763–11784.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library.

Xi Fang, Weijie Xu, Fiona Anting Tan, Jiani Zhang, Ziqing Hu, Yanjun Qi, Scott Nickleach, Diego Socolinsky, Srinivasan Sengamedu, and Christos Faloutsos. 2024. Large language models (llms) on tabular data: Predic-tion, generation, and understanding-a survey. *arXiv preprint arXiv:2402.17944*.

Manuel Faysse, Hugues Sibille, Tony Wu, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2024. Colpali: Efficient document retrieval with vision language models. *arXiv preprint arXiv:2407.01449*.

Raul Castro Fernandez, Aaron J Elmore, Michael J Franklin, Sanjay Krishnan, and Chenhao Tan. 2023. How large language models will disrupt data management. *Proceedings of the VLDB Endowment*, 16(11):3302–3309.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. 2023. Imagebind: One embedding space to bind them all. In *CVPR*.

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.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. Tabllm: Few-shot classification of tabular data with large language models. In *International Conference on Artificial Intelligence and Statistics*, pages 5549–5581. PMLR.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*.

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

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, and 1 others. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Yannis Katsis, Saneem Chemmengath, Vishwajeet Kumar, Samarth Bharadwaj, Mustafa Canim, Michael Glass, Alfio Gliozzo, Feifei Pan, Jaydeep Sen, Karthik Sankaranarayanan, and Soumen Chakrabarti. 2021. Ait-qa: Question answering dataset over complex tables in the airline industry. *Preprint*, arXiv:2106.12944.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. 2022. Ocr-free document understanding transformer. In *European Conference on Computer Vision*, pages 498–517. Springer.

Siddique Latif, Moazzam Shoukat, Fahad Shamshad, Muhammad Usama, Yi Ren, Heriberto Cuayáhuitl, Wenwu Wang, Xulong Zhang, Roberto Togneri, Erik Cambria, and 1 others. 2023. Sparks of large audio models: A survey and outlook. *arXiv preprint arXiv:2308.12792*.

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.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. 2024. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 2024. Video-LLaVA: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5971–5984, Miami, Florida, USA. Association for Computational Linguistics.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023a. Improved baselines with visual instruction tuning.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. In *NeurIPS*.

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

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-4 technical report. *arXiv preprint arxiv:2303.08774*.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186, Online. Association for Computational Linguistics.

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

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Ananya Singha, José Cambronero, Sumit Gulwani, Vu Le, and Chris Parnin. 2023. Tabular representation, noisy operators, and impacts on table structure understanding tasks in llms. *arXiv preprint arXiv:2310.10358*.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024. Table meets llm: Can large language models understand structured table data? a benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 645–654.

Boris van Breugel and Mihaela van der Schaar. 2024. Why tabular foundation models should be a research priority. *arXiv preprint arXiv:2405.01147*.

Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. 2024. Chain-of-table: Evolving tables in the reasoning chain for table understanding. In *The Twelfth International Conference on Learning Representations*.

Xumeng Wen, Han Zhang, Shun Zheng, Wei Xu, and Jiang Bian. 2024. From supervised to generative: A novel paradigm for tabular deep learning with large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3323–3333.

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.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, and 1 others. 2024. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*.

Chao Ye, Guoshan Lu, Haobo Wang, Liyao Li, Sai Wu, Gang Chen, and Junbo Zhao. 2023. Ct-bert: learning better tabular representations through cross-table pre-training. *arXiv preprint arXiv:2307.04308*.

Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. Tabert: Pretraining for joint understanding of textual and tabular data. *arXiv preprint arXiv:2005.08314*.

Yue Yu, Wei Ping, Zihan Liu, Boxin Wang, Jiaxuan You, Chao Zhang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Rankrag: Unifying context ranking with retrieval-augmented generation in llms. *arXiv preprint arXiv:2407.02485*.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, and 1 others. 2024. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. *arXiv preprint arXiv:2407.19669*.

Yunjia Zhang, Jordan Henkel, Avrilia Floratou, Joyce Cahoon, Shaleen Deep, and Jignesh M Patel. 2023. Reactable: Enhancing react for table question answering. *arXiv preprint arXiv:2310.00815*.

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.

Appendix

A Datasets

In Table 6, we present the details of datasets curated from (Zheng et al., 2024), which we refined for both retrieval and generation purposes. The full dataset in (Zheng et al., 2024) is shown in Table 7.

A.1 Reranking and generation datasets

We show our reranking training datasets in Table 5.

	Train	Val
Retrieval-augmented QA (RQA)	13K	1K
Context reranking	854K	87K
Retrieval-augmented reranking (RAR)	20K	2K

Table 5: Reranking and Generation training and testing data.

B Experimental details

B.1 Train TabRAG-mistral-7b

We increased the max sequence length from 2048 to 2560 to accommodate longer text sequences. We first pretrain and finetuned the RAGTab-mistral-7b model following the pipeline in (Liu et al., 2023b), We pre-train our model on the pretraine dataset in (Zheng et al., 2024) for 1 epoch with a learning rate of 1e-3 and a batch size of 32. We fine-tune on the finetune dataset in (Zheng et al., 2024) for 3 epochs, with a learning rate of 2e-5 and a batch size of 32. We use the Adam optimizer with no weight decay and a cosine learning rate with a warmup ratio of 1%.

This fine-tuning phase lasted for 3 epochs, employing a learning rate of 2e-5 and a batch size of 16. We utilized the Adam optimizer without weight decay and implemented a cosine learning rate schedule with a 1% warmup ratio.

Task Category	Task Name	Dataset	Table Style	Domain	# Tables		# Samples	
					Train	Test	Train	Test
Table Question Answering (TQA)	Flat TQA	WTQ (2015)	W	Wikipedia	1.6K	0.4K	17K	4K
	Free-form TQA	FeTaQA (2022)	W	Wikipedia	8K	2K	8K	2K
	Hierarchical TQA	HiTab (2022)	E	Wikipedia & gov. reports	3K	0.5K	8K	1.5K
Table Fact Verification	TFV	TabFact (2020)	E, M	Wikipedia	9K	1K	31K	6.8K
Table to Text (T2T)	Cell Description	ToTTo (2020)	W	Wikipedia	15K	7.7K	15K	7.7K
		HiTab_T2T (2022)	E	Wikipedia & gov. reports	3K	1.5K	3K	1.5K
	Game Summary	Rotowire (2017)	E	NBA games	3.4K	0.3K	3.4K	0.3K
	Biography Generation	WikiBIO (2016)	E	Wikipedia	4.9K	1K	4.9K	1K
Total					48 K	15K	88K	9K

Table 6: Breakdown statistics of the TabRAG dataset. W, E and M represents Web page, Excel, and Markdown tables, respectively.

MMTab	Task Category	Task Name	Dataset	Table Style	Domain	Held-in	# Tables		# Samples		Avg. Length
							Train	Test	Train	Test	(input/output)
MMTab-instruct	Table Question Answering (TQA)	Flat TQA	WTQ (2015)	W	Wikipedia	Yes	1.6K	0.4K	17K	4K	45.9/10.4
		Free-form TQA	FeTaQA (2022)	W	Wikipedia	Yes	8K	2K	8K	2K	32.3/18.69
		Hierarchical TQA	HiTab (2022)	E	Wikipedia goverment reports	Yes	3K	0.5K	8K	1.5K	63.5/12.6
			AIT-QA (2021)	E	Airline	No	-	0.1K	-	0.5K	41.8/10.2
		Multi-choice TQA	TabMCQ (2016)	M	science exams	No	-	0.05K	-	1K	47.9/13.2
		Tabular Numerical Reasoning	TABMWP (2023)	W	math exams	Yes	30K	7K	30K	7K	54.2/51.9
			TAT-QA (2021)	M	financial reports	Yes	1.7K	0.2K	5.9K	0.7K	40.1/16.5
	Table Fact Verification (TFV)	TFV	TabFact (2020)	E, M	Wikipedia	Yes	9K	1K	31K	6.8K	49.9/18.3
			InfoTabs (2020)	W	Wikipedia	Yes	1.9K	0.6K	18K	5.4K	54.2/18.6
			PubHealthTab (2022)	W	public health	No	-	0.3K	-	1.9K	71.9/18.4
	Table to Text (T2T)	Cell Description	ToTTo (2020)	W	Wikipedia	Yes	15K	7.7K	15K	7.7K	31.1/14.8
			HiTab_T2T (2022)	E	Wikipedia goverment reports	Yes	3K	1.5K	3K	1.5K	39.1/14.7
		Game Summary	Rotowire (2017)	E	NBA games	Yes	3.4K	0.3K	3.4K	0.3K	27.6/291.7
		Biography Generation	WikiBIO (2016)	E	Wikipedia	Yes	4.9K	1K	4.9K	1K	18.1/84.2
	Total						82K	-	232K	-	66.1/66.9
MMTab-eval	Total						-	23K	-	49K	46.3/32.6

Table 7: Breakdown statistics of the constructed **MMTab** dataset. W, E and M represents Web page, Excel, and Markdown tables, respectively.