VDocRAG: Retrieval-Augmented Generation over Visually-Rich Documents

Ryota Tanaka[1,2] Taichi Iki[1] Taku Hasegawa[1] Kyosuke Nishida[1] Kuniko Saito[1] Jun Suzuki[2]
[1]NTT Human Informatics Laboratories, NTT Corporation [2]Tohoku University
https://vdocrag.github.io

Abstract

We aim to develop a retrieval-augmented generation (RAG) framework that answers questions over a corpus of visually-rich documents presented in mixed modalities (e.g., charts, tables) and diverse formats (e.g., PDF, PPTX). In this paper, we introduce a new RAG framework, VDocRAG, which can directly understand varied documents and modalities in a unified image format to prevent missing information that occurs by parsing documents to obtain text. To improve the performance, we propose novel self-supervised pre-training tasks that adapt large vision-language models for retrieval by compressing visual information into dense token representations while aligning them with textual content in documents. Furthermore, we introduce OpenDocVQA, the first unified collection of open-domain document visual question answering datasets, encompassing diverse document types and formats. OpenDocVQA provides a comprehensive resource for training and evaluating retrieval and question answering models on visually-rich documents in an open-domain setting. Experiments show that VDocRAG substantially outperforms conventional text-based RAG and has strong generalization capability, highlighting the potential of an effective RAG paradigm for real-world documents.

1. Introduction

Large language models (LLMs) have demonstrated impressive performance on diverse natural language tasks [2, 16, 24, 55]. These models struggle with factual errors despite their increased model and data scale [39, 40]. To remedy this problem, retrieval-augmented generation (RAG) methods [18, 31] can retrieve knowledge from an external corpus, potentially reducing hallucination and increasing knowledge coverage. Most previous RAG frameworks assume the context is composed entirely of text, with no graphical elements. In contrast, a significant amount of real-world information is stored in visually-rich documents, such as charts, tables, web pages, and office documents. These documents often contain both textual and visual objects, with content spread structurally across various locations depending on diverse formats and types.

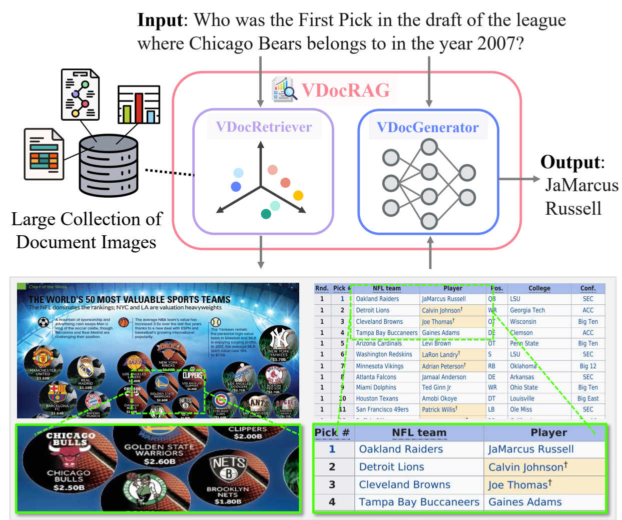


Figure 1. Our framework of VDocRAG and examples from OpenDocVQA. VDocRAG consists of VDocRetirver and VDocGenerator, which can retrieve relevant documents and generate answers by understanding the original appearance of documents.

Thus, document visual question answering (DocumentVQA) [42, 43, 56, 57] aims to build an agent capable of reading and comprehending document images to answer the question. Here, most existing DocumentVQA questions operate in a closed setting without requiring any retrieval. While this definition simplifies the QA model, it does not reflect many real-world use cases where the question is asked through some open-domain natural language interface, such as QA systems searching information across in-house documents or customer service chatbots on e-commerce websites. To address this limitation, recent works have introduced retrieval tasks on document images [17, 37]. However, these cannot fully develop models that effectively integrate the retrieved information into the final output. This gap hinders the application of DocumentVQA models in more realistic, open-domain scenarios.

In this paper, we introduce a new RAG framework, VDocRAG, which can directly understand varied documents and modalities in a unified image format to avoid tedious parsing and potential information loss that occurs in conventional text-based RAG. As depicted in Figure 1, VDocRAG consists of two main components, both of which effectively leverage the visual features of documents. First, VDocRetriever retrieves document images related to the question from a corpus of document images. Second, VDocGenerator uses these retrieved images to generate the answer. To encode document images and interact with the encoded information, we adapt pre-trained large vision language models (LVLMs) [1, 29] as the backbone for VDocRAG. Since LVLMs are inherently generative models, it is sub-optimal for embeddings as they prevent the representations from capturing information across the entire input sequence due to the training objective (i.e., next-token prediction). To bridge this gap, we introduce new self-supervised pre-training tasks that harness the understanding and generation capabilities of LVLMs to enhance representation learning. Specifically, we compress the entire image representation into a dense token representation, by aligning the text in documents via retrieval and generation tasks.

Furthermore, we introduce OpenDocVQA, the first unified collection of open-domain DocumentVQA datasets encompassing a wide range of document types and formats. OpenDocVQA provides a comprehensive resource for training and evaluating retrieval and question answering models on visually-rich documents in an open-domain setting. Experiments demonstrate that VDocRAG substantially outperforms conventional text-based RAG and has strong generalization performance.

Our main contributions are summarized as follows:

- We introduce a new RAG framework, VDocRAG, which can directly understand diverse real-world documents purely from visual features.
- We are the first to explore pre-training tasks designed for document retrieval-oriented adaptation of LVLMs, by compressing visual document representations.
- We introduce OpenDocVQA, the first unified open-domain DocumentVQA dataset with diverse documents.

2. Related Work

Retrieval-augmented generation (RAG). RAG in the NLP community aims at retrieving external knowledge to reduce factual errors and enhance performance in various knowledge-intensive tasks [3, 5, 39, 40, 49]. Inspired by the success of RAG in NLP, this technique has also applied applications across different domains, including images [8, 50, 51, 64], codes [45, 70], videos [7, 61], audio [26, 62], and 3D [53, 69]. However, most existing works have focused on retrieving knowledge from only plain-text documents or non-text media. In contrast, we tackle the challenge of extracting knowledge from visually-rich documents organized in complex, multimodal formats.

Visual document retrieval and visual RAG. With the success of LLMs, there is a growing trend to build large vision language models (LVLMs) that integrate image understanding capabilities by combining image encoders [32, 48, 67] with LLMs [1, 10, 29, 33, 35, 58]. Concurrent works in visual document retrieval [13, 17, 37] and visual RAG [9, 38, 66] leverage LVLMs to directly encode visually-rich documents through images. However, these approaches have trouble understanding diverse real-world documents due to the limitations of their datasets and training strategies. The existing visual document retrieval dataset, ViDoRe [37], contains questions that might not require retrieval and handle a limited number of document types, resulting in the gap between real-world scenarios. In contrast, our dataset covers open document types and provides questions that are verified by humans to require retrieval and to have context-independent conditions for the retrieval. From the perspective of training, despite the significant gap between generative pre-training tasks and retrieval tasks in LVLMs, previous works [9, 17, 37, 38, 66] leverage LVLMs without specific training for bridging the gap. To address this, we introduce pre-training tasks that transfer the understanding and generation capabilities of LVLMs to retrievers.

Document visual question answering (DocumentVQA). DocumentVQA is a high-level document understanding task that involves answering questions on visually-rich documents. These documents include a variety of elements such as handwritten and digital text [42, 56], complex layouts [28, 68, 71], and graphical elements [41, 43, 57]. However, previous studies have assumed closed settings that do not require retrieval, except for $\text{Dureader}_{\text{vis}}$ [46]. Our work differs from $\text{Dureader}_{\text{vis}}$ as follows. First, OpenDocVQA covers a wide range of document formats and domains, while $\text{Dureader}_{\text{vis}}$ focuses on screenshots of websites, limiting its generalizability. Second, OpenDocVQA reflects more real-world scenarios that require both single- and multi-hop reasoning over documents, while $\text{Dureader}_{\text{vis}}$ requires only single-hop reasoning. Lastly, even lexical search methods yield sufficient performance in $\text{Dureader}_{\text{vis}}$ due to its reliance on textual content. In contrast, OpenDocVQA requires a visual semantic search, where visual and contextual information can be exploited.

3. OpenDocVQA Task and Dataset

3.1. Task Formulation

Given a large collection of N document images $\mathcal{I} = \{I_1, ..., I_N\}$ and a question Q, the goal of OpenDocVQA task is to output an answer A by finding the relevant k images $\hat{\mathcal{I}} \in \mathcal{I}$, where $k \ll N$. We decompose the task into two stages. **Visual document retrieval**: given Q and $\mathcal{I}$,

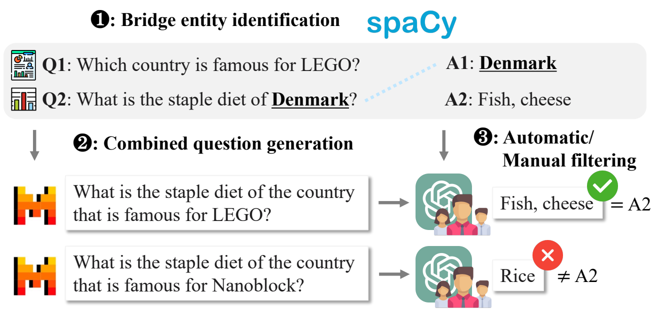


Figure 2. Process of creating multi-hop DocumentVQA questions.

the model retrieves the relevant k images $\hat{\mathcal{I}}$ from which to derive the answer. **DocumentVQA**: the model takes Q and the retrieved images $\hat{\mathcal{I}}$ as input, to generate A.

OpenDocVQA covers multiple open-domain DocumentVQA datasets with diverse document types. To reflect real-world scenarios, we evaluate models with both **single-pool** and **all-pool** settings. In the single-pool setting, retrieval is performed from a specific pool of documents provided by each original dataset. The all-pool setting requires retrieving from the entire candidate pool, which includes documents from a wide range of domains.

3.2. Dataset Collection

Filtering of DocumentVQA datasets. We collected and filtered instances of seven existing document VQA datasets [28, 41–43, 56, 57, 68]. Most of their questions are context-**dependent** conditions, where they cannot be answered without referencing the accompanying document (e.g., *What is the title?*). Therefore, we filtered out questions lacking sufficient context for retrieval. To address this, we initially applied heuristic rules to automatically select likely context-**independent** questions, reducing the pool by 20.9%. Then, we manually reviewed and verified the remaining examples to ensure their context independence.

Reformulation of TableQA dataset. We used QA pairs from Open-WikiTable [27], an open-domain TableQA dataset that required retrieving tables from Wikipedia to answer the question. Since the original dataset provides tables in only textual format (HTML data), we took the screenshot images of tables from the corresponding Wikipedia pages to reformulate the task as the OpenDocVQA.

Creation of new multi-hop questions. To enhance the model's ability to interact with multiple document sources (e.g., charts and tables), we semi-automatically created a multi-hop DocumentVQA dataset, MHDocVQA, using the single-hop QA pairs collected in the previous steps. As shown in Figure 2, the creating process involved the following steps: (1) We first used spaCy [19] to identify a *bridge entity* (e.g., *Denmark*) in the answer to a single-hop question and then searched for this entity in other single-hop questions. (2) Next, we used Mixtral-8x22B [24] to combine the two single-hop questions. (3) We filtered the generated multi-hop questions using another LLM (GPT-4o [2]), which answered the questions based on the context of the two initial single-hop questions and their answers. If the predicted answer was the same as the answer to the second single-hop question, the multi-hop question was validated. Finally, we manually reviewed the filtered questions to ensure their quality before including them in our dataset.

	ViDoRe [17]	Dureader$_{\text{vis}}$ [46]	OpenDocVQA
Retrieval	✓	✓	✓
QA	✗	✓	✓
Context-Independent	✗	✓	✓
Visual Semantic Search	✓	✗	✓
Multi-Hop	✗	✗	✓
Document Contents	T, L, F, C, D	T, L	T, L, F, C, D
Answer Types	–	Ext	Ext, Abs
#Document Types	6	1	Open
#QAs	3,810	15,000	43,474
#Images (Pages)	8,310	158,000	231,339

Table 1. Copmarison of related datasets. Document contents include (T)able, (L)ist, (F)igure, (C)hart, and (D)iagram. Answer types are Extractive (Ext) and Abstractive (Abs).

Negative candidates mining. We produced negative image candidates for retrievers to sift through for every question, used only during inference. We first extracted OCR text from images in the COYO-700M dataset [6], a web-scaled image collection. Subsequently, we mined negative images where the OCR text exhibits high lexical overlap with the question but does not contain the correct answer.

3.3. Comparison with Related Datasets

Table 1 shows the statistics of OpenDocVQA and other related datasets, including ViDoRe [17] and Dureader$_{\text{vis}}$ [46]. OpenDocVQA has three unique key properties: First, it is the first large-scale collection of open-domain DocumentVQA datasets to address open document types, whereas ViDoRe considers six document types for only the retrieval task and Dureader$_{\text{vis}}$ is limited to webpages. Second, the questions in OpenDocVQA are context-independent and require visual semantic search, whereas ViDoRe's questions are context-dependent, and even lexical search methods yield sufficient performance in Dureader$_{\text{vis}}$. This indicates our dataset better reflects real-world scenarios. Lastly, unlike ViDoRe and Dureader$_{\text{vis}}$, OpenDocVQA requires multi-hop reasoning with extractive (e.g., *span, list*) and abstractive (e.g., *arithmetic, counting, no answer*) answer types, providing a more challenging setting.

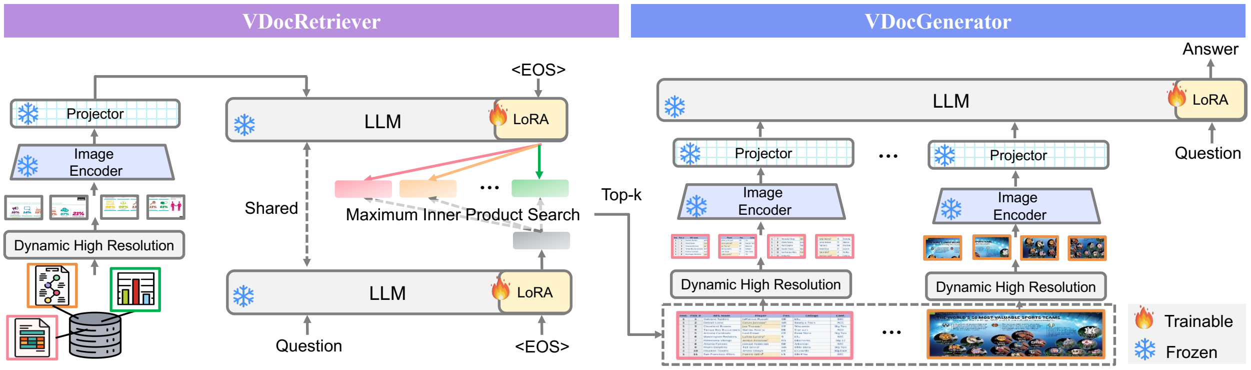


Figure 3. Overview of our VDocRAG model. VDocRetriever retrieves document images related to the question from a corpus of document images, and VDocGenerator uses these retrieved images to generate the answer.

4. Proposed Model

4.1. Architecture Overview

As shown in Figure 3, VDocRAG consists of two components: VDocRetriever and VDocGenerator. Our approach adopts the pre-trained LVLMs to unify the varied formats and modalities in a single form as an image for direct document understanding.

Dynamic high-resolution image encoding. To encode high-resolution images with various aspect ratios, a dynamic cropping [14, 65] is utilized to split the image into smaller patches while maintaining the integrity of the original aspect ratio. Each patch is a small image with 336×336 size and we treat them as individual inputs for the image encoder. After encoding images, we convert them via a projector (two-layer MLP) into visual document features $\mathbf{z}_\mathrm{d}$.

VDocRetriever. VDocRetriever is an LVLM-based dual-encoder architecture that encodes queries and document images independently. We append an `<EOS>` token to the end of the question and visual document features $\mathbf{z}_\mathrm{d}$, and then feed them into the LLM to obtain the question and visual document embeddings ($\mathbf{h}_\mathrm{q}$, $\mathbf{h}_\mathrm{d}$) by taking the last layer `<EOS>` vector. Then, it retrieves k documents $\hat{\mathcal{I}}$ with the k highest similarity scores to the question. Formally, the similarity scores between the question and visual document embeddings are computed via maximum inner product search [15], as follows: $\textsc{sim}(\mathbf{h}_\mathrm{q}, \mathbf{h}_\mathrm{d}) = \frac{\mathbf{h}_\mathrm{q}^\top \mathbf{h}_\mathrm{d}}{\|\mathbf{h}_\mathrm{q}\| \|\mathbf{h}_\mathrm{d}\|}$.

VDocGenerator. VDocGenerator adapts LVLM to generate answers A given the question Q and the retrieved k documents $\hat{\mathcal{I}}$ obtained from VDocRetriever. After encoding the retrieval result, we concatenate the question and the encoded result, then feed this combined input into the LLM.

4.2. Self-Supervised Pre-training Tasks

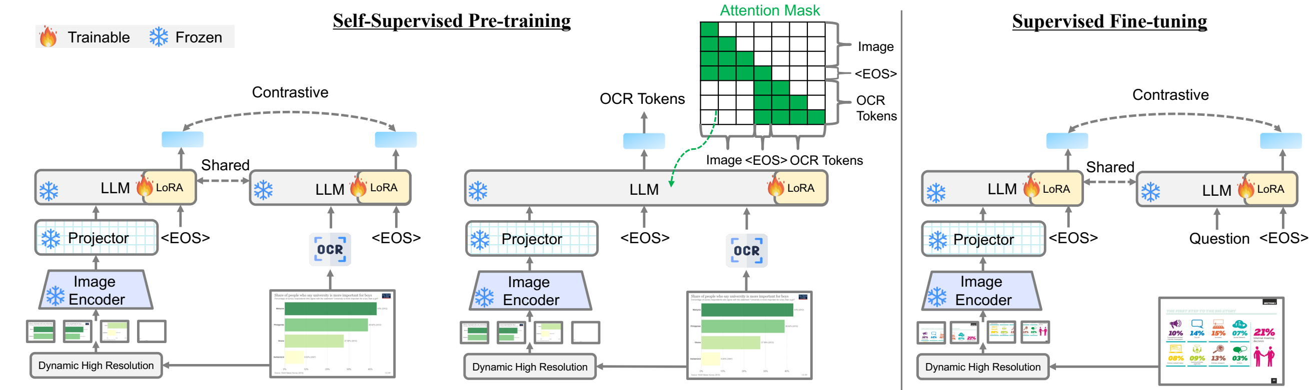
Figure 4a and 4b show our pre-taining tasks in VDocRetriever. The goal of pre-training is to transfer the powerful understanding and generation abilities of LVLMs to facilitate their usage in visual document retrieval. To this end, we propose two new self-supervised pre-training tasks to compress the entire image representation into the `<EOS>` token at the end of the input image. Our pre-training process passes the document image, and its extracted OCR text is used as a pseudo target. Full pre-training objectives is defined as $\mathcal{L} = \mathcal{L}_\mathrm{RCR} + \mathcal{L}_\mathrm{RCG}$.

Representation Compression via Retrieval (RCR). We compress image representations with a contrastive learning task that retrieves images relevant to their corresponding OCR text, by leveraging LVLM's image understanding capabilities. As shown in Figure 4a, we first construct positive OCR text-image pairs ($\mathbf{h}_\mathrm{o}, \mathbf{h}_\mathrm{d+}$) from raw unlabeled document images. Then, we adopt in-batch negatives to calculate the contrastive loss by InfoNCE [44] as follows:

$$\mathcal{L}_\mathrm{RCR} = -\log \frac{\exp(\textsc{sim}(\mathbf{h}_\mathrm{o}, \mathbf{h}_\mathrm{d+})/\tau)}{\sum_{i \in \mathcal{B}} \exp(\textsc{sim}(\mathbf{h}_\mathrm{o}, \mathbf{h}_{\mathrm{d}_i})/\tau)}, \quad (1)$$

where τ is a temperature hyperparameter to scale the logits, and $\mathcal{B}$ represents the batch size.

Representation Compression via Generation (RCG). We propose a representation training strategy that leverages the generative capabilities of LVLMs through a customized attention mask matrix. As depicted in Figure 4b, representations for the image tokens, including the `<EOS>` token, are obtained via a standard auto-regressive process. In contrast, for the subsequent L OCR token representations, we mask the image token representations and allow only the attention of `<EOS>` token and the preceding OCR tokens. This approach facilitates pooling the image representations



(a) Representation Compression via Retrieval (RCR) (b) Representation Compression via Generation (RCG) (c) Visual Document Retrieval

Figure 4. Our pre-training tasks using unlabeled documents and fine-tuning in VDocRetriever. The RCR task retrieves relevant images given corresponding OCR tokens, and the RCG task outputs OCR tokens by paying attention to only the <EOS> token.

Dataset	Documents	%Filtered	#Images	#Train&Dev	#Test
DocVQA [42]	Industry	84.8	12,767	6,382	–
InfoVQA [43]	Infographic	61.2	5,485	9,592	1,048
VisualMRC [56]	Webpage	71.9	10,229	6,126	–
ChartQA [41]	Chart	94.0	20,882	–	150
OpenWikiTable [27]	Table	0.0	1,257	4,261	–
DUDE [28]	Open	92.3	27,955	2,135	496
MPMQA [68]	Manual	81.7	10,435	3,054	–
SlideVQA [57]§	Slide	66.7	52,380	–	760
MHDocVQA§	Open	9.5	28,550	9,470	–

Table 2. Datasets in OpenDocVQA. § denotes datasets requiring multi-hop reasoning. Note that MHDocVQA was created using only the training datasets.

into <EOS> token. The loss function is defined as:

$$\mathcal{L}_{\text{RCG}} = -\frac{1}{L}\sum_{i=1}^{L} \log p(y_i | y_{<i}, \texttt{<EOS>}), \quad (2)$$

where y_i denotes the i-th token of the OCR.

4.3. Supervised Fine-tuning

We first fine-tune the VDocRetriever with the contrastive learning objective using query-document pairs with in-batch negatives (see Figure 4c). Then, we apply the trained VDocRetriever to search over the corpus $\mathcal{I}$ to feed the top-k documents into the VDocGenerator. Finally, we train the VDocGenerator using the next-token prediction objective.

5. Experiments

5.1. Experimental Setup

Pre-training dataset. For pre-training, we gathered 500k unlabeled documents containing image and OCR text pairs filtered from the DocStruct4M [20]. To avoid data contamination, we excluded any images that appear in the test set.

Fine-tuning and evaluation datasets. We evaluated our models in both zero-shot and supervised settings. The zero-shot evaluation assessed the models' generalization capabilities on unseen datasets, while the supervised evaluation measured performance when training samples were available. As shown in Table 2, we trained our models on seven datasets and evaluated them on four datasets, including ChartQA and SlideVQA in the zero-shot setting, and InfoVQA and DUDE in the supervised setting.

Implementation details. We initialized VDocRAG with Phi3V [1], a state-of-the-art LVLM trained on high-resolution images and multi-image data. The parameters of VDocRetriever and VDocGenerator were not shared. We employed LoRA [21] with LLM while keeping other parameters frozen during training. We trained VDocRAG for one epoch on eight A100-80G GPUs with AdamW [36] optimizer and FlashAttention [11], using batch sizes of 16 for pre-training and 64 for fine-tuning. We set the temperature τ to 0.01. We applied Tesseract [54] to extract OCR text in images. By default, we used the top three documents obtained from VDocRetirver.

Retrieval baselines. We compared VDocRetriever with two categories of retrievers. The first category includes off-the-shelf text retrieval models on extracted text and image retrieval models. These consist of **BM25** [52], a lexical matching model; **Contriver** [22], **E5** [59], and **GTE** [34], which are popular strong text embedding models based on BERT [12]; **E5-Mistral** [60] and **NV-Embed-v2** [30], which are state-of-the-art LLM-based embedding models; **CLIP** [47], a dual-encoder vision-language model; **DSE** [37] and **VisRAG-Ret** [66], which are state-of-the-art visual document retrieval models. The second category includes fine-tuned models trained on OpenDocVQA. To

Model	Init	Docs	Scale	#PT	#FT	ChartQA Single	ChartQA All	SlideVQA Single	SlideVQA All	InfoVQA Single	InfoVQA All	DUDE Single	DUDE All
Off-the-shelf													
BM25 [52]	–	Text	0	0	0	54.8	15.6	40.7	38.7	50.2	31.3	57.2	47.5
Contriever [22]	BERT [12]	Text	110M	1B	500K	66.9	59.3	50.8	46.5	42.5	21.0	40.6	29.7
E5 [59]	BERT [12]	Text	110M	270M	1M	74.9	66.3	53.6	49.6	49.2	26.9	45.0	38.9
GTE [34]	BERT [12]	Text	110M	788M	3M	72.8	64.7	55.4	49.1	51.3	32.5	42.4	36.0
E5-Mistral [60]	Mistral [23]	Text	7.1B	0	1.85M	72.3	70.0	63.8	57.6	60.3	33.9	52.2	45.2
NV-Embed-v2 [30]	Mistral [23]	Text	7.9B	0	2.46M	75.3	70.7	61.7	58.1	56.5	34.2	43.0	38.6
CLIP [47]	Scratch	Image	428M	400M	0	54.6	38.6	38.1	29.7	45.3	20.6	23.2	17.6
DSE [37]	Phi3V [1]	Image	4.2B	0	5.61M	72.7	68.5	73.0	67.2	67.4	49.6	55.5	47.7
VisRAG-Ret [66]	MiniCPM-V [63]	Image	3.4B	0	240K	**87.2***	75.5*	74.3*	68.4*	71.9*	51.7*	56.4	44.5
Trained on OpenDocVQA													
Phi3 [1]	Phi3V [1]	Text	4B	0	41K	72.5	65.3	53.3	48.4	53.2*	33.0*	40.5*	32.0*
VDocRetriever†	Phi3V [1]	Image	4.2B	0	41K	84.2$_{+11.7}$	74.8$_{+9.5}$	71.0$_{+17.7}$	65.1$_{+16.7}$	66.8*$_{+13.6}$	52.8*$_{+19.8}$	48.4*$_{+7.9}$	41.0*$_{+9.0}$
VDocRetriever	Phi3V [1]	Image	4.2B	500K	41K	86.0$_{+1.8}$	**76.4**$_{+1.6}$	**77.3**$_{+6.3}$	**73.3**$_{+8.2}$	**72.9***$_{+6.1}$	**55.5***$_{+2.7}$	**57.7***$_{+9.3}$	**50.9***$_{+9.9}$

Table 3. Retrieval results under the single- (Single) and all-pool (All) settings. * indicates performance on test data for which corresponding training samples are available. All other results represent zero-shot performance. Init, FT, and PT denote the initialization model, fine-tuning, and pre-training, respectively. Performance gains in green and blue are compared to the base LLM and VDocRetirver†, respectively.

Generator	Retriever	Docs	ChartQA Single	ChartQA All	SlideVQA Single	SlideVQA All	InfoVQA Single	InfoVQA All	DUDE Single	DUDE All
Closed-book										
Phi3	–	–	20.0	20.0	20.3	20.3	34.9*	34.9*	23.1*	23.1*
Text-based RAG										
Phi3	Phi3	Text	28.0	28.0	28.6	28.0	40.5*	39.1*	40.1*	35.7*
Phi3	Gold	Text	36.6	36.6	27.8	27.8	45.6*	45.6*	55.9*	55.9*
VDocRAG (Ours)										
VDocGenerator	VDocRetriever	Image	**52.0**$_{+24.0}$	**48.0**$_{+20.0}$	**44.2**$_{+15.6}$	**42.0**$_{+14.0}$	**56.2***$_{+15.7}$	**49.2***$_{+10.1}$	**48.5***$_{+8.4}$	**44.0***$_{+8.3}$
VDocGenerator	Gold	Image	74.0	74.0	56.4	56.4	64.6*	64.6*	66.4*	66.4*

Table 4. DocumentVQA results. All models are fine-tuned on OpenDocVQA. The results marked with * denote performance on unseen test samples, and the other results represent zero-shot performance. The performance gain in green is compared to the text-based RAG that has the same base LLM. Gold knows the ground-truth documents. Models answer the question based on the top three retrieval results.

verify the effectiveness of encoding documents through images, we fine-tuned the LLM in VDocRetriever (**Phi3** [1]), using extracted text to represent documents. Additionally, we included a variant of VDocRetriever without pre-training (**VDocRetriever†**).

QA baselines. We compared VDocRAG against **closed-book** and **text-based RAG** models. These baselines used the same model initialization as VDocRAG but fine-tuned only the LLM (Phi3). The closed-book model received only the question as input, while the text-based RAG used the top three documents retrieved by the Phi3 retriever. Moreover, we assessed possible upper-bound performance by testing generation with ground-truth (Gold) documents.

Evaluation metrics. We evaluated retrieval performance using **nDCG@5**, a widely used metric in information retrieval [17, 25]. For the DocumentVQA task, we followed the evaluation protocol of each dataset, we used **ANLS** [4] for InfoVQA and DUDE, **Relaxed Accuracy** [41] for ChartQA, **F1** for SlideVQA as evaluation metrics.

5.2. Retrieval Results

Table 3 shows that VDocRetriever† achieved significantly higher retrieval performance than the text-based Phi3 retriever on all datasets under the same conditions. This indicates that our model can effectively encode documents in image format for retrieval tasks. Furthermore, VDocRetriever exhibits superior zero-shot generalization on unseen datasets, ChartQA and SlideVQA, outperforming both off-the-shelf text retrievers and the state-of-the-art visual document retrieval models. Notably, DSE was initialized with the same LVLM as ours and fine-tuned on 13.7 times more data. This highlights that our pre-training strategy and the OpenDocVQA dataset offer unique advantages, which are not adequately addressed by existing approaches.

5.3. Retrieval-Augmented Generation Results

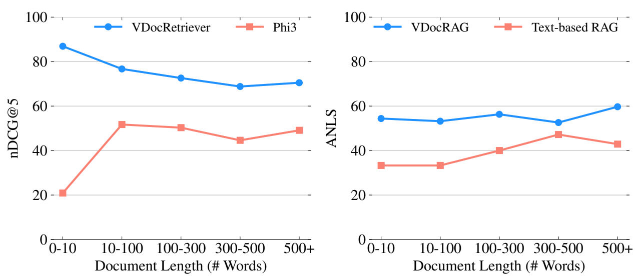
Table 4 shows that VDocRAG significantly outperformed both the closed-book LLM and the text-based RAG on

Model	SlideVQA	InfoVQA
VDocRetriever	**77.3**	**72.9**
w/o RCR	$75.9_{-1.4}$	$71.1_{-1.8}$
w/o RCG	$71.7_{-5.6}$	$68.8_{-4.1}$
w/o RCG & RCR	$71.0_{-6.3}$	$66.8_{-6.1}$
w/o LLM & Projector (↪ CLIP encoders)	$43.7_{-33.6}$	$37.9_{-35.0}$

Table 5. Ablation study of our pre-training tasks and model architecture in the retrieval task under the single-pool setting.

Model	Retrieval		QA	
	SlideVQA	InfoVQA	SlideVQA	InfoVQA
VDocRAG	**77.3**	**72.9**	**44.2**	**56.2**
w/o MHDocVQA	$75.0_{-2.3}$	$71.4_{-1.5}$	$43.4_{-0.8}$	$53.8_{-2.4}$
w/o except MHDocVQA	$68.8_{-8.5}$	$61.7_{-11.2}$	$41.1_{-3.1}$	$44.0_{-12.2}$

Table 6. Ablation study of our dataset in retrieval and QA tasks under the single-pool setting.



(a) Retrieval performance (b) QA performance

Figure 5. Performance under different document lengths on InfoVQA (single-pool setting).

the DocumentVQA task, even when all models were the same initialization. Additionally, when the retrieval results were fixed to ground-truth (Gold) documents, VDocRAG demonstrated superior performance to text-based RAG. This underscores the importance of visual cues in extracting answers from documents and suggests that VDocGenerator has a higher upper-bound performance. Both text-based RAG and VDocRAG exhibited substantial improvements when provided with ground-truth documents, highlighting potential areas for enhancing retrieval accuracy and improving the generator's robustness to retrieval noise.

5.4. Analysis

Can our pre-training tasks be beneficial? Table 5 shows that VDocRetriever outperformed the model without pre-training. Removing each pre-training task or both RCG and RCR tasks decreased performance, indicating that both tasks contribute complementarily. These validate that our pre-training effectively learns to compress image features while aligning them with textual contents in images.

Model	Retrieval		QA	
	OCR	Encoding	Generation	Total
Text-based RAG_{Phi3}	590.0	70.7	422.7	1083.4
VDocRAG	–	204.4	789.7	994.1

Table 7. Efficiency analysis on InfoVQA. The average time (ms) to encode a single document or generate a single answer is measured on a single A100 GPU.

Model	Retrieval		QA	
	SlideVQA	InfoVQA	SlideVQA	InfoVQA
Text-based $\text{RAG}_{\text{LLama3}}$	60.1	61.8	37.8	49.5
$\text{VDocRAG}_{\text{Idefics3}}$	**73.4**	**72.5**	**48.9**	**59.9**
w/o Pre-train	70.3	69.8	47.2	59.6

Table 8. Analysis with different LVLM (Idefics3) in retrieval and QA tasks under the single-pool setting.

Does LLM help understanding document images? Table 5 shows that retrieval performance dropped substantially when the LLM block was removed, leaving only the CLIP text/vision encoder, even with the same visual transformer backbone. This suggests that LLM can capture finer-grained visual details and enhance semantic understanding.

Does our dataset improve the performance? Table 6 shows that removing MHDocVQA caused a performance decrease, indicating that MHDocVQA requires distinct reasoning skills compared to other collected datasets in OpenDocVQA. Additionally, excluding all OpenDocVQA datasets except MHDocVQA led to a significant performance drop. This confirms that our collected datasets effectively supplement the missing capabilities of LVLM in document retrieval and understanding.

How well does VDocRAG perform under different document lengths? Figure 5 shows that VDocRAG consistently outperforms text-based RAG, indicating that VDocRAG can better understand documents through visual information. In general, we observed that the VDocRAG's relative performance over text-based RAG is larger for images with 0-10 words (+66.0 in retrieval, +21.1 in QA) than for those with 500+ words (+28.4 in retrieval, +16.7 in QA).

Is VDocRAG more efficient than text-based RAG? Table 7 shows that VDocRAG is more efficient than text-based RAG. Especially, VDocRAG requires 69% less inference time to retrieve documents than text-based RAG. Although VDocRetriever takes more time for document encoding and generation, it eliminates the time-consuming OCR processing necessary for text-based RAG.

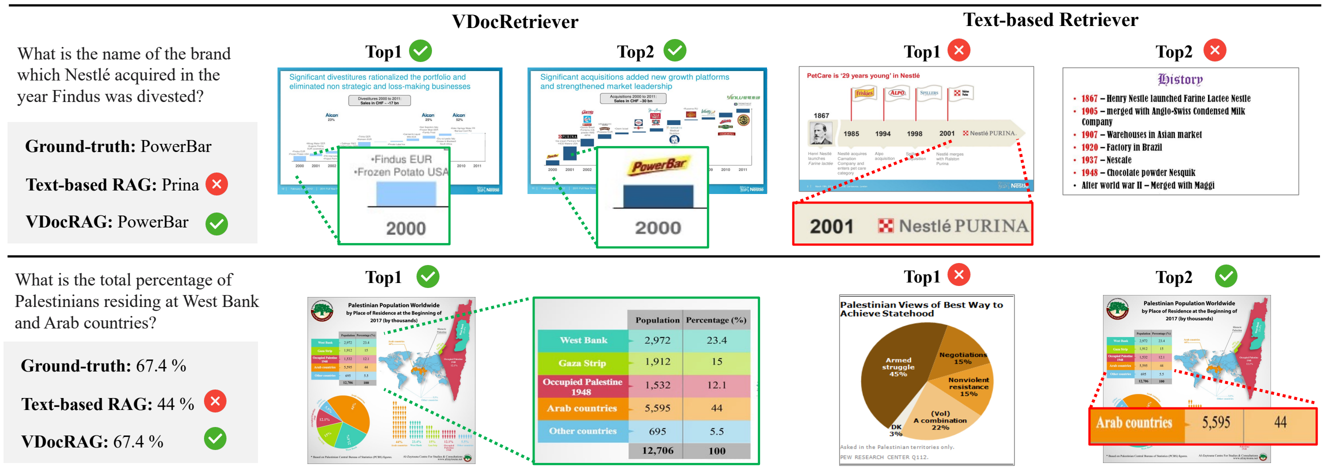
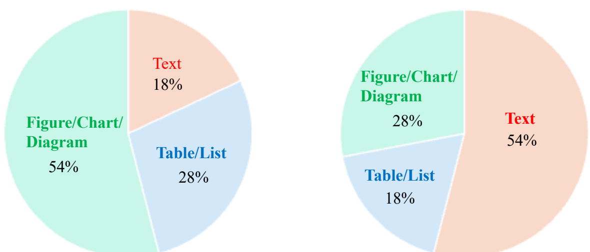


Figure 6. Qualitative results of VDocRAG compared to text-based RAG.



(a) VDocRAG answers correctly, but Text-based RAG answers incorrectly (b) VDocRAG answers incorrectly, but Text-based RAG answers correctly

Figure 7. Root causes of correct and incorrect predictions.

Can our method apply different LVLMs? To investigate the impact of different LVLMs on VDocRAG, we replaced Phi3V with Idefics3 [29], a state-of-the-art LVLM that uses Llama3-8B [16] as its backbone LLM. As observed in Table 8, the performance trend was consistent with that of Phi3V, highlighting the versatility and broad applicability of our method.

Qualitative results. Figure 6 illustrates the performance of our model through qualitative examples. In the top example, VDocRAG demonstrates strong performance on a question requiring multi-hop reasoning and graph understanding across multi-page slides. In the bottom example, VDocRAG also performs better on a question that requires parsing on the table with cells spanning multiple rows and columns. In contrast, text-based RAG depends solely on OCR text information, leading to a superficial understanding of the text and incorrect predictions.

Human evaluation. To better understand the prediction differences between VDocRAG and text-based RAG, we manually analyzed the generated outputs by identifying the root causes of 50 correct and 50 incorrect predictions, randomly sampled from test samples. Figure 7a shows that VDocRAG significantly enhances the understanding of visual data (e.g., charts). Conversely, Figure 7b reveals that VDocRAG encounters challenges with text-heavy documents (e.g., books), primarily due to the OCR capabilities. We observed that text-based RAG correctly answers questions when visual data includes long titles or subtitles, which have a high textual overlap with the question. These observations are in line with the results shown in Figure 5.

6. Conclusion

We introduced a new RAG framework, VDocRAG, which can directly understand various real-world documents. We enhanced VDocRAG with two key contributions: (1) pre-training tasks capable of learning image representation efficiently by leveraging the powerful capabilities of LVLMs, and (2) OpenDocVQA, the first unified open-domain DocumentVQA dataset that encompasses a wide range of visually-rich documents. Our holistic evaluations on four datasets show that VDocRAG significantly outperformed conventional text-based RAG, shedding light on the development of an effective RAG over real-world documents.

Limitations. While we focused on pre-training to align images and OCR data for document retrieval, leveraging caption data instead of OCR data offers the potential for retrieving images that do not contain text. Moreover, this study did not address reducing the computational cost of creating search indexes for extensive image collections. We plan to reduce the cost of VDocRAG with more efficient techniques. Lastly, joint training of QA and retrieval components simultaneously further optimizes their interactions.

References

[1] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv:2404.14219*, 2024. 2, 5, 6, 3

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv:2303.08774*, 2023. 1, 3

[3] Akari Asai, Sewon Min, Zexuan Zhong, and Danqi Chen. Retrieval-based language models and applications. In *ACL*, pages 41–46, 2023. 2

[4] Ali Furkan Biten, Rubèn Tito, Andrés Mafla, Lluís Gómez i Bigorda, Marçal Rusiñol, C. V. Jawahar, Ernest Valveny, and Dimosthenis Karatzas. Scene text visual question answering. In *ICCV*, pages 4290–4300, 2019. 6

[5] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. Improving language models by retrieving from trillions of tokens. In *ICML*, pages 2206–2240, 2022. 2

[6] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. `https://github.com/kakaobrain/coyo-dataset`, 2022. 3

[7] Jingwen Chen, Yingwei Pan, Yehao Li, Ting Yao, Hongyang Chao, and Tao Mei. Retrieval augmented convolutional encoder-decoder networks for video captioning. *TOMCCAP*, pages 1–24, 2023. 2

[8] Wenhu Chen, Hexiang Hu, Chitwan Saharia, and William W Cohen. Re-imagen: Retrieval-augmented text-to-image generator. *arXiv:2209.14491*, 2022. 2

[9] Jaemin Cho, Debanjan Mahata, Ozan Irsoy, Yujie He, and Mohit Bansal. M3DocRAG: Multi-modal retrieval is what you need for multi-page multi-document understanding. *arXiv:2411.04952*, 2024. 2

[10] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. *arXiv:2305.06500*, 2023. 2

[11] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with io-awareness. In *NeurIPS*, pages 16344–16359, 2022. 5

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186, 2019. 5, 6

[13] Kuicai Dong, Yujing Chang, Xin Deik Goh, Dexun Li, Ruiming Tang, and Yong Liu. MMDocIR: Benchmarking multi-modal retrieval for long documents. *arXiv:2501.08828*, 2025. 2

[14] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Songyang Zhang, Haodong Duan, Wenwei Zhang, Yining Li, et al. Internlm-xcomposer2-4khd: A pioneering large vision-language model handling resolutions from 336 pixels to 4k hd. *arXiv:2404.06512*, 2024. 4

[15] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. *arXiv:2401.08281*, 2024. 4

[16] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv:2407.21783*, 2024. 1, 8

[17] Manuel Faysse, Hugues Sibille, Tony Wu, Gautier Viaud, Céline Hudelot, and Pierre Colombo. ColPali: Efficient document retrieval with vision language models. *arXiv:2407.01449*, 2024. 1, 2, 3, 6

[18] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *ICML*, pages 3929–3938, 2020. 1

[19] Matthew Honnibal and Ines Montani. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017. 3

[20] Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Chen Li, Ji Zhang, Qin Jin, Fei Huang, et al. mplug-docowl 1.5: Unified structure learning for ocr-free document understanding. *arXiv:2403.12895*, 2024. 5

[21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *arXiv:2106.09685*, 2021. 5

[22] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *arXiv:2112.09118*, 2021. 5, 6, 3

[23] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. *arXiv:2310.06825*, 2023. 6

[24] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv:2401.04088*, 2024. 1, 3

[25] Ehsan Kamalloo, Nandan Thakur, Carlos Lassance, Xueguang Ma, Jheng-Hong Yang, and Jimmy Lin. Resources for brewing beir: Reproducible reference models and an official leaderboard, 2023. 6

[26] Yuma Koizumi, Yasunori Ohishi, Daisuke Niizumi, Daiki Takeuchi, and Masahiro Yasuda. Audio captioning using pre-trained large-scale language model guided by audio-based similar caption retrieval. *arXiv:2012.07331*, 2020. 2

[27] Sunjun Kweon, Yeonsu Kwon, Seonhee Cho, Yohan Jo, and Edward Choi. Open-WikiTable : Dataset for open domain question answering with complex reasoning over table. In *Findings of ACL*, pages 8285–8297, 2023. 3, 5, 1

[28] Jordy Landeghem, Rubén Tito, Łukasz Borchmann, Michał Pietruszka, Paweł Józiak, Rafał Powalski, Dawid Jurkiewicz, Mickaël Coustaty, Bertrand Ackaert, Ernest Valveny, et al. Document understanding dataset and evaluation (dude). In *ICCV*, pages 19528–19540, 2023. 2, 3, 5, 1

[29] Hugo Laurençon, Andrés Marafioti, Victor Sanh, and Léo Tronchon. Building and better understanding vision-language models: insights and future directions. *arXiv:2408.12637*, 2024. 2, 8

[30] Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nv-Embed: Improved techniques for training llms as generalist embedding models. *arXiv:2405.17428*, 2024. 5, 6, 3

[31] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *NIPS*, pages 9459–9474, 2020. 1

[32] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, pages 12888–12900, 2022. 2

[33] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, pages 19730–19742, 2023. 2

[34] Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. *arXiv:2308.03281*, 2023. 5, 6

[35] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv:2304.08485*, 2023. 2

[36] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv:1711.05101*, 2017. 5

[37] Xueguang Ma, Sheng-Chieh Lin, Minghan Li, Wenhu Chen, and Jimmy Lin. Unifying multimodal retrieval via document screenshot embedding. *arXiv:2406.11251*, 2024. 1, 2, 5, 6, 3

[38] Xueguang Ma, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, Wenhu Chen, and Jimmy Lin. VISA: Retrieval augmented generation with visual source attribution. *arXiv:2412.14457*, 2024. 2

[39] Seiji Maekawa, Hayate Iso, Sairam Gurajada, and Nikita Bhutani. Retrieval helps or hurts? a deeper dive into the efficacy of retrieval augmentation to language models. In *NAACL*, pages 5506–5521, 2024. 1, 2

[40] Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *ACL*, pages 9802–9822, 2023. 1, 2

[41] Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of ACL*, pages 2263–2279, 2022. 2, 3, 5, 6, 1

[42] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A dataset for vqa on document images. In *WACV*, pages 2200–2209, 2021. 1, 2, 5

[43] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and C.V. Jawahar. InfographicVQA. In *WACV*, pages 1697–1706, 2022. 1, 2, 3, 5

[44] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018. 4

[45] Md Rizwan Parvez, Wasi Uddin Ahmad, Saikat Chakraborty, Baishakhi Ray, and Kai-Wei Chang. Retrieval augmented code generation and summarization. *arXiv:2108.11601*, 2021. 2

[46] Le Qi, Shangwen Lv, Hongyu Li, Jing Liu, Yu Zhang, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ting Liu. DuReader$_{\text{vis}}$: A Chinese dataset for open-domain document visual question answering. In *Findings of ACL*, pages 1338–1351, 2022. 2, 3

[47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 5, 6

[48] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21(140):1–67, 2020. 2

[49] Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. In-context retrieval-augmented language models. *TACL*, pages 1316–1331, 2023. 2

[50] Rita Ramos, Desmond Elliott, and Bruno Martins. Retrieval-augmented image captioning. In *EACL*, pages 3666–3681, 2023. 2

[51] Rita Ramos, Bruno Martins, Desmond Elliott, and Yova Kementchedjhieva. Smallcap: lightweight image captioning prompted with retrieval augmentation. In *CVPR*, pages 2840–2849, 2023. 2

[52] Stephen Robertson, Hugo Zaragoza, et al. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389, 2009. 5, 6

[53] Junyoung Seo, Susung Hong, Wooseok Jang, Inès Hyeonsu Kim, Minseop Kwak, Doyup Lee, and Seungryong Kim. Retrieval-augmented score distillation for text-to-3d generation. *arXiv:2402.02972*, 2024. 2

[54] Ray Smith. An overview of the tesseract ocr engine. In *ICDAR*, pages 629–633, 2007. 5

[55] Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv:2210.09261*, 2022. 1

[56] Ryota Tanaka, Kyosuke Nishida, and Sen Yoshida. VisualMRC: Machine reading comprehension on document images. In *AAAI*, pages 13878–13888, 2021. 1, 2, 3, 5

[57] Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito. SlideVQA: A dataset for document visual question answering on multiple images. In *AAAI*, pages 13636–13645, 2023. 1, 2, 3, 5

[58] Ryota Tanaka, Taichi Iki, Kyosuke Nishida, Kuniko Saito, and Jun Suzuki. Instructdoc: A dataset for zero-shot generalization of visual document understanding with instructions. In *AAAI*, pages 19071–19079, 2024. 2

[59] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv:2212.03533*, 2022. 5, 6

[60] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models. In *ACL*, pages 11897–11916, 2024. 5, 6, 3

[61] Jilan Xu, Yifei Huang, Junlin Hou, Guo Chen, Yuejie Zhang, Rui Feng, and Weidi Xie. Retrieval-augmented egocentric video captioning. In *CVPR*, pages 13525–13536, 2024. 2

[62] Dongchao Yang, Songxiang Liu, Rongjie Huang, Chao Weng, and Helen Meng. Instructtts: Modelling expressive tts in discrete latent space with natural language style prompt. *TASLP*, pages 2913–2925, 2024. 2

[63] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, Qianyu Chen, Huarong Zhou, Zhensheng Zou, Haoye Zhang, Shengding Hu, Zhi Zheng, Jie Zhou, Jie Cai, Xu Han, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv:2408.01800*, 2024. 6

[64] Michihiro Yasunaga, Armen Aghajanyan, Weijia Shi, Rich James, Jure Leskovec, Percy Liang, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. Retrieval-augmented multimodal language modeling. In *ICML*, pages 39755–39769, 2023. 2

[65] Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, Qin Jin, Liang He, Xin Lin, and Fei Huang. UReader: Universal OCR-free visually-situated language understanding with multimodal large language model. In *EMNLP Findings*, pages 2841–2858, 2023. 4

[66] Shi Yu, Chaoyue Tang, Bokai Xu, Junbo Cui, Junhao Ran, Yukun Yan, Zhenghao Liu, Shuo Wang, Xu Han, Zhiyuan Liu, et al. VisRAG: Vision-based retrieval-augmented generation on multi-modality documents. *arXiv:2410.10594*, 2024. 2, 5, 6

[67] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, pages 11975–11986, 2023. 2

[68] Liang Zhang, Anwen Hu, Jing Zhang, Shuo Hu, and Qin Jin. MPMQA: multimodal question answering on product manuals. In *AAAI*, pages 13958–13966, 2023. 2, 3, 5, 1

[69] Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. Remodiffuse: Retrieval-augmented motion diffusion model. In *ICCV*, pages 364–373, 2023. 2

[70] Shuyan Zhou, Uri Alon, Frank F Xu, Zhiruo Wang, Zhengbao Jiang, and Graham Neubig. Docprompting: Generating code by retrieving the docs. *arXiv:2207.05987*, 2022. 2

[71] Fengbin Zhu, Wenqiang Lei, Fuli Feng, Chao Wang, Haozhou Zhang, and Tat-Seng Chua. Towards complex document understanding by discrete reasoning. In *ACMM*, pages 4857–4866, 2022. 2