SLIDEQA: A MULTIMODAL BENCHMARK FOR LECTURE SLIDE QUESTION ANSWERING

Nathan McNaughton, Nicholas Eliacin & Shanaya Malik
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
{nathanmc03, nicholas_eliacin, shanaya_malik}@berkeley.edu

ABSTRACT

Existing question-answering (QA) systems rely predominantly on text-only retrieval, even though lecture slides are a primary medium for university instruction. This is insufficient because slides integrate visuals such as diagrams, formulas, tables, charts, and spatial structure that text-based systems fail to capture, often leading to incomplete or incorrect answers. Moreover, there is no standardized benchmark for evaluating multimodal QA over lecture slides. To address this gap, we introduce SlideQA, a multimodal benchmark for lecture slide QA built from three public NLP courses at UC Berkeley, Stanford, and Johns Hopkins. The benchmark includes five categories of questions: (1) text-only reasoning, (2) image and diagram interpretation, including mathematical expressions, (3) table comprehension, (4) chart and graph analysis, and (5) layout-aware reasoning.

We construct SlideQA from publicly available lecture PDFs and curate 450 question-answer pairs (150 per course) spanning three institutions and diverse slide formats. We evaluate five systems: closed-book GPT-4o, a text-only BM25 baseline, a dense text retrieval baseline using sentence-transformer embeddings, a ColPali retrieval-augmented generation (RAG) pipeline, and an oracle VLM upper bound. ColPali RAG achieves an average LLM-Judge score of 3.62 (on a 1–5 scale), compared to 2.12 for dense text retrieval, 1.81 for BM25 text-only, and 4.57 for oracle VLM, confirming that multimodal retrieval substantially outperforms text-based approaches while narrowing the gap to oracle slide access. SlideQA provides a reproducible benchmark for multimodal reasoning over instructional materials and enables systematic evaluation of retrieval-augmented VLM agents in educational settings.

1 INTRODUCTION

Lecture slides are the primary medium for delivering instruction in university courses, yet the information in slides resists conventional text-based question answering. A lecture slide combines mathematical notation, annotated diagrams, data tables, and explanatory text within a deliberate spatial layout. Text extraction via OCR or PDF parsing loses this structure entirely, reducing a multimodal document to tokens. This is particularly problematic for technical courses such as NLP, where slides routinely contain architecture diagrams and equation derivations that cannot be solely represented as plain text. In spite of growing interest in document visual question answering (DocVQA), there are no existing benchmarks that specifically target lecture slide QA for university-level technical content. SlideVQA (Tanaka et al., 2023) focuses on general-domain business presentations, while ChartQA (Masry et al., 2022) and InfographicVQA (Mathew et al., 2022) target specific visual modalities in isolation—none capture the multimodal diversity of lecture slides.

We introduce SlideQA, a multimodal QA benchmark constructed from publicly available lecture slides of three NLP courses at UC Berkeley (CS 288), Stanford (CS 224N), and Johns Hopkins (CS 601.471). SlideQA defines five question categories: (1) text-only reasoning, (2) image/diagram interpretation, (3) table comprehension, (4) chart/graph analysis, and (5) layout-aware reasoning. We evaluate five systems: closed-book GPT-4o, a text-only BM25 baseline, a dense text retrieval baseline, a ColPali RAG pipeline, and an oracle VLM upper bound. ColPali RAG outperforms

text-only retrieval by 2.1× on Token F1 and achieves an average LLM-Judge score of 3.62 vs. 2.12 for dense text and 1.81 for BM25 text-only, confirming that the benchmark requires genuine visual understanding.

Our contributions are: (1) SlideQA, a benchmark of 450 curated questions (150 per course) across three institutions and five categories; (2) an automated pipeline for generating and curating slide-based QA pairs; (3) a ColPali retrieval-augmented generation pipeline that substantially outperforms text-only retrieval (LLM-Judge 3.62 vs. 1.81) and narrows the gap to oracle VLM access (4.57); (4) baseline evaluations demonstrating that dense semantic retrieval modestly outperforms sparse BM25 (LLM-Judge 2.12 vs. 1.81) while multimodal visual retrieval with ColPali provides a substantially larger gain (3.62), confirming that visual grounding—not retrieval paradigm—is the key bottleneck.

2 Related Work

Our work draws on three areas: (1) document and slide-specific QA benchmarks, (2) visual and structural reasoning, and (3) multimodal retrieval-augmented generation.

Document and Slide-Specific Benchmarks. SlideVQA (Tanaka et al., 2023) introduced approximately 52,000 slide images for multi-image reasoning but focuses on general-domain business presentations where visual elements tend to be decorative. The Lecture Presentations Multimodal Dataset (Lee et al., 2023) explored multimodal understanding in educational videos but did not isolate slide-level QA. SlideQA targets university-level NLP lecture slides, which contain mathematical formulas, algorithm diagrams, and dense tables where the visual layout carries semantic meaning that text extraction cannot recover.

Visual and Structural Reasoning. ChartQA (Masry et al., 2022) targets reasoning over data visualizations, while InfographicVQA (Mathew et al., 2022) tests comprehension of complex layouts combining text, icons, and graphics. These benchmarks address individual visual modalities in isolation, but lecture slides require reasoning across multiple modalities simultaneously. SlideQA addresses this through five question categories enabling fine-grained analysis of which capabilities are needed. Our per-category LLM-Judge scores for ColPali RAG range from 3.41 (image/diagram) to 3.95 (chart/graph), while text-only scores range from 1.22 to 2.47, confirming that different categories test genuinely different capabilities.

Multimodal Retrieval-Augmented Generation. Traditional RAG pipelines discard visual information by converting documents to text before retrieval. VisRAG (Yu et al., 2024) and ColPali (Faysse et al., 2024) bypass this bottleneck by embedding document pages directly as images. SlideQA adopts this paradigm: our four systems—closed-book, text-only BM25, ColPali RAG, and oracle VLM—span the full range from no retrieval to perfect slide access, enabling direct measurement of how much retrieval and visual grounding each contribute.

3 Baseline Setup

3.1 Implementation Details

Throughout this paper, we use *text-only* to refer specifically to the BM25 sparse retrieval baseline, and *text-based* as a general adjective for any approach that operates on extracted slide text rather than slide images (encompassing both BM25 and dense text retrieval).

We evaluate five systems using GPT-4o (via OpenRouter). The **closed-book baseline** provides GPT-4o with only the question and no slide context, measuring how much can be answered from parametric knowledge alone. The **text-only baseline (BM25 + LLM)** retrieves the top-3 slides using BM25 over extracted slide text (PyMuPDF with OCR fallback) and passes the concatenated text to GPT-4o with no visual input. The **dense text retrieval baseline** replaces BM25 sparse matching with cosine similarity over sentence embeddings (`all-MiniLM-L6-v2`) computed from the same extracted slide text, passing retrieved text to GPT-4o without visual input. The **ColPali RAG system** retrieves visually relevant slides using image-based vision-language embeddings and passes

the retrieved slide images to GPT-4o. The **oracle VLM baseline** provides GPT-4o with the gold evidence slide image directly, bypassing retrieval and serving as an upper bound given perfect slide selection.

3.2 Dataset Details

SlideQA is a synthetic dataset constructed from lecture slides of three publicly available NLP courses:

- **UC Berkeley CS 288**: 17 lectures, 1,301 slides. Modern PDF slides with diagrams, tables, charts, and math.
- **JHU CS 601.471**: 19 lectures, 1,264 slides. Older PPT slides converted to PDF, predominantly text-heavy.
- **Stanford CS 224N**: 19 lectures, 1,207 slides. Diagram- and formula-heavy slides.

For each course, we generated QA drafts by prompting GPT-4o with each slide image to produce question–answer pairs across the five categories. Raw drafts are curated through quality filters removing trivial questions, answers shorter than 2 characters or longer than 10 words, and vague or self-referential questions. We select 150 QA pairs per course via balanced sampling (minimum 5 per category), prioritizing hard/medium difficulty and multimodal categories, followed by manual review using a Streamlit annotation tool. The 450 curated pairs (150 per course) serve as the full evaluation benchmark.

Table 1: Dataset summary across three courses. Categories: TO = text_only, ID = image_diagram, TB = table, CG = chart_graph, LA = layout_aware.

Course	Lectures	Slides	Raw	Curated	TO	ID	TB	CG	LA
CS 288	17	1,301	1,300	150	15	60	28	21	26
CS 601	19	1,264	1,264	150	15	59	27	16	33
CS 224N	19	1,207	1,207	150	15	57	27	28	23
Total	55	3,772	3,771	450	45	176	82	65	82

The category distribution reflects the slide format differences across institutions. All three courses are balanced at 15 text_only questions each by design. CS 288 and CS 224N's diagram-heavy modern slides yield the most image_diagram questions (40% and 38% respectively), while CS 601's older PPT-style slides produce more layout_aware questions (22%). Section 6.4 shows representative examples from each category alongside the evidence slides.

3.3 Evaluation Metrics

We report Token F1 and LLM-Judge scores. Token F1 measures token-level overlap with the reference answer. However, Token F1 is overly strict for SlideQA because many correct answers differ lexically from the reference despite being semantically equivalent. We therefore use LLM-Judge as our primary metric: GPT-4o scores each predicted answer on a 1–5 correctness scale given the question and reference answer. Token F1 is reported for comparability, while LLM-Judge better captures semantic correctness under paraphrase and visually grounded descriptions. All systems are prompted to produce concise answers of at most 10 words.

4 ColPali RAG Pipeline

4.1 Motivation

The baselines described in Section 3 share a common limitation: they all operate on extracted slide text and discard visual content. Our proposed system replaces text-based retrieval with image-based retrieval using vision-language embeddings, preserving the visual and structural information that text extraction loses. In practice, the key challenge is retrieving relevant slides from a large

collection without text extraction; our goal is to bridge the gap between text-only retrieval and the oracle upper bound.

4.2 Pipeline Design

Our approach replaces text-based retrieval with image-based retrieval using vision-language embeddings. The pipeline consists of four stages:

1. **Slide Preprocessing**: Lecture PDFs are converted into individual slide images using a PDF-to-image pipeline.
2. **Embedding**: Each slide image is encoded into a dense vector representation using a vision-language model.
3. **Retrieval**: Given a question, we compute a query embedding and retrieve the top-k most relevant slides using cosine similarity.
4. **Answer Generation**: Retrieved slide images are passed to GPT-4o along with the question to generate an answer.

4.3 Implementation Details

We use top-$k = 3$ retrieval for all experiments. Slide embeddings are stored in a vector index and queried using cosine similarity. All answers are generated using GPT-4o with image input, with a system prompt requesting concise answers of at most 10 words.

5 Experiments

5.1 Setup

We evaluate five systems on the full 450-question SlideQA benchmark: **Closed-book**, **Text-only (BM25 + LLM)**, **Dense Text RAG** (sentence-transformer retrieval), **ColPali RAG** (ours), and **Oracle VLM**. All systems use GPT-4o (`gpt-4o-2024-11-20`) via OpenRouter as the answer generator with temperature $= 0$ for deterministic outputs and a concise answer format of at most 10 words requested via the system prompt. For retrieval-based systems, we retrieve the top k=3 slides per query. BM25 retrieval uses the rank-bm25 library over PyMuPDF-extracted slide text (with Tesseract OCR fallback for image-only slides). Dense text retrieval uses `all-MiniLM-L6-v2` sentence embeddings with cosine similarity over the same extracted text. ColPali embeddings are computed using `vidore/colpali-v1.2` and stored in a flat cosine-similarity index. LLM-Judge scoring uses a separate GPT-4o call with temperature $= 0$ and a 1–5 integer scoring rubric.

5.2 Results

Table 2 reports Token F1 and LLM-Judge scores (1–5 scale) for each course across the full 450-question benchmark.

ColPali RAG substantially outperforms all text-based retrieval systems and closed-book generation on all three courses. The average LLM-Judge score of 3.62 for ColPali RAG represents a $1.7\times$ improvement over dense text retrieval (2.12), a $2.0\times$ improvement over BM25 text-only (1.81), and a $1.8\times$ improvement over closed-book (1.98). Dense text retrieval (2.12) modestly outperforms BM25 (1.81), indicating that dense semantic matching provides a small but consistent benefit over keyword matching. The Oracle VLM ceiling of 4.57 indicates the headroom remaining from retrieval imperfection.

5.3 Discussion

The consistent advantage of ColPali RAG over text-only retrieval across all courses confirms that slide images encode information that extracted text cannot recover. The gap between ColPali RAG and Oracle VLM (3.62 vs. 4.57) reflects retrieval imperfection: when the gold evidence slide is absent from the top-k results, the generator is conditioned on irrelevant visual context. CS 601 shows the lowest RAG performance (3.39), consistent with its text-heavy PPT-style slides producing less distinctive visual embeddings than the diagram-rich slides of CS 288 and CS 224N. The similar

Table 2: Main results on SlideQA (150 questions per course, 450 total). LLM-Judge is on a 1–5 scale. Oracle VLM is given the gold evidence slide directly and serves as an upper bound. †Our method.

	Closed-Book	Text-Only	Dense Text	ColPali RAG†	Oracle VLM
Token F1					
CS 288	0.088	0.143	0.157	0.292	**0.377**
CS 601	0.090	0.186	0.209	0.296	**0.373**
CS 224N	0.077	0.100	0.108	0.316	**0.394**
Average	0.085	0.143	0.158	0.301	**0.381**
LLM-Judge (1–5)					
CS 288	2.09	1.79	2.11	3.73	**4.70**
CS 601	2.03	2.11	2.57	3.39	**4.42**
CS 224N	1.81	1.53	1.68	3.75	**4.60**
Average	1.98	1.81	2.12	3.62	**4.57**

performance of closed-book (1.98), BM25 text-only (1.81), and dense text retrieval (2.12) suggests that text-based retrieval adds minimal grounding over parametric knowledge alone regardless of the retrieval paradigm, reinforcing the necessity of visual retrieval for this task.

6 Analysis

6.1 Ablations

Retrieval vs Closed-Book Comparing ColPali RAG (average LLM-Judge: 3.62) to closed-book generation (1.98) isolates the contribution of retrieval. The large gap demonstrates that retrieved slide images provide essential grounding that parametric knowledge cannot substitute.

Text vs Multimodal Retrieval Comparing ColPali RAG (average Token F1: 0.301) to dense text retrieval (0.158) and BM25 (0.143) isolates the benefit of visual over textual retrieval. The $1.9\times$ Token F1 improvement over dense text retrieval confirms that ColPali embeddings recover information that even dense semantic search over extracted text cannot, with the bottleneck being the loss of visual information during text extraction rather than the retrieval paradigm. Table 3 shows per-course retrieval recall; averaged across courses, the pipeline surfaces at least one relevant slide in the top five for 78% of questions (R@5=0.776). The remaining 22% where retrieval fails drives the residual gap between ColPali RAG and Oracle VLM.

Table 3: ColPali retrieval recall at k. R@k is the fraction of questions for which the gold evidence slide appears in the top k retrieved results.

Course	R@1	R@3	R@5
CS 288	0.500	0.747	0.793
CS 601	0.373	0.600	0.707
CS 224N	0.480	0.807	0.827
Average	0.451	0.718	0.776

6.2 Category Breakdown

Table 4 reports LLM-Judge scores per question category, averaged across all three courses.

ColPali RAG shows the largest improvement over text-only systems on chart/graph questions (3.95 vs. 1.22 BM25, 1.19 dense text), where neither keyword matching nor dense semantic search can recover visual data encoded in the chart—confirming that visual information loss during extraction

Table 4: LLM-Judge scores (1–5) by question category, averaged across all three courses. †Our method.

Category	Closed-Book	Text-Only	Dense Text	ColPali RAG†	Oracle VLM
Text-only	2.29	2.47	3.09	3.84	**4.67**
Image/Diagram	2.06	1.81	2.15	3.41	**4.60**
Table	1.62	1.85	2.06	3.74	**4.55**
Chart/Graph	1.84	1.22	1.19	3.95	**4.59**
Layout-Aware	2.06	1.85	2.25	3.58	**4.46**

is the bottleneck, not the retrieval paradigm. Table questions benefit similarly (3.74 vs. 1.85 BM25, 2.06 dense text), as slide tables are often not fully preserved by PDF text extraction. Text-only questions show the smallest gap (3.84 vs. 2.47 BM25, 3.09 dense text), with dense text retrieval coming closest to ColPali since the answers are primarily text-grounded. Layout-aware questions (3.58 RAG vs. 1.85 BM25, 2.25 dense text) benefit substantially from visual retrieval, though they remain harder than structured visual categories as they require holistic spatial reasoning over the full slide.

6.3 Error Analysis

Three primary failure modes emerge from analysis of incorrect ColPali RAG responses. First, *retrieval failure*: when the gold evidence slide is absent from the top-k results ($\sim$22% of questions at k=5), the model is conditioned on irrelevant visual context and generates confident but factually wrong answers. Second, *VLM hallucination*: even with the correct slide retrieved, GPT-4o occasionally ignores specific visual elements in favor of parametric knowledge, most commonly on layout-aware questions that require precise spatial reading of the slide. Third, *metric misalignment*: Token F1 penalizes correct paraphrased answers, creating a large divergence from LLM-Judge scores (ColPali RAG: Token F1 0.301 vs. LLM-Judge 3.62/5.0). This motivates our use of LLM-Judge as the primary evaluation metric.

6.4 Qualitative Examples

Text-only failure, multimodal success (CS 288, image/diagram). **Question:** What mathematical expression is used for length normalization?
Gold: $p(\bar{x}) \propto \frac{1}{|\bar{x}|^{\alpha}} \sum_{i=1}^{|\bar{x}|} \log p(x_i|x_1, \ldots, x_{i-1})$
Text-only: "The provided context does not include information about a mathematical expression used for length normalization."
ColPali RAG: $p(\bar{x}) \propto \frac{1}{|\bar{x}|^{\alpha}} \sum_{i=1}^{|\bar{x}|} \log p(x_i \mid x_1, \ldots, x_{i-1})$
Oracle VLM: $p(\bar{x}) \propto \frac{1}{|\bar{x}|^{\alpha}} \sum_{i=1}^{|\bar{x}|} \log p(x_i \mid x_1, \ldots, x_{i-1})$

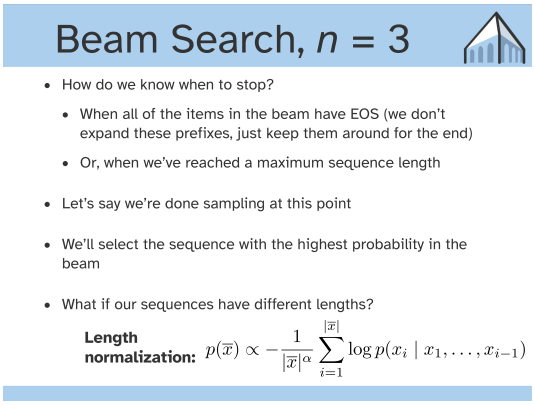

Figure 1: Evidence slide for the length normalization example. The formula appears only as a rendered image; OCR-extracted text does not capture it.

The formula is visually encoded on the slide and absent from extracted text. ColPali RAG retrieves the relevant image and matches oracle VLM.

Chart/graph failure for text-only (CS 224N, chart/graph). Question: How are the words 'man', 'woman', and 'king' represented in the graph?
Gold: They are represented as points with vectors indicating relationships.
Text-only: "The provided context does not include specific information about how the words 'man', 'woman', and 'king' are represented in the graph."
ColPali RAG: "In the graph, the words 'man', 'woman', and 'king' are represented as points in a 2D space, with arrows indicating the direction and relationship between them."
Oracle VLM: "In the graph, the words 'man', 'woman', and 'king' are represented as points in a vector space. Arrows indicate the direction and relationship between these points, illustrating the analogy 'man:woman :: king:?'."

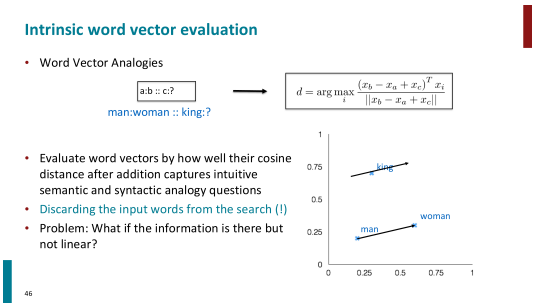

Figure 2: Evidence slide for the word vector analogy example. The relationship between man/woman/king is expressed entirely through the scatter plot; no equivalent textual description exists in the extracted slide text.

The answer requires reading a vector-space diagram. BM25 finds no relevant text; ColPali retrieves the slide image and correctly describes the representation.

Retrieval failure (CS 288, text-only category). Question: How is the loss function $L(\theta)$ defined in the training process?
Gold: $L(\theta) = -\sum_{t=1}^{T} \log(\hat{y}_t(y_t))$
Text-only: "The loss function $L(\theta)$ is defined using cross entropy loss for logistic regression [...]"
ColPali RAG: "The loss function $L(\theta)$ is used for learning, and an example given is cross-entropy."
Oracle VLM: $\mathcal{L}(\theta) = -\sum_{t=1}^{T} \log(\hat{y}_t(y_t))$

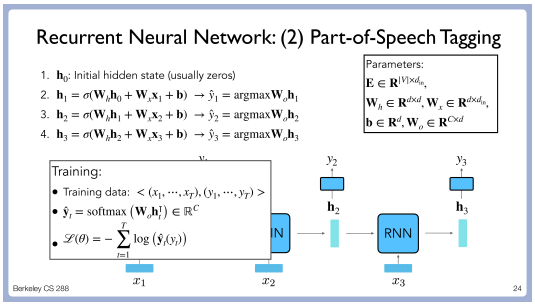

Figure 3: Evidence slide for the retrieval failure example. The gold formula $\mathcal{L}(\theta) = -\sum_{t=1}^{T} \log(\hat{y}_t(y_t))$ appears in the lower-left training box. ColPali RAG retrieved a different slide and produced an incorrect vague answer.

ColPali RAG did not retrieve the gold evidence slide and was conditioned on irrelevant context, producing a vague answer. Oracle VLM recovers the exact formula, confirming retrieval failure as the bottleneck.

Oracle VLM ceiling (CS 224N, table). Question: Name two example tasks or benchmarks for multiple-choice QA.
Gold: GLUE and MMLU.
Text-only: "GLUE and MMLU."
ColPali RAG: "The slide does not provide specific example tasks or benchmarks for multiple-choice QA."
Oracle VLM: "GLUE and MMLU."

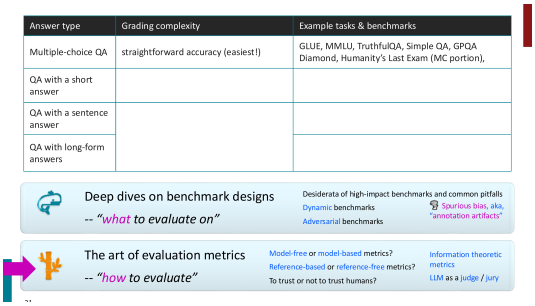

Figure 4: Evidence slide for the oracle VLM ceiling example. GLUE and MMLU appear in the top-right cell of the table. Text-only retrieval happened to surface this text; ColPali RAG retrieved the wrong slide and declined to answer.

Text-only happens to recover the correct answer from extracted text, but retrieval failure prevents ColPali RAG from doing so. Oracle VLM, given the gold slide directly, answers correctly.

7 Conclusion

We introduced SlideQA, a multimodal benchmark for lecture slide question answering across three public NLP courses, targeting five reasoning categories: text-only, image/diagram, table, chart/graph, and layout. Our experiments show that text-only retrieval is insufficient and multimodal retrieval with ColPali substantially improves answer quality (LLM-Judge 3.62 vs. 2.12 for dense text, 1.81 for BM25 text-only; oracle ceiling 4.57). The remaining gap to the oracle baseline is driven primarily by retrieval failures—when the gold evidence slide is missing from the top-k context ($\sim$22% of questions), the generator is conditioned on irrelevant slides. Future work should improve slide retrieval precision, scale SlideQA to more courses and domains, and explore stronger multimodal models for visually grounded educational QA.

DEMO LINK

https://cs288-shanaya.streamlit.app/

CONTRIBUTION STATEMENT

All three authors contributed equally to this project, and the work was divided as follows:

Nathan McNaughton led the ColPali RAG pipeline implementation, including slide image preprocessing, vision-language embedding computation, vector index construction, retrieval-augmented generation integration, and the paper write-up (`build_index.py`, `baselines/colpali_rag.py`, `report.tex`).

Nicholas Eliacin led benchmark construction, including PDF processing, automated QA generation with GPT-4o, curation filters, and balanced sampling across the five question categories, and the paper write up (`process_pdfs.py`, `generate_qa.py`, `curate_qa.py`, `slideqa_dataset.py`, `report.tex`).

Shanaya Malik led evaluation infrastructure and result analysis, including the LLM-Judge pipeline, Token-F1 scoring, per-category and per-course breakdowns, the Streamlit annotation and demo app, and the paper write-up (`evaluate.py`, `run_judge.py`, `recall_at_k.py`, `category_breakdown.py`, `app.py`, `report.tex`).

All authors participated in dataset design, experiment planning, presentation preparation, and report writing.

GENAI ACKNOWLEDGEMENT

We acknowledge the use of GitHub Copilot (`https://github.com/features/copilot`) for tab-based code completion in some of the pipeline scripts, and for conceptually explaining the ColPali model. We have the ability to explain and independently replicate all code in this project if asked by an instructor.

REFERENCES

Manuel Faysse, Céline Hudelot, et al. ColPali: Efficient document retrieval with vision language models. *arXiv preprint arXiv:2407.01449*, 2024.

Dong Won Lee, Chaitali Mondal, Shaoyen Wang, et al. Lecture presentations multimodal dataset: Towards understanding multimodality in educational videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL*, 2022.

Minesh Mathew, Viraj Bagal, Rubèn Tito, et al. InfographicVQA. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022.

Ryota Tanaka, Kyosuke Nishida, Kosuke Yoshida, Takuya Hasegawa, and Itsumi Saito. SlideVQA: A dataset for document visual question answering on multiple images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

Shi Yu et al. VisRAG: Vision-based retrieval-augmented generation on multi-modality documents. *OpenReview*, 2024.