

Peer Review

Review of: "Document Haystacks: Vision-Language Reasoning Over Piles of 1000+ Documents"

Evgenii Arsentev¹

1. Independent researcher

1. What the work claims

The paper makes two contributions. The first is a pair of retrieval-plus-QA benchmarks, DocHaystack and InfoHaystack, built by filtering DocVQA and InfographicVQA down to questions whose answer should exist in exactly one document. The filtering has three stages (§3, Figure 2): GPT-4o labels each question as "general" or "specific", human reviewers check the "specific" ones for unique identifiers and search the Tesseract OCR text of other documents for collisions, and a final pass removes questions that GPT-4o can answer with no image at all. What survives is 109 DocVQA questions over 59 evidence documents and 155 InfographicVQA questions over 66 evidence documents (§3, "Final dataset profile"). Each question is then posed against a pool of 100, 200, or 1,000 documents. A training set of 2,835 questions (899 DocVQA, 1,936 InfographicVQA) is built "similarly".

The second contribution is V-RAG (§4, Figure 3). Three text-to-image encoders (CLIP ViT-L/14@336, SigLIP ViT-SO400M/14@384, OpenCLIP ConvNeXt-XXL@1024) each score the question against every page; the cosine similarities are averaged, the top 60 pages go to LLaVA-OneVision-7B with a yes/no relevance prompt, and the top 5 survivors go to an answering LMM. The headline claims are a 9-point and 11-point gain in Recall@1 over the best baseline on DocHaystack-1000 and InfoHaystack-1000 (Table 2), large VQA gains when any of four LMMs is wrapped in V-RAG (Table 3), and a further gain from LoRA fine-tuning Qwen2-VL-7B with distractor images (Table 3, row Qwen2-VL-f.t.+V-RAG).

2. Whether the conclusion holds on the data

The benchmark idea is sound, and I think it is the more lasting part of the paper. Searching OCR output for the retained identifiers across other documents is exactly the check I would have asked for; many

"needle" datasets skip it. The generic-knowledge filter is also sensible, and Table 1 motivates it well: GPT-4o answers 26.4% of DocVQA and 54.9% of InfographicVQA questions without seeing the image (Table 1, row DocVQA and row InfographicVQA, column GPT-4o). That second figure alone is a useful warning to the field.

The retrieval result is real in direction. It is smaller and less clearly attributed than the text says.

Where the headline gain comes from. On DocHaystack-1000, the strongest baseline is not a vision encoder but BM25 over OCR text: 56.88 (Table 2, row BM25 (OCR), R@1, column DocHaystack-1000) against V-RAG 66.06, a difference of $66.06 - 56.88 = 9.18$ points. The best text-to-image baseline in that column is OpenCLIP at 34.86. The Introduction, however, says the 9% gain is "compared to previous best text-to-image retrieval methods". Against text-to-image methods, the gap is $66.06 - 34.86 = 31.20$ points; against the best method overall, it is 9.18. Both are fine numbers, but the sentence mixes them. On InfoHaystack-1000, the best baseline is OpenCLIP, 53.55, and $64.52 - 53.55 = 10.97$, which does match the "11%". In both cases, these are percentage points, not relative percent, and the abstract should say so.

What the ablation actually shows. Table 4 is the most informative table in the paper and, read carefully, it points somewhere other than the prose. The three-encoder ensemble without the LMM filter reaches R@1 42.20 on DocHaystack-1000 (Table 4, row CLIP+SigLIP+OpenCLIP without VLM-filter). That is below BM25's 56.88 in Table 2. Adding the filter lifts it to 66.06, so the filter contributes $66.06 - 42.20 = 23.86$ points and the ensemble over the best single encoder contributes $42.20 - 34.86 = 7.34$. On InfoHaystack-1000, the ensemble helps even less: 56.13 for all three encoders against 53.55 for OpenCLIP alone, and adding OpenCLIP to CLIP+SigLIP lowers R@1 from 59.35 to 56.13 (Table 4, rows CLIP+SigLIP and CLIP+SigLIP+OpenCLIP, R@1, column InfoHaystack-1000). The text says that "adding SigLIP and OpenCLIP incrementally improves results" and that the advantage comes from "V-RAG's ensemble of multiple vision encoders". The first statement is contradicted by that one row. The second is not what the table isolates: most of the gain is the 7B relevance filter. I would like to see the filter applied on top of BM25 and on top of OpenCLIP alone. If BM25 + filter matches V-RAG, the practical recommendation changes a lot, because BM25 is far cheaper than running a ConvNeXt-XXL at 1024 px over the whole pool.

Mis-stated cells. In §5.2, the text says that on "InfoHaystack-100 and InfoHaystack-200" V-RAG reaches R@1 of 74.84% and 64.52%, higher than the previous best by 8% and 11%. In Table 2, those two values are InfoHaystack-200 (74.84) and InfoHaystack-1000 (64.52). InfoHaystack-100 is 79.35. The gaps check out once relabelled: $74.84 - 66.45 = 8.39$ and $64.52 - 53.55 = 10.97$. The Table 2 caption says V-RAG "consistently outperforms baseline models" on R@1, R@3, and R@5, but on InfoHaystack-200 R@5,

OpenCLIP has 89.03 against V-RAG's 88.39, and on InfoHaystack-100 R@5, they tie at 92.90. The body text's "on most categories" is the accurate version.

The VQA gains quoted in the Introduction. The Introduction states that V-RAG brings GPT-4o "over a 55% acc improvement on DocHaystack-200 and a 34% acc improvement on InfoHaystack-200". I could not reproduce either figure from Table 3. For DocHaystack-200, GPT-4o+V-RAG minus GPT-4o is $72.48 - 23.85 = 48.63$ points; as a relative change, it is $72.48 / 23.85 \approx 3.04$, i.e., about +204%. For InfoHaystack-200, the difference is $63.23 - 20.00 = 43.23$ points. The nearest value to 55 is DocHaystack-100: $81.65 - 27.52 = 54.13$, which is still not "over 55". Perhaps the numbers come from an earlier table version. They need to be reconciled.

Fairness of the no-RAG baselines. The Table 3 caption says GPT-4o and Qwen2-VL were run in "low-resolution mode" with adjusted image size so that 100 or 200 pages fit in context. Documents with small print become unreadable at low resolution, so the no-RAG rows measure, in part, the resolution cut rather than long-context reasoning. The comparison is still meaningful as "what you can do today without retrieval", but the text reads it as evidence about LMM reasoning over many images. The resolution actually used (pixels per page, or the API detail setting) should be stated, and a sentence should say plainly that the baseline and the V-RAG rows do not see images at the same resolution.

Fine-tuning result. The text calls Qwen2-VL-f.t.+V-RAG the top performer "especially for the larger 1000-document tasks". On InfoHaystack-1000, it scores 60.00, identical to untuned Qwen2-VL+V-RAG at 60.00 (Table 3, column InfoHaystack 1000). On InfoHaystack-100, the gain is $67.10 - 65.81 = 1.29$ points, which with 155 questions is two questions ($1/155 \approx 0.645$ points per question). The DocHaystack gains are larger, e.g., $73.39 - 66.06 = 7.33$ points at 1000, which is 8 of 109 questions. I would soften the claim for InfoHaystack.

Sample size and uncertainty. Every percentage in Tables 2 to 4 is a count out of 109 or 155. For example, $66.06 = 72/109$ and $56.88 = 62/109$, so the headline DocHaystack-1000 gain over BM25 is ten questions. A rough normal-approximation 95% interval for a proportion near 0.66 with $n = 109$ is $\pm 1.96 \times \sqrt{(0.66 \times 0.34 / 109)} \approx \pm 8.9$ points. Paired tests are tighter than that, but I cannot compute one because the paper gives no per-question outcomes. Questions are also clustered: 109 questions sit on 59 documents, so several questions share a gold page, and a single retrieval failure can knock out several questions at once. No confidence intervals, no paired test (McNemar or bootstrap over questions), and no seeds or repeated runs appear anywhere. With test sets this small, I think that is the main statistical gap.

Benchmark construction details that affect every number. The paper does not say how the 99, 199, or 999 distractor pages are chosen, whether the pool is the same for all questions, or whether the 100 and 200 pools are subsets of the 1,000 pool. Distractors drawn from the same DocVQA collection (same industry archives, same layouts) make a very different task from random pages. The uniqueness check searched "across other documents" in the OCR of "images in the dataset": it is unclear whether that means the haystack pool or the whole source corpus. If distractors are later drawn from outside the searched set, uniqueness is no longer guaranteed. The stage-by-stage funnel is also missing: how many questions entered each filter and how many left. With a final set of 264 questions, readers need to know whether it came from thousands or from a few hundred. Nothing is said about the number of annotators, their agreement, or the time spent.

Selection bias from the GPT-4o filter. Questions that GPT-4o could answer blind were removed, and then GPT-4o is one of the evaluated models. This filter specifically removes the items on which GPT-4o's parametric knowledge would help, which disadvantages GPT-4o relative to Gemini and Qwen2-VL in Table 3. It is a reasonable design choice for a vision benchmark. It should be disclosed next to Table 3, and ideally, the filter would be a union over several models (Table 1 already has LLaVA-OneVision and Qwen2-VL results). Table 1's caption calls them "LMMs" while the text calls the step an "LLM" filter, and the paper does not say which split Table 1 was computed on.

Judge metric. Accuracy is decided by GPT-4o-mini. The claim that this "achieves higher consistency and alignment with human judgment" carries no number: no agreement rate, no comparison to ANLS (the standard DocVQA and InfographicVQA metric), no sample of disagreements. The prompt is said to be in the Appendix. As rendered on Qeios, no appendix is present, so I could not inspect it.

Efficiency. The abstract says V-RAG lets LMMs "efficiently operate across thousands of images." No latency, throughput, GPU memory, or cost is reported. Per query, the pipeline runs three encoders over the pool (cacheable) and then 60 separate 7B VLM calls for the filter, plus the answer call. That can well be acceptable, but "efficiently" needs a number. The title says "1000+ documents"; the largest pool evaluated is exactly 1,000.

Baseline coverage. MIRAGE scores 3.67 on DocHaystack-100 (Table 3, row MIRAGE), below every other system by a wide margin. That is plausible for a model trained on natural images, but with no configuration details, a reader cannot tell a real result from a setup error. I also missed document-specific visual retrievers, the late-interaction page retrievers released in 2024 being the obvious family, and a

simple hybrid of BM25 and one vision encoder. Given that BM25 is the strongest DocHaystack baseline, a hybrid is the baseline V-RAG most needs to beat.

Overall, the conclusions that the benchmarks are hard and that retrieval plus an LMM relevance filter helps a great deal are supported. The specific attribution to the encoder ensemble, several quoted gains, and the "consistently" and "efficiently" claims go beyond what Tables 2 to 4 show.

3. Reproducibility

The authors did better here than many LMM papers, and I want to credit that. They give exact snapshots or checkpoints for GPT-4o (gpt-4o-2024-08-06), LLaVA-OneVision (LLaVA-OneVision-Qwen2-7b-0V-HF), and Qwen2-VL (Qwen2-VL-7B-Instruct), and exact encoder variants for all five vision retrievers (§5.1, "Baselines"). The filter prompt is quoted verbatim, as is the question-classification prompt in §3. The fine-tuning recipe gives batch size 32, peak learning rate $1e-4$, one epoch, LoRA rank 8, and 1–10 distractors per question. Code and data are linked to a GitHub repository; I did not audit its contents and cannot say whether the pools and splits are there.

What is missing:

- Gemini has no version at all. The cited source for Gemini (reference [2]) is a third-party blog URL, not a Google model card, and does not identify a model either.
- The GPT-4o-mini judge snapshot is not given, and its prompt sits in an appendix that is not on the Qeios page.
- No decoding parameters for any generator or judge: temperature, max tokens, top-p. For the yes/no filter, it matters whether a greedy token or a probability threshold is used.
- No seeds for the distractor sampling during fine-tuning or anywhere else.
- No compute: GPUs, training time for the LoRA run, inference time per query.
- LoRA target modules, alpha, dropout, warm-up, and schedule shape are not stated.
- The prompt used for the generic-knowledge step ("{question}, answer briefly.") is given, but the LLM that compares the blind answer with the ground truth is not named, nor is its prompt.
- How the pool of 1,000 pages is sampled, and the Tesseract version and settings used for the uniqueness search.
- Whether the 2,835 training questions come from documents disjoint from the 125 test evidence documents. Without that, the fine-tuned row cannot be read as generalization.

One rendering note: Figures 4 and 5 carry no numbers in the rendered text, so I could not check the top-k ablation discussed in §5.3.

4. *What specifically to fix*

1. Abstract and §1: state the 9 and 11 as percentage points, and say that on DocHaystack-1000 the reference is BM25 over OCR (56.88, Table 2), not a text-to-image method.
2. §1, second-to-last paragraph: the "over 55%" and "34%" GPT-4o gains do not match Table 3 (48.63 and 43.23 points on the -200 columns). Correct them or explain how they were computed.
3. §5.2, "Retrieving results": relabel 74.84 and 64.52 as InfoHaystack-200 and InfoHaystack-1000. Change "consistently" in the Table 2 caption, since OpenCLIP wins InfoHaystack-200 R@5 (89.03 vs 88.39).
4. §5.3 and Table 4. Rewrite the reading of this table: the filter adds 23.86 points on DocHaystack-1000 and the ensemble adds 7.34; adding OpenCLIP lowers InfoHaystack-1000 R@1 from 59.35 to 56.13. Then add the missing rows: BM25 + filter, OpenCLIP + filter, SigLIP+OpenCLIP, CLIP+OpenCLIP.
5. Equation 1 and the averaging step. Cosine similarities from CLIP, SigLIP, and OpenCLIP live on different scales (SigLIP is trained with a sigmoid loss and learned bias), so a plain mean lets one model dominate. Report per-encoder score statistics, or compare against rank fusion or z-scored averaging.
6. LMM-filter paragraph, §4: say what happens when fewer than k pages, or zero pages, get a "yes". Table 2 shows $R@3 = R@5$ for V-RAG on DocHaystack-100 (88.99), DocHaystack-200 (84.40), and InfoHaystack-200 (88.39), which suggests the filtered list is often shorter than five; report the average number of survivors and how often the gold page is filtered out.
7. Give 95% confidence intervals for every cell in Tables 2–4, and a paired test for V-RAG vs BM25 and V-RAG vs OpenCLIP on the -1000 columns. Per-question outputs in the repository would let readers do this themselves.
8. §3, "Final dataset profile": add the filtering funnel (counts in and out at each of the three steps), the number of annotators, and any agreement measure.
9. Describe how haystack pools are built: source of distractors, whether pools are fixed or per-question, whether $100 < 200 < 1000$, and whether the OCR uniqueness search covered every page that can appear in a pool.

10. State whether training documents overlap with the 59 + 66 test evidence documents. If they overlap, the fine-tuned row in Table 3 needs a separate caveat.
11. Disclose, next to Table 3, that the generic-knowledge filter used GPT-4o, and consider repeating the filter with the union of the three models in Table 1.
12. Metric, §5.1: report agreement between GPT-4o-mini and humans on a sample (with the sample size), give the judge snapshot, and add ANLS as a secondary metric so results can be compared with the DocVQA literature. Put the judge prompt in the Qeios version; right now it is absent.
13. Table 3 caption: give the exact resolution used for the no-RAG GPT-4o and Qwen2-VL runs, and the Gemini model name and version.
14. Explain the MIRAGE configuration behind its 3.67 on DocHaystack-100.
15. Add one timing table (encoding the 1,000-page pool, filter calls per query, total wall-clock per question) to support "efficiently".
16. Fix the reference list. Reference [2] points to a blog, not to Gemini documentation. The Qwen-VL entry repeats "Zhang Y" seven times in the author list. The Mini-Gemini entry cites arXiv:2310.12345, which looks like a placeholder; please check the authors and identifier against the actual paper.
17. Provide Figure 5's values as a table so the top-k ablation can be checked.
18. Title: "1000+" should become "1,000" unless a larger pool is added.

5. Recommendation

Major revisions. The benchmark is a useful, carefully motivated resource, and the uniqueness check with OCR search is a real contribution. The retrieval pipeline clearly helps. But several numbers in the text do not match the tables, the ablation does not support attributing the gain to the encoder ensemble, the haystack construction is under-specified, and no uncertainty is reported on test sets of 109 and 155 questions. None of this needs new ideas. It needs a filter-on-BM25 ablation, confidence intervals, a construction section that someone else could follow, and a corrected reading of Tables 2 to 4.

Declarations

Potential competing interests: No potential competing interests to declare.