Answer 99.16% of DocVQA Without Images in QA: Agentic Document Extraction

Nov, 2025

TL;DR

We ran on the DocVQA validation split and got **5,286 correct out of 5,331 (99.16%)**. Of those **45** wrong answers, only **18** are true **parsing** shortcomings. DocVQA is usually used to evaluate vision-language models, but we are pioneering the use of this popular dataset to establish the accuracy of our **Agentic Document Extraction (ADE) Parse API**.

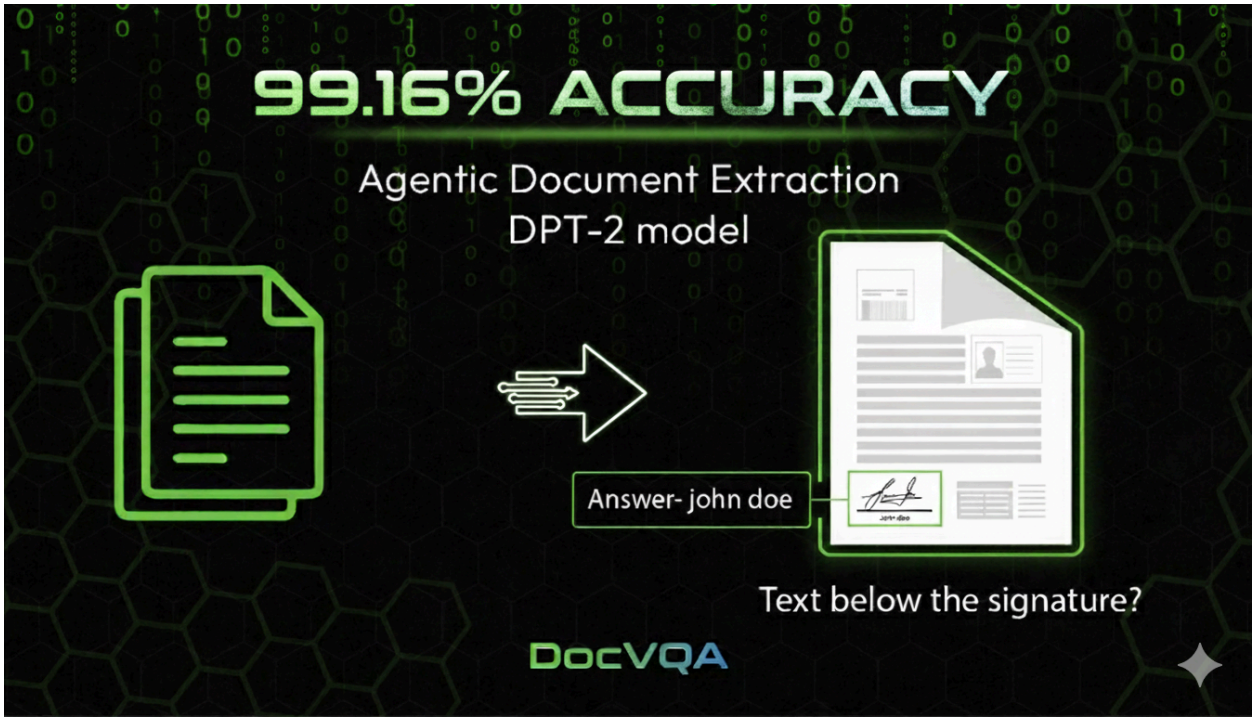
The key takeaway: an LLM can answer **99.16%** of DocVQA questions using **only the parsed API response from ADE**, with **no image access during the QA step**. Our latest offering, ADE with the **Document Pre-trained Transformer 2 (DPT-2) model**, looks at the image once to parse, and it captures the document so completely that the QA step can skip pixels and still be right almost every time.

With **visual grounding** (the bounding boxes and layout that point back to the exact text, tables, figures, and even individual table cells on each page), you **parse once**, then run

unlimited queries against the structured output, and you keep **traceability** for every answer.

We publish **representative hard-document successes**, **all 45 mistakes**, and we provide **reproducible code** for transparency. Benchmarks build confidence; **pair them with your data to bulletproof your analysis**. We urge you to **test ADE on your hardest documents** in the playground and tell us where we can improve together to fully automate your document processing pipeline.

View Successes and Failures | **Access code on GitHub**



Real Business Value from Accurate Parsing

Again, I'd like to drill this key takeaway, when we say we got 99.16% accuracy on DocVQA, we're not testing a vision-language model, We're testing whether our parsed output preserves enough information that a human (or LLM) who never saw the original document but saw the parsed representation can still answer questions correctly.

Think about what that might might mean:

- Parse a document **once**
- Store the structured output
- Answer **millions** of questions from your database
- Save bucks by never re-processing the images

This is how document systems scale. And it's only feasible if your parsing is exceptional. Remember, parsing doesn't just mean answering without images, it is how you run document processing pipelines with control and trust:

- **Cost and latency:**

You will need only a single pass through unstructured modality of images/pdf. Everything after that runs on structured data.

- **Provenance and audit**

Every value you care about links back to the exact span, cell, and page.

- **Human review**

Bounding Box coordinates and Confidence scores can jump your reviewers to the right spots.

- **Search and analytics**

Visually Grounded text indexes cleanly for filters, trends, and BI.

- **Schema agility**

You can always add or change fields anytime without touching pixels again.

- **Privacy and governance.**

This empowers you to redact spans, enforce retention., share structure, and keep your images private.

- **Better Routing**

You can use parsed signals to send the hardest 1–2% to a visual model or a human evaluator.

- **Quality Control and Reuse**

Structured data makes it easier to track parsing quality by section, table, or field over time and reuse the same data for extraction, QA, support, or compliance.

- **RAG and agents**

Finally, as you might have already guessed, RAG and Agents work better over grounded chunks than over raw images.

The Classification of Errors Tells the Real Story

Of the 45 errors, only 18 are genuine parsing failures. The rest are either not the typical focus of a parser or errors that can be removed by downstream prompt engineering to get the exact answer.

Error Type	Count	%	What It Means
Incorrect Parse	13	28.9%	OCR/parsing errors (character confusion, misreads). Some of the images are really tricky due to occlusions and terrible scans, but we are continuously improving and excelling even over the hardest edge cases with each model release
Prompt/LLM Misses	18	40%	Reasoning or interpretation failures. Your downstream heuristics or careful prompt design can easily tackle these misses.
Not ADE Focus	9	20%	Spatial layout questions outside any Parser's core focus
Missed Parse	5	11.1%	Information not extracted during parsing. There are certain cases where the parser misses to extract. Devil is in the details, I recommend you taking a look at these examples to understand why it's missed.
Dataset Issues	18	—	Questionable ground truth (excluded from count and accuracy calculation)

Our VQA Approach is Even Harder

It becomes quickly imperative that getting everything extracted from the document irrespective of the questions asked is harder as compared to standard DocVQA:

Standard VQA systems:

1. Take (image + question)
2. Feed to vision-language model
3. Get an answer

Our approach:

1. Parse document to get the output; you get both a markdown and a JSON representation.
2. Answer **all** questions from md/json **without the image**

Answering Questions

We used ADE Playground to answer questions from the parsed output. The playground pipeline never sees the original images, only refers to the extracted markdown output. For evaluation, we performed the exact string match (case-insensitive) exactly as per the official DocVQA evaluation.

The Journey to 99.16%

Version	Method	Accuracy	Input
Baseline	ADE Playground Chat	95.36%	Markdown output from ADE's DPT-2 model

Final	ADE Playground Chat + **Guided prompting**	**99.16%**	Markdown + Visual Grounding information from DPT-2 model

The improvement came from:

- Using JSON output instead of markdown because our JSON has baked in spatial information (visual grounding) not just extracted text
- Optimized prompt structure for better question interpretation and guiding the playground app to better understand how to leverage the parsed output for some of the tricky documents in the DocVQA dataset, refer to the additional reading section at the end or the **gallery for examples.**

Note: you can replicate the results by choosing an LLM to replace our playground chat feature.

About DocVQA

DocVQA is a question-answering benchmark on real scanned documents from the UCSF Industry Documents Library. Created by researchers at UC San Diego and Allen Institute for AI, it's designed to test document understanding capabilities.

Validation set: 5,349 questions across 1,286 document images

Question types: Factual extraction, spatial reasoning, table understanding

Evaluation: Exact string match (case-insensitive)

Current leaderboard (test set, October 2024):

- Qwen2-VL: 97.25% (with image access)
- Human baseline: ~96-98% (estimated)

Our result: **99.16%** (on validation set using ADE's DPT-2 model)

The next step is to release the results on the test set in the near future. Based on validation performance and the similar statistical distribution of data and complexity of the documents, we expect similar state-of-the-art results.

Complete Transparency: Every shortcoming is accessible

In AI/ML it's widely recognized that benchmarks can be "gamed." Trust is not built by hiding or deceiving so we make sure we strive for utmost transparency by publishing the verifiable misses and the code.

We include:

- Interactive gallery with all 45 errors and 63 out of 5,331 correct answers
- Source code to reproduce everything
- A detailed breakdown for each result, with images, predictions, ground truth, question IDs, and classified shortcomings in the interactive gallery

You can evaluate for yourself whether these errors matter for your use case.

Again, this is about building trust. We want you to see exactly what document parsing can do today, what it can't do, and where the remaining challenges are.

Try It Yourself

Want to see how ADE parsing performs on your documents?

Try Agentic Document Extraction (ADE): va.landing.ai

Questions? Contact us

Join our Discord Community to get instant support and chat with fellow Document AI enthusiasts.