QUBER RESEARCH — VENDOR EVALUATION

Reducto

A new agentic document extraction vendor: what is verifiable, what is marketing, and whether to evaluate it hands-on

Reducto sells an API that parses PDFs and then re-checks its own output with vision models. This document sorts what can be verified about the company, the product, and the benchmarks from what is marketing. It compares Reducto on paper against Landing.ai's ADE (Agentic Document Extraction) platform and against docling, the open-source document parser our pipeline is built on, and it recommends a hands-on evaluation. It assumes no prior knowledge of Reducto. It does assume the reader knows our pipeline's standing concern: extraction failures that arrive silently, as pages or tables missing without an error.

Engineering Reference · Generated 2026-07-31
Primary sources: `reducto.ai`, `docs.reducto.ai`, benchmark publishers · QUE-323

Contents

Audience and Scope

Written for engineers deciding whether Reducto merits evaluation time. Every claim below is tagged with where it comes from, because separating vendor claims from independently verifiable facts is the purpose of the document. Sources were fetched July 30, 2026 and are collected in section 9, cited by number.

1 The Findings That Matter

Four findings decide whether Reducto merits evaluation time.

- **Every benchmark in document extraction is won by the vendor who authored or paid for it.** Reducto wins its own RD-TableBench and the benchmark it commissioned from micro1, an AI evaluation company. LlamaIndex's ParseBench puts LlamaParse first and Reducto third. Unstructured's SCORE-Bench puts Unstructured first. Published rankings cannot pick a vendor here. Only our corpus can. Section 5 documents each benchmark's authorship.
- **The company is real and heavily backed. The accuracy claims are not independently verified.** $108M raised from the venture firms Benchmark and a16z, named customers in finance and legal, a documented product — and not one substantive independent evaluation anywhere we could find. Sections 2 and 5.
- **Their own documentation names the failure mode we care most about.** Reducto concedes that its agentic loop can only refine what the parsing layer detected. A table the parser never recognized cannot be recovered downstream. That is the same silent-gap class we document for docling. The exact sentence is quoted in section 4.
- **Evaluating costs nothing.** 15,000 free credits buy roughly 3,700 pages of their most expensive per-page parse mode, before the optional chart agent's per-chart charge — enough for a full hands-on evaluation with no procurement. What to measure is in section 8.

2 Company and Credibility

The company checks out as a serious, well-funded startup. Adit Abraham and Raunak Chowdhuri founded it in San Francisco in 2023 and took it through Y Combinator W24 [1, 2]. Abraham is the CEO, from MIT, previously a Google product manager on Ads and Search. Chowdhuri is the CTO, from MIT, previously at NVIDIA. Funding totals $108M: a $24.5M Series A led by the venture firm Benchmark in April 2025 and a $75M Series B led by a16z in October 2025, with First Round, BoxGroup, and YC participating [3, 4].

Customer evidence is company-published and finance-heavy, which makes it marketing material until someone confirms it independently. Named customers include Harvey, Scale AI, Rogo, Vanta, JLL, Toast, and Zip [5]. Three testimonials are finance-relevant [5]:

- An investment platform that is also named Benchmark — a customer, not the venture firm that led the Series A — says it runs 3.5M+ pages a year through Reducto and embeds citations from it in the reports it produces.
- LEA, another named customer, sells services to wealth-management firms that each manage $10B+ in assets.
- An unnamed "Global Top 5 Hedge Fund" engineering leader is quoted saying Reducto "helped us parse documents we previously could not".

The widely quoted 99.24% is not a benchmark result. A 99.24% accuracy figure circulates in coverage of Reducto. It comes from the production workload of one healthcare customer, Anterior, on prior-authorization documents. It says nothing about table extraction on financial filings.

Three company-scale numbers rest on sources that cannot be checked. The billion-page processing total and the 6x volume growth between the April and October 2025 rounds are Reducto's own statements. The headcount of about 30 comes from a third-party directory, itself unverified [1].

3 The Product

3.1 The endpoints

Nine endpoints accept 30+ file types: Parse, Extract, Split, Classify, Edit, Pipelines, Generate, Redact, and Translate [6]. Two of them are uncommon among extraction vendors. Edit writes values back into a document, which is form filling. Pipelines chains the endpoints together server-side. Parse settings include an `ocr_system` selector and an `extraction_mode` defaulting to `hybrid`; the pages we retrieved do not say what modes it chooses between [7].

3.2 The parse response

A parse response is a list of chunks, and each chunk holds blocks. Landing.ai's ADE used the same chunks-holding-blocks organization in its v1 API; its v2 replaced chunks with one markdown string plus a tree of pages and blocks [8].

- **What a chunk holds.** Each chunk holds its text as markdown in `content`, plus an `embed` field with a variant of the same content prepared for vector embedding; the documentation says it includes generated summaries.
- **Nine block types** — Title, Section Header, Header/Footer, Text, Table, Figure, Key Value, List Item, Checkbox.
- **Every block records where it came from** — a bounding box normalized to 0–1 as left/top/width/height, plus two page numbers, `page` and `original_page`, which differ only when a page-range option restricted the parse.
- **Confidence on parses and extracts** — each parsed block carries a coarse high/low string and a numeric 0-to-1 `parse_confidence`; fields returned by the Extract endpoint carry an `extract_confidence` as well. Docling offers no equivalent today. Its "conversion confidence" is a roadmap item.
- **Tables in HTML, markdown, JSON, or CSV.** The documentation states that nested tables are supported. No page we could reach states how merged cells and multi-page continued tables are represented — the table-format doc page 404'd on three attempts. That gap must be closed early in any evaluation.

Extract returns JSON matching a schema you supply. Every field in that JSON names its page, its bounding box, the source text it came from, and both confidence scores. Passing a `jobid://` reference instead of a file runs a new schema against a parse already stored, so you do not pay to parse the document again [9].

3.3 Chart extraction

A separate chart agent reads a chart image back into a table of values. It detects axes, tick marks, legends, and data series with specialist models, then extracts the values and verifies them against the image in a loop. By default it aligns read values to the chart's tick marks; a pixel-perfect mode reads positions without that snapping. The documentation does not define the two modes beyond those names [10]. It covers line, area, bar, scatter, stacked bar, and multi-series charts.

Reducto states its own limitations plainly. Individual data points may be off by 5–10% despite visual verification. Dense multi-series and overlapping charts are hard. Charts without printed data labels are hard. No quantitative benchmark is published — the blog shows visual comparisons against frontier models only [10].

3.4 Deployment and integration

The documentation describes the integration and deployment options in enough detail to check them [11]:

- Python, Node, and Go SDKs, plus REST.
- An MCP server.
- Async on every endpoint, with webhooks.
- Self-hosting, documented in depth: a Helm chart onto Kubernetes, hybrid VPC against S3, Azure, or GCS, fully air-gapped installs with offline model delivery, and bring-your-own-LLM so their hosted models are optional.

SOC 2 Type II and HIPAA are claimed. EU and AU data residency exist on the Growth tier.

4 The Agentic Claims

"Agentic" refers to two documented mechanisms here, not just a marketing label. One documented limitation outweighs both of them for us: the loop cannot recover what the parser never detected. The mechanisms first; the limitation follows in full.

- **Agentic OCR** re-reads its own parse output instead of making a single pass. Roughly six vision models, mixing classical computer vision with vision language models, look at the document as an image and check what the parse produced. Reducto claims the loop detects misplaced columns and rows, field-value mismatches, corrupted table structure, missing regions, misclassified blocks, text-flow breaks, and hallucinated artifacts [12].
- **Deep Extract**, in beta and switched on with `deep_extract: true`, extracts, checks the result against the source document, identifies gaps, and extracts again until it meets a quality threshold. It splits the document into sections and dispatches a sub-agent to each. You can state the verification criteria in the system prompt — for example, "iterate until line items sum to the document total" [9].

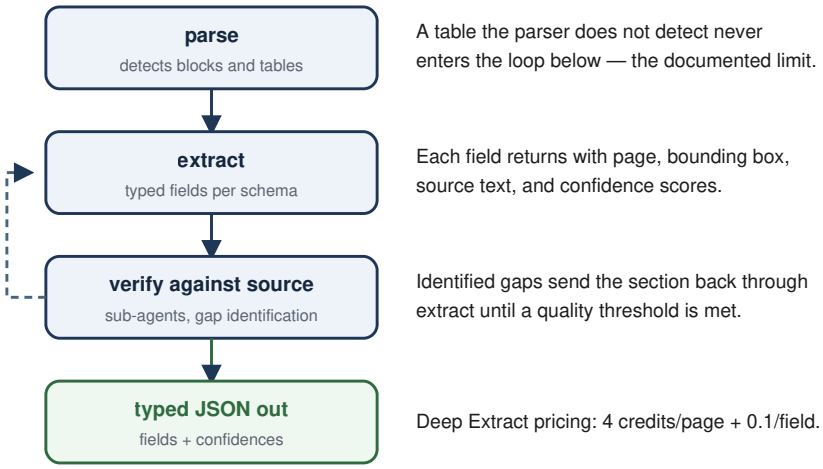

Figure 1 — The Deep Extract loop. Verification starts after parsing, so a table the parser never detected is invisible to every later stage.

THE AGENTIC LOOP CANNOT RECOVER A TABLE THE PARSER MISSED

From their own documentation, verbatim: "Deep Extract can only refine data that the parsing layer successfully detects. If a table isn't recognized during parsing, the extraction loop cannot recover it." The loop repairs misread tables. It cannot recover missed ones. This is the same class of silent gap our `docs/EXTRACTION_PIPELINE.md` documents for docling. Any evaluation must therefore measure

table detection recall separately from cell accuracy. A single reported accuracy rate hides exactly the failure mode that creates liability on a financial document.

5 Benchmarks and Their Authors

Four published benchmarks include Reducto — all we could find that do — and each one is won by its author or sponsor. The table states that finding. Section 5.1 gives the detail behind each row.

Benchmark	Author / funder	Winner	Reducto's result
RD-TableBench (2024)	Reducto	Reducto, 90.2%	1st
LongExtract-Bench (2026)	micro1, commissioned and designed by Reducto	Reducto, 99.6% precision	1st
ParseBench (arXiv)	LlamaIndex	LlamaParse, 84.88%	3rd of 14, 67.8%
SCORE-Bench (2025)	Unstructured	Unstructured	Top recall, third-highest hallucination rate

5.1 What each one actually shows

- **RD-TableBench** — a marketing artifact. It is 1,000 hand-labeled complex table images, labeled by people Reducto employs, scored by a cell-level alignment algorithm. Only part of the evaluation framework is released, so it cannot be reproduced. The blog reports Reducto's own 90.2% average but not the competitors' numbers. After about 20 months it has 14 GitHub stars, 3 commits, and no license [13].
- **LongExtractBench** — a 13-point precision margin over the next-best extraction platform, on a benchmark the winner designed and paid for; suspicion, not confidence, is the right reading. It is 225 documents averaging 358 pages, including financial filings and asset-backed-securities reports, with ground truth drafted by language models and reconciled by human annotators. The publisher discloses, verbatim: "Reducto commissioned this benchmark. Reducto is also one of the systems under test and the top performer on these documents". The publisher also discloses that Reducto authored the methodology and the ground-truth harness [14, 15]. The full field is tabulated below.

- **ParseBench** — the most relevant to our documents despite its authorship, because 75% of its table dataset is financial and insurance documents. It covers ~2,000 human-verified pages across 14 systems, including docling and Landing.ai. Reducto places third overall at 67.8%, behind LlamaParse at 84.88% and Gemini 3 Flash. The authors' own conclusion is that no system is consistently strong across the paper's five scoring dimensions. We did not retrieve the five dimension names or per-dimension scores beyond the table dimension, where LlamaParse scores 90.74% [16].
- **SCORE-Bench** — the one adverse datapoint from outside Reducto's control. Its authorship follows the same pattern as the rest — Unstructured wrote it and wins it — but what it reports about Reducto is adverse, and an adverse result on a competitor's benchmark is the closest this field offers to independent evidence. Reducto's agentic mode scored best in the field on Percent Tokens Found at 0.924, the share of the document's real tokens recovered. It scored 0.124 on Percent Tokens Added, the share of output with no source in the document: the third-highest hallucination rate, against 0.036 for the best. On filings, a fabricated figure is worse than a missed one. Dataset and code are public [17].

The LongExtractBench field as published [14]. A failure is a document run that did not complete:

System	Completed of 225	Precision	Recall	Failure rate	Mean latency
Reducto Deep Extract	225	99.6%	99.6%	0.0%	540s
Extend MAX	210	86.4%	92.7%	3.6%	738s
LlamaExtract Agentic	203	80.0%	77.5%	9.8%	524s
GPT-5.5	198	95.8%	52.7%	12.0%	327s
Datalab Extract Balanced	166	92.8%	33.8%	26.2%	1,609s
Claude Opus 4.8	116	92.0%	70.7%	36.0%	420s
Gemini 3.1 Pro	112	95.8%	48.6%	48.4%	187s

THE PRESS RELEASE OMITS THE DISCLOSURE THE BLOG MAKES

> Reducto's blog does disclose commissioning LongExtractBench. The press release does not. It says only that "micro1 released an independent benchmark," with no mention of the financial relationship anywhere in it. A company confident in its numbers does not need that omission [15].

One more distortion to know about when searching this market: `llms.reducto.ai` is a Reducto-owned subdomain that ranks high in search and reads as neutral comparison content. Landing.ai and LlamaIndex run equivalent subdomains [18].

Independent practitioner evidence is essentially absent. We found no engineering blog, paper, or substantive community thread evaluating Reducto without a commercial stake. Two Hacker News fragments exist. Neither could be retrieved with enough context to count as evidence in either direction.

6 Cost

Credits cost $0.015 each, and 15,000 come free, which is about $225 of usage [19, 20]. The rates that matter to us:

Operation	Credits	Effective
Parse, standard / complex	1 / 2 per page	$0.015 / $0.030 per page
Parse agentic, standard / complex	2 / 4 per page	$0.030 / $0.060 per page
Advanced chart agent	+4 per chart	+$0.060 per chart
Extract, standard	2 per page	$0.030 per page
Deep Extract (beta)	4 per page + 0.1 per field, min 30 per document	$0.060 per page + $0.0015 per field
Split standard / deep	2 / 4 per page	$0.030 / $0.060 per page
Classify	0.5 per page, first 5 pages by default	~$0.0375 for a default 5-page classification

The agentic rows run the review loop described in section 4. Async batch parsing takes 20% off with a 12-hour completion guarantee. Passing a file directly to Extract charges for Parse on top of Extract; referencing a stored parse with `jobid://` avoids the double charge [20].

On our documents, a 150-page 10-K through agentic-complex parse plus Deep Extract runs roughly $18–20, about 1,200 credits. The free tier therefore covers roughly a dozen such full-depth runs, or about 3,700 pages of agentic-complex parsing alone. It is a batch tool, not an interactive one. Deep Extract averaged 540 seconds per document on 358-page documents, with a p95 over half an hour [14].

Three tiers are offered, and Growth and Enterprise pricing is custom [19]:

- **Standard** — 200 concurrent pages.
- **Growth** — 350 concurrent pages, zero data retention, BAA, EU and AU residency.
- **Enterprise** — 500+ concurrent pages, VPC and on-prem deployment, SSO, custom SLA.

7 On Paper Against ADE and Docling

No trustworthy published ranking separates Reducto from ADE. The one benchmark containing both, ParseBench, is authored by a third vendor that wins it (section 5). What can be compared is capability and price. Reducto returns chunks where ADE's v2 API moved to a structure tree. Both ground every field in bounding boxes. Both offer enterprise deployment and zero-data-retention tiers. On price, ADE's credits cost $0.01 to Reducto's $0.015, and its free tier is 1,000 credits to Reducto's 15,000 [21]. Each platform has features the other does not:

- **Only Reducto has** the Edit endpoint, Deep Extract's verification loop, the dedicated chart agent, per-field confidence scores, and the fifteen-times-larger free tier.
- **Only ADE has** one trained model rather than an orchestration of about six, and a published result with public reproduction code: its previous-generation DPT-2 model's 99.16% on DocVQA, a document question-answering benchmark.

Part of that comparison draws on Landing.ai's own competitive page, which is not neutral [18, 21].

Choosing Reducto over docling means paying per page and depending on a vendor. Docling costs nothing per page, runs on our own hardware, and we have tuned it. Reducto is $0.015–0.060 per page, and self-hosting is available only at the Enterprise tier. What Reducto offers that docling does not: per-field confidence today, the agentic repair loop, chart extraction as a paid add-on, and handwriting claims. ParseBench is the one benchmark containing both. Its authors report no dimension win for docling, and we did not retrieve docling's per-dimension scores, so the size of the gap is unknown [16].

8 Assessment

Reducto is real — a real company, a real product with documented depth, and real enterprise deployment. What is not real, yet, is any independent evidence for the accuracy claims. Every benchmark win they promote comes from an evaluation they wrote or paid for, their most-quoted accuracy figure is one customer's production statistic, and the one adverse outside measurement ranks them third-highest in the field for hallucination.

Trust the API reference; do not trust the rankings. The documentation is specific, the limitations sections are candid, and the self-hosting instructions describe verifiable engineering. The accuracy claims are unproven, and the LongExtractBench press release presents a commissioned result as independent.

Evaluate hands-on. The free tier removes every reason not to. 15,000 free credits buy roughly 3,700 pages of agentic-complex parsing, or about a dozen filings through the full parse-plus-Deep-Extract depth (section 6), with no procurement. Three measurements, none of which any published benchmark answers:

- **Table detection recall, separately from cell accuracy.** Their own docs concede the agentic loop cannot recover an undetected table. This is our standing silent-gap concern, and it is the number nobody publishes.
- **The hallucination rate on our filings.** SCORE-Bench measured 0.124 tokens added in agentic mode. On a financial document a fabricated figure is a liability; this single number decides whether Reducto is usable for us regardless of everything else.
- **Footnotes and multi-level headers on a real 10-K.** The documents already in our repository — the 3M 10-K and the Verizon 1Q26 statements — have the table shapes that separate these tools: multi-level headers, footnoted rows, and tables continued across pages. Nothing published tests them.

Ask Reducto early how merged cells and multi-page continued tables are represented. We could not determine either from the documentation, as section 3.2 records.

9 Sources

All sources fetched July 30, 2026. Bracketed numbers in the text refer to this table.

#	Source
1	Y Combinator company page — `ycombinator.com/companies/reducto`; headcount via startupintros.com (unverified)
2	a16z investment announcement — `a16z.com/announcement/investing-in-reducto/`
3	Series B post — `reducto.ai/blog/reducto-series-b-funding`
4	Series A coverage — Fortune, April 25, 2025
5	Customers and testimonials — `reducto.ai/customers`, `reducto.ai/extract`
6	API overview — `docs.reducto.ai/overview`
7	Parse API reference — `docs.reducto.ai/api-reference/parse`
8	Parse response format — `docs.reducto.ai/parsing/response-format`
9	Deep Extract — `docs.reducto.ai/configs/extract/deep-extract`
10	Chart extraction — `reducto.ai/blog/reducto-chart-extraction`, `docs.reducto.ai/parsing/chart-extraction`
11	Deployment options — `docs.reducto.ai/onprem/enterprise_deployment_options`, `docs.reducto.ai/llms.txt`, AWS Marketplace listing
12	Agentic OCR — Reducto product pages and blog
13	RD-TableBench — `reducto.ai/blog/rd-tablebench`, `github.com/reductoai/rd-tablebench`
14	LongExtractBench — `micro1.ai/benchmark/long-extraction`; public 50-document subset on HuggingFace
15	Reducto's benchmark post and the PR Newswire release, July 1, 2026 — `reducto.ai/blog/reducto-leads-benchmark-complex-document-extraction`

#	Source
16	ParseBench — `arxiv.org/abs/2604.08538` (all authors run-llama.ai)
17	SCORE-Bench — `unstructured.io/blog/introducing-score-bench-an-open-benchmark-for-document-parsing`
18	Vendor-run comparison content, neither neutral — `llms.reducto.ai` and `landing.ai/llms/landingai-ade-vs-reducto-document-extraction-platform-comparison`
19	Pricing — `reducto.ai/pricing`
20	Credit usage — `docs.reducto.ai/reference/credit-usage`
21	ADE pricing — `docs.landing.ai/ade/ade-pricing`