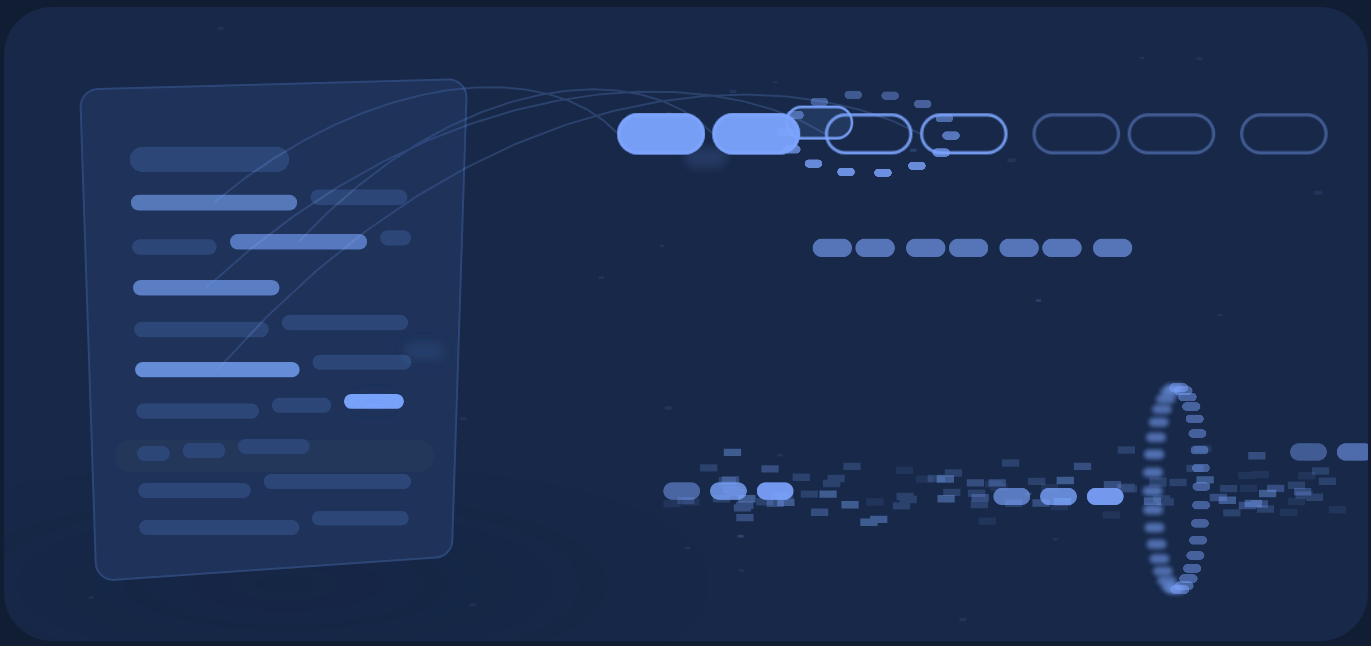
Rillor

METHODS NOTE

How to test document extraction accuracy before you buy

A pilot plan for LLM document extraction: your own labeled documents, a scoring rule per field, recall where omissions matter, table structure scores and confidence checked against errors.



7 November 2026 Document and regulation intelligence rillor.com/insights/testing-document-extraction-accuracy

Rillor

Contents

01 Summary

02 Background

03 Method

04 Findings

05 Implications

06 Limits of this analysis

07 Sources

Summary

- Document extraction accuracy depends on the scoring rule, the documents and the unit counted. The same human answers score 0.981 ANLS and 94.36% exact match in DocVQA [1]. The same model outputs score 57.8 on field-level F1 and 84.4 on a document-level tree similarity [10].
- A pilot sample should come from the buyer's own documents, with unseen layouts reported apart. A published baseline scored F1 of 0.524 on unseen layouts and 0.742 on well-seen ones [9]. Public test sets carry label errors [32] and may sit in a model's training data [33].
- Each field type needs a written scoring rule. String similarity scores a near-miss number as zero and penalizes a correct date in another format [3, 4], and published rule sets disagree on whether a number with trailing zeros, such as "4.0" or "1.00", matches the plain number [6, 12].
- Where omissions cost most, read precision at a fixed recall, as CUAD does [11]. Tables need a structure score such as TEDS, with simple and complex tables reported apart [13, 14]. Page parsing scores vary by document type and by benchmark version [16, 17].
- Confidence scores should be tested against errors before they route fields to review. In one single-author study of 267 invoice fields, a model's own confidence measures reached ROC AUC of 0.692 to 0.744, and a classifier that adds a second extraction call and OCR, image and layout features reached 0.928 [19].
- Every pilot error rate needs an interval. Zero errors in 100 fields still allows a true error rate up to 3.62% (derived) [29]. NIST's example sampling plan accepts a batch with 10% defects 22.3% of the time [27].

Background

Document extraction turns pages into fields: an invoice number, a due date, a total, a clause, the rows of a table. Tools range from trained layout models to large language models prompted with a schema [6, 19].

A single accuracy percentage depends on choices made before anyone counts. DocVQA reports human performance on its test split as 0.981 ANLS and 94.36% exact-match accuracy, from the same answers (2020) [1]. One cloud provider's document-processing service reports no accuracy metric, because optional fields and labels with several values make accuracy "less meaningful" [12]. Another provider writes that "Reported accuracy scores can be inconsistent when the analyzed documents differ from documents used in training" [23].

Public scores also travel poorly. The VRDU authors write that results on two widely used receipt datasets might suggest "a solved problem", while a dataset reflecting practical settings "shows that there is much room for improvement" (2023) [6]. The organizers of the ICDAR 2019 receipt competition noted that a 99% business requirement was out of reach for every OCR method entered, and only one key-information method passed 90%, at 90.49% [8].

This note is for buyers and engineers running a pilot of a model-based or LLM-based document extraction tool. It sets out a scoring method they can defend before trusting a vendor's figure, and a way to size review after go-live.

Method

What was examined. The published definition of each measure used to score extraction: exact match, token F1, ANLS and its extension for structured output, type-aware matching, field-level and document-level scores, span matching with precision at fixed recall, average precision, TEDS, page parsing scores, calibration and risk-coverage. Each is checked for what counts as a match and how far it moves on a published example. The note also covers the statistics a pilot needs: intervals, sample sizes, a paired test for two tools and acceptance sampling.

Sources. Thirty-five sources, each opened and read for the passages cited: 17 peer-reviewed conference papers, 2 competition reports, 5 preprints (one an extended workshop paper), a benchmark repository, 7 sections of the NIST/SEMATECH e-Handbook of Statistical Methods and documentation from 3 cloud providers. Several papers have authors at companies [3, 6, 7, 9, 10, 13, 15, 19].

How. Each measure is described from the paper that introduced it. Figures marked "derived" are arithmetic on a source's own numbers or formula. Intervals use NIST's exact binomial method [29], checked against NIST's worked example. No product or model is ranked or named.

Where sources disagree. The confidence study gives its failure rate as 26% and 26.6%, and its base rate as 73.3% and 73.4% [19]. The olmOCR paper gives its PDF count as 1,400, 1,402 and 1,403 [18]. NIST prints one average outgoing quality limit as 0.372 and as 0.0372 [27]. This note uses the figure each source's own data supports.

Findings

1. Building the labeled sample: document mix and unseen layouts

Use the buyer's own documents. Sainz and colleagues describe a model trained on a benchmark's test split and then evaluated on it, and state that contamination "causes an overestimation of the performance" [33]. Documents the vendor has not seen, with labels held back until scoring, avoid the question.

Hold out layouts. DocILE's test set is 25% zero-shot layouts unseen in training, 25% few-shot layouts with up to three training documents and 50% many-shot layouts [9]. Its text-only baseline scored key-information F1 of 0.524, 0.620 and 0.742 on the three groups (2023) [9]. In VRDU's Registration Forms at 200 training documents, micro-F1 on unseen templates was 13 to 27 percentage points below mixed templates across four models (derived from its Table 4, and the paper's text gives 13 to 17 for three of them) [6]. One provider asks for digital and scanned PDFs in training and warns that documents that look alike to people can look different to a model [23].

Sample for diversity. OmniDocBench clustered pages from over 200,000 PDFs by visual features, sampled 6,000 and balanced them by page type, layout and language down to 981 pages [16]. PubTabNet drew 5,000 tables with spanning cells and 5,000 without, since a raw sample would be "strongly biased towards tables without spanning cells" [13]. A buyer can group documents by sender, template or scan quality and draw from every group.

Include absent fields. DUDE's "None" answer type "demands that the model correctly identifies that the answer cannot be provided" [4]. ANLS* penalizes any answer where the truth is None [3]. One provider notes that a model can return "an empty field with high confidence" [23]. Without absent fields, a pilot cannot measure invented values.

Check the labels. Kleister's NDA labels reached Cohen's kappa of 0.971 after a super-annotator's check, and its Charity test set reached 0.9 on 100 documents labeled twice [7]. Northcutt and colleagues estimate at least 3.3% label errors on average across ten benchmark test sets (2021) and note that practitioners' own data is often noisier [32]. Each disagreement between tool and label goes to a second reviewer before it counts against the tool.

Budget for labeling. DocILE's 6,680 documents took about 2,500 annotator hours, or about 22 minutes each (derived) [9].

2. A scoring rule for each field: exact, normalized, tolerance and ANLS

Exact match. In the ICDAR 2019 receipt competition an extracted text counted as correct only "if both submitted content and category of the extracted text matches the groundtruth" [8]. SQuAD's metrics first "ignore punctuations and articles" [5]. One provider's optional fuzzy matching normalizes whitespace, punctuation at either end, case and currency symbols [12].

ANLS. ST-VQA defines ANLS as the mean over questions of the maximum similarity to any accepted answer. Similarity is one minus the normalized Levenshtein distance, set to zero once the distance reaches 0.5 [2]. An answer more than half wrong is treated as "the wrong text selected from the options" [2]. DocVQA made ANLS primary because exact match "awards a zero score even when the prediction is only a little different from the target answer" [1].

Where string similarity fails. ANLS* scores 0.2 against 0.199999999 as 0.0, and "31.12.2023" against "31.Dec 2023" as 0.58. A completely wrong answer scores 0, the same as giving no answer [3]. DUDE explains its human baseline of 74.76 ANLS by answer format: "62% vs. 62 (0.67 ANLS), 1958-04-29 vs. 4-29-58 (0 ANLS)" [4]. Partial credit also rewards a wrong street number, which VRDU treats as a mismatch [6].

Type-aware matching. VRDU uses "different matching functions for each entity according to its data type": prices become numbers and dates are parsed before comparison [6]. Kleister writes dates in ISO 8601 and found that about 5% of income and spending values need their scale inferred from context [7]. Rule sets disagree. VRDU treats "4" and "4.0" as equal [6]. One provider states that "1 and 1.00 don't match, even when fuzzy matching is enabled" [12].

Structured output. ANLS* extends ANLS to the dictionaries that extraction tools return. In its examples a missing key scores 0.5, an invented key 0.67 and a missing list element 0.5 [3].

Field type	Example rule	What it guards against	Basis
Identifiers, codes, addresses	Exact match after normalizing case, whitespace and punctuation	Partial credit for a wrong digit	[6, 8, 12]
Amounts	Parse to a number, compare value, currency and scale, with any tolerance written down	String scores on numbers, a missing scale	[3, 6, 7]
Dates	Parse, then compare as dates	Format differences scored as errors	[4, 6, 7]
Names and free text	Normalized exact match, or ANLS where OCR noise is expected	Zero credit for small OCR slips	[1, 2, 5]
Lists and line items	Match items one to one, count missing and extra items	Invented or dropped rows	[3, 9]
Fields absent from the document	Any returned value is an error	Invented values	[3, 4]

Fix the rules before seeing any tool's output, so that no rule is tuned to one tool's format.

3. Field-level and document-level accuracy

Two scores from the same outputs. In one OCR-free extraction study, field-level F1 fails a field "Even if a single character is missed" [10]. A second measure scores the whole document as a tree: max(0, 1 - TED(pr, gt) / TED(empty, gt)), where TED is the tree edit distance [10].

Dataset (2022)	Field-level F1	Tree-based accuracy	Gap (derived)
Receipts (CORD)	84.1	90.9	6.8
Tickets	94.1	98.7	4.6
Business cards	57.8	84.4	26.6
Receipts (private set)	78.6	88.6	10.0

Source: [10].

Counting rules inside F1. In one provider's evaluation, a wrong value against a matching annotation counts as one false positive and one false negative, and all-label metrics are weighted by how often each label appears [12]. VRDU contrasts micro-F1, which "weighs every instance of an entity equally", with macro-F1, the mean of each entity's F1 [6]. Common fields dominate a micro average.

Fields differ. On Kleister's NDA set, five baselines scored F1 of 45.73 to 60.33 on the contract term and 92.87 to 96.50 on jurisdiction, where annotators agreed 95% and 100% of the time. On its charity set they scored 64.43 to 74.20 on spending and 95.93 to 96.80 on the report date (2021) [7]. Every model did worse on longer documents [7].

The complete-record rate. If each of 20 fields is right 98% of the time and errors are independent, 0.98^20 = 66.8% of records are fully right, and 81.8% at 99% per field (derived from the binomial formula in [27]). Errors share causes such as unreadable source material and OCR noise [7, 19], so they cluster by document. Count the complete-record rate directly on the pilot.

4. Spans and omissions: precision, recall and precision at fixed recall

CUAD's task, for 41 labels, is "finding needles in a haystack" [11]. With most clauses unlabeled, its authors use precision and recall, which "are responsive to class imbalance" [11].

Matching. CUAD lowercases, strips punctuation, splits on spaces and computes the Jaccard similarity of the two word sets. A prediction matches at 0.5 or more [11]. For the Parties label it also accepts a labeled segment that is a substring of a prediction, and records the exception [11].

Precision at fixed recall. Moving the confidence threshold traces precision against recall. CUAD reports precision at 80% and 90% recall and area under the precision-recall curve [11]. Its authors write that "recall is more important than precision since CUAD is about finding needles in haystacks", and that at about 30% precision "a lawyer would need to read through about 2 irrelevant clauses for every 1 relevant clause" [11]. The buyer fixes the recall it needs and reads off precision, which sets the review load.

Default thresholds. One provider sets the F1-maximizing threshold by default, and a higher threshold gives "higher precision" and "lower recall" [12]. Predictions below it are ignored "even if the prediction is correct" [12]. F1 weighs a missed field and a wrong field equally, so a buyer whose omissions cost more sets the threshold from recall.

Average precision and line items. DocILE requires every instance of a field type to be found and uses average precision as its main metric [9]. Adding predictions scored below all others "can never decrease the metric" [9], so report precision and recall at the operating threshold too.

5. Tables scored as structure with TEDS

The older metric. PubTabNet's authors note that adjacency-relation F1 "cannot detect errors caused by empty cells and misalignment of cells beyond immediate neighbors" and gives no partial credit for cell content [13].

TEDS. A table becomes an HTML tree whose leaves are cells with colspan, rowspan and content. Insertions and deletions cost 1. A cell-for-cell substitution costs 1 when the two cells' row or column spans differ, and otherwise a normalized Levenshtein comparison of their contents. Any other substitution costs 1. TEDS = 1 - EditDist(Ta, Tb) / max(|Ta|, |Tb|), averaged over tables [13]. The ICDAR 2021 competition used the same definition [14].

Perturbation (2019)	TEDS	Adjacency relation F1
90% of the cells in the first row shifted	34.9%	80.3%
10% of cell content perturbed	93.2%	19.1%

Source: [13].

As content perturbation rose from 10% to 90%, TEDS fell steadily from 90% to 40% [13].

Simple and complex tables. In the ICDAR 2021 table task, 30 teams were scored on 9,064 tables [14]. The nine listed teams fell within 1.52 TEDS points of each other overall, and each scored 2.6 to 4.8 points lower on complex tables than on simple ones (derived from the results table) [14].

Sensitivities. GriTS scores a table "directly in its natural form as a matrix" [15]. Its authors show a content-only TEDS variant reacting differently to missing columns than to missing rows, and note that position-based comparison can turn one segmentation mistake into "a disproportionate penalty" [15]. Report structure and content apart, and add unit checks of the kind olmOCR-Bench uses: the output "contains a table with a cell with a given value, and that its neighboring cells have certain properties" [18].

6. Page parsing before extraction, by document type

Parsing comes first. Kleister stresses "the importance of using a PDF processing tool of good quality" and a dedicated PDF tool for born-digital files [7]. On its mixed charity set, changing the OCR engine moved five models' F1 by 2.07 to 8.30 points (derived from its Table 4) [7]. One provider notes that text-read confidence "doesn't affect the confidence of the key/value extraction results, so you should check both" [23].

By document type. OmniDocBench (paper version 2, March 2025) covers nine document sources, 19 layout categories and 15 attribute labels. It scores text and reading order with normalized edit distance and tables with TEDS [16]. Its authors report four patterns [16]:

- Pipeline tools perform well on academic papers and financial reports.
- General vision-language models generalize better to slides, handwritten notes and fuzzy scans.
- Most vision-language models fail on dense newspapers.
- Every model drops on multi-column and complex layouts, and almost all score higher on English than on Chinese pages.

Versions. Version 1.5 (September 2025) raised newspaper and note images from 72 to 200 DPI, added 374 pages and redefined the overall metric. Version 1.6 (April 2026) added 296 pages and a new matching algorithm, and version 1.7 followed on 30 April 2026 [17]. The benchmark now describes 1,651 pages and 10 document types [17], against 981 pages in the paper [16]. A parsing score needs its version.

Pass or fail checks. olmOCR-Bench uses "simple, natural binary rules, like software unit tests": text present, headers and footers absent, passages in order, table cells beside the right neighbors. It has about 7,010 tests over about 1,400 PDFs, reported with bootstrapped 95% intervals [18].

7. Confidence scores that predict errors, and routing fields to review

Vocabulary. Guo and colleagues define calibration by example: "given 100 predictions, each with confidence of 0.8, we expect that 80 should be correctly classified" [20]. A reliability diagram plots accuracy against confidence, and expected calibration error averages the gap across bins. They found modern neural networks poorly calibrated (2017) [20]. Geifman and El-Yaniv define coverage as the share of inputs accepted, and the risk-coverage curve as the error rate among accepted inputs against coverage [21].

Vendor confidence is testable. One provider states that a field confidence of 0.95 means the prediction is "likely correct 19 out of 20 times", and that "not all document fields return a confidence score" [23]. A reliability diagram on the pilot tests the claim [20].

One study of LLM extraction. Kumar's paper is a single-author study by an author at Perfios Software Solutions, an extended workshop paper posted in June 2026 [19]. It ran one hosted model on 1,339 invoice fields with 55 field types, 1,072 for training and 267 for testing, and the model failed on 26.6% of them. It covers English business documents [19].

Confidence measure (267 test fields, 2026)	ROC AUC	Area under risk-coverage curve
Mean token log-probability	0.705	0.145
Model's stated confidence	0.692	0.146
Agreement across five samples	0.744	0.138
OCR features alone	0.896	0.079
Trained classifier, all features	0.928	0.043

Source: [19]. All features: model uncertainty, agreement with a second extraction call, OCR, image quality and layout. A lower risk-coverage area is better.

At a 0.5 threshold the three model-only measures became "near-all-positive classifiers" [19]. One OCR feature checks whether the value appears exactly in the page's OCR text, which flags values with no grounding in the page [19]. The author traces many errors to "failures the model cannot observe: unreadable source material, ambiguous layouts, OCR noise" [19].

At 80% coverage, accepted fields were 99.1% right against a 73.4% base rate [19]. That is about 2 errors in about 214 fields, with an exact 95% interval of 0.11% to 3.34% (derived) [29]. Isotonic recalibration cut expected calibration error from 0.199 to 0.034 [19]. Numeric fields were well calibrated and free-text fields overconfident at high scores [19]. The test design carries over. The figures belong to one dataset and one model.

Stated confidence in general. Xiong and colleagues found that language models stating their own confidence "tend to be overconfident", with values mostly between 80% and 100% and "typically in multiples of 5" (ICLR 2024) [22]. Their tasks were reasoning and knowledge questions.

Routing triggers. One provider's human-review service starts review on key confidence, specified keys that are missing, a confidence range, or "a sample of forms" sent at random [24]. The random sample catches errors made with high confidence.

Before routing on confidence, a pilot can:

1. Plot observed against stated accuracy by confidence bin [20].
2. Plot the error rate among accepted fields against coverage [21].
3. Set the threshold from the error rate the business accepts, with an exact interval [29].
4. Repeat both plots by field type [19].
5. Keep a random sample in review whatever the confidence [24].

8. Acceptance sampling for ongoing review

What sampling can say. NIST calls acceptance sampling "the middle of the road" between no inspection and 100% inspection [25]. It states that "the main purpose of acceptance sampling is to decide whether or not the lot is likely to be acceptable, not to estimate the quality of the lot" [25].

The terms. A single plan (n, c) inspects n items and rejects the lot when it finds more than c defects [26]. The acceptable quality level is the producer's baseline. The lot tolerance percent defective is the level the consumer wants accepted rarely, and consumer's risk is the chance of accepting it. The operating characteristic curve plots the probability of acceptance against the defect rate [26].

Lot defect rate	Probability of acceptance, n = 52, c = 3
1%	0.998
2%	0.980
5%	0.739
7%	0.502
10%	0.223
12%	0.115

Source: NIST e-Handbook 6.2.3.2 [27].

The plan accepts a lot with 2% defects 98.0% of the time and a lot with 10% defects 22.3% of the time [27]. A design needs two points: the producer's risk at the acceptable level and the consumer's risk at the tolerance level [27].

Rectifying review. Where inspection does no harm, rejected lots are inspected in full and corrected [26]. In NIST's example, with lots of 10,000, the (52, 3) plan and 3% incoming defects, average outgoing quality is 0.02775 and average total inspection is 753 items per lot [27]. A batch of extracted records that fails its sample can go to full review the same way.

Switching rules. Mil. Std. 105D (1963, later ANSI Z1.4 and ISO 2859) moves between normal, tightened and reduced plans by rule, tightening when quality slips [28]. A review plan can raise or lower the share of fields checked the same way.

Model changes reset the baseline. Chen, Zaharia and Zou found that "the behavior of the “same” LLM service can change substantially in a relatively short amount of time", with one task's accuracy at 84% in the March 2023 version and 51% in June [34]. Extraction was outside their study. The finding still supports re-running the test set after any model or prompt change, then restarting at normal inspection.

9. Sizing the pilot and comparing two tools

NIST's handbook presents the Wilson interval, whose lower limit cannot go negative as the common normal-approximation interval's can, and an exact binomial method for small samples [29].

Errors found	Fields checked	Exact 95% interval (derived)
0	100	0% to 3.62%
0	300	0% to 1.22% (one-sided upper limit 0.99%)
5	500	0.33% to 2.32%
10	1,000	0.48% to 1.83%

Computed with the exact method in [29].

NIST's sample-size example detects a rise from 10% to 20% defective, with a one-sided 5% test and a 10% risk of missing the change, using about 102 items, or 112 with a continuity correction [30]. The normal approximation needs at least 500 trials at a 1% rate [35]. These figures treat fields as independent, so clustered errors call for more documents.

Raschka, citing Dietterich, recommends McNemar's test for two models scored on one test set [31]. Only the cells where one model is right and the other wrong, B and C, "tell us how the models differ", and the statistic is (B - C)^2 / (B + C) [31]. In his example the models score 99.7% and 99.6% on 10,000 cases in two scenarios. The p-value is 0.0039 with B = 11 and C = 1, and 0.1138 with B = 25 and C = 15 [31].

Implications

For buyers. A defensible pilot:

1. Draws the sample from your own documents, grouped by sender, template and scan quality, with labels held back until scoring [9, 16, 33].
2. Sets aside layouts the vendor has not seen and reports them apart [6, 9].
3. Includes documents where required fields are absent [3, 4].
4. Labels a subset twice and reviews every tool-label disagreement [7, 32].
5. Fixes a scoring rule for each field type before any output is seen [6, 12].
6. Reports precision, recall and F1 per field, the complete-record rate, and results by document type and layout exposure [6, 10, 12].
7. Reads precision at the recall target where omissions cost most [11].
8. Scores tables with TEDS, simple and complex apart, with unit checks on key cells [13, 14, 18].
9. Tests the confidence score against errors before routing on it [19, 20, 21].
0. Gives every error rate an exact interval and compares tools with McNemar's test on the same documents [29, 31].

A vendor's figure is comparable to the buyer's own only when it states the documents, the scoring rule, the unit counted and the benchmark version [1, 10, 17].

For engineers running the pilot. Version the scoring code apart from the extraction code. Record the model, prompt and parser version with every run, since a hosted model can change between versions [34] and a benchmark can change its formula [17]. Store each prediction with its confidence and the page text it came from, so a value-in-text check and a reliability diagram can be run later [19, 20].

For review and operations teams. Route fields on a confidence band, a missing required field and a random sample [24]. Use the random sample to estimate the error rate among auto-accepted fields, with an interval [29]. Pick a sampling plan from its operating characteristic curve, send failed batches to full review and adjust sampling with switching rules [26, 27, 28]. In Rillor's document and regulation intelligence service, a person reviews every record before it is used: the LLM proposes each field beside the passage it came from, a reviewer accepts, corrects or rejects it, and corrections go into the test set for the next extraction run, so a repeated error is caught before review.

Limits of this analysis

- This note covers how to measure extraction. It ranks no product or model and sets no required accuracy level, since no source reviewed sets one for any document type.
- Many benchmark results date from 2016 to 2023. They are used for what they show about metrics and document types, never as current accuracy levels.
- Evidence on confidence for LLM extraction rests on one single-author study with 267 test fields, one model and English business documents [19]. The general calibration evidence comes from other tasks [20, 22].
- Several sources are written by company staff [3, 6, 7, 9, 10, 13, 15, 19], and three are provider documentation that can change after the viewing date [12, 23, 24].
- The record-level and interval figures marked "derived" treat fields as independent. Errors that cluster by document change both, and no source here measures by how much.
- No source reviewed measures the accuracy or speed of reviewers who check a field beside its source passage, or compares tools with manual data entry.
- OCR character error rates, handwriting-heavy forms and non-Latin scripts are outside this note, apart from OmniDocBench's finding that almost all systems scored higher on English than on Chinese pages [16].

Sources

1. DocVQA: A Dataset for VQA on Document Images. Minesh Mathew, Dimosthenis Karatzas and C.V. Jawahar, WACV 2021, arXiv 2007.00398, version 3, 5 January 2021. https://arxiv.org/abs/2007.00398

2. Scene Text Visual Question Answering. Ali Furkan Biten, Rubèn Tito, Andres Mafla and others, ICCV 2019, arXiv 1905.13648, version 2, 16 October 2019. https://arxiv.org/abs/1905.13648

3. ANLS* - A Universal Document Processing Metric for Generative Large Language Models. David Peer, Philemon Schöpf, Volckmar Nebendahl and others, arXiv 2402.03848, version 10, 22 April 2025. https://arxiv.org/abs/2402.03848

4. Document Understanding Dataset and Evaluation (DUDE). Jordy Van Landeghem, Rubén Tito, Łukasz Borchmann and others, ICCV 2023, arXiv 2305.08455, version 3, 11 September 2023. https://arxiv.org/abs/2305.08455

5. SQuAD: 100,000+ Questions for Machine Comprehension of Text. Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev and Percy Liang, EMNLP 2016, arXiv 1606.05250, version 3, 11 October 2016. https://arxiv.org/abs/1606.05250

6. VRDU: A Benchmark for Visually-rich Document Understanding. Zilong Wang, Yichao Zhou, Wei Wei and others, KDD 2023, arXiv 2211.15421, version 3, 16 September 2023. https://arxiv.org/abs/2211.15421

7. Kleister: Key Information Extraction Datasets Involving Long Documents with Complex Layouts. Tomasz Stanisławek, Filip Graliński, Anna Wróblewska and others, ICDAR 2021, arXiv 2105.05796, 12 May 2021. https://arxiv.org/abs/2105.05796

8. ICDAR2019 Competition on Scanned Receipt OCR and Information Extraction. Zheng Huang, Kai Chen, Jianhua He and others, competition report, arXiv 2103.10213, 18 March 2021. https://arxiv.org/abs/2103.10213

9. DocILE Benchmark for Document Information Localization and Extraction. Štěpán Šimsa, Milan Šulc, Michal Uřičář and others, ICDAR 2023, arXiv 2302.05658, version 2, 3 May 2023. https://arxiv.org/abs/2302.05658

10. OCR-free Document Understanding Transformer. Geewook Kim and others, ECCV 2022, arXiv 2111.15664, version 5, 6 October 2022. https://arxiv.org/abs/2111.15664

11. CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review. Dan Hendrycks, Collin Burns, Anya Chen and Spencer Ball, NeurIPS 2021 Datasets and Benchmarks Track, arXiv 2103.06268, version 2, 8 November 2021. https://arxiv.org/abs/2103.06268

12. Evaluate performance, Document AI documentation. Google Cloud, last updated 20 October 2026, viewed 24 October 2026. https://cloud.google.com/document-ai/docs/workbench/evaluate

13. Image-based table recognition: data, model, and evaluation. Xu Zhong, Elaheh ShafieiBavani and Antonio Jimeno Yepes, ECCV 2020, arXiv 1911.10683, version 5, 4 March 2020. https://arxiv.org/abs/1911.10683

14. ICDAR 2021 Competition on Scientific Literature Parsing. Antonio Jimeno Yepes, Peter Zhong and Douglas Burdick, ICDAR 2021, arXiv 2106.14616, 8 June 2021. https://arxiv.org/abs/2106.14616

15. GriTS: Grid table similarity metric for table structure recognition. Brandon Smock, Rohith Pesala and Robin Abraham,

ICDAR 2023, arXiv 2203.12555, version 3, 23 May 2023. https://arxiv.org/abs/2203.12555

16. OmniDocBench: Benchmarking Diverse PDF Document Parsing with Comprehensive Annotations. Linke Ouyang, Yuan Qu, Hongbin Zhou and others, CVPR 2025, arXiv 2412.07626, version 2, 25 March 2025. https://arxiv.org/abs/2412.07626

17. OmniDocBench repository README, updates through version 1.7. opendatalab on GitHub, viewed 24 October 2026. https://github.com/opendatalab/OmniDocBench

18. olmOCR: Unlocking Trillions of Tokens in PDFs with Vision Language Models. Jake Poznanski, Aman Rangapur, Jon Borchardt and others, Allen Institute for AI, arXiv 2502.18443, version 3, 2 July 2025. https://arxiv.org/abs/2502.18443

19. Beyond Logprobs: A Multi-Signal Confidence Engine for LLM-Based Document Field Extraction. Nitesh Kumar, extended version of a paper accepted at the RobustifAI Workshop, IJCAI-ECAI 2026, arXiv 2606.24420, 23 June 2026. https://arxiv.org/abs/2606.24420

20. On Calibration of Modern Neural Networks. Chuan Guo, Geoff Pleiss, Yu Sun and Kilian Q. Weinberger, ICML 2017, arXiv 1706.04599, version 2, 3 August 2017. https://arxiv.org/abs/1706.04599

21. Selective Classification for Deep Neural Networks. Yonatan Geifman and Ran El-Yaniv, NIPS 2017, arXiv 1705.08500, version 2, 1 June 2017. https://arxiv.org/abs/1705.08500

22. Can LLMs Express Their Uncertainty? An Empirical Evaluation of Confidence Elicitation in LLMs. Miao Xiong, Zhiyuan Hu, Xinyang Lu and others, ICLR 2024, arXiv 2306.13063, version 2, 17 March 2024. https://arxiv.org/abs/2306.13063

23. Interpret and improve accuracy and confidence scores, Document Intelligence documentation. Microsoft Learn, dated 8 April 2026, updated 19 October 2026, viewed 24 October 2026. https://learn.microsoft.com/en-us/azure/ai-services/document-intelligence/concept/accuracy-confidence?view=doc-intel-4.0.0

24. Use Amazon Augmented AI with Amazon Textract. Amazon SageMaker AI Developer Guide, Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/sagemaker/latest/dg/a2i-textract-task-type.html

25. NIST/SEMATECH e-Handbook of Statistical Methods, 6.2.1 What is Acceptance Sampling? National Institute of Standards and Technology, viewed 24 October 2026. https://www.itl.nist.gov/div898/handbook/pmc/section2/pmc21.htm

26. NIST/SEMATECH e-Handbook of Statistical Methods, 6.2.2 What kinds of Lot Acceptance Sampling Plans (LASPs) are there? National Institute of Standards and Technology, viewed 24 October 2026. https://www.itl.nist.gov/div898/handbook/pmc/section2/pmc22.htm

27. NIST/SEMATECH e-Handbook of Statistical Methods, 6.2.3.2 Choosing a Sampling Plan with a given OC Curve. National Institute of Standards and Technology, viewed 24 October 2026. https://www.itl.nist.gov/div898/handbook/pmc/section2/pmc232.htm

28. NIST/SEMATECH e-Handbook of Statistical Methods, 6.2.3.1 Choosing a Sampling Plan: MIL Standard 105D. National Institute of Standards and Technology, viewed 24 October 2026. https://www.itl.nist.gov/div898/handbook/pmc/section2/pmc231.htm

29. NIST/SEMATECH e-Handbook of Statistical Methods, 7.2.4.1 Confidence intervals. National Institute of Standards and Technology, viewed 24 October 2026. https://www.itl.nist.gov/div898/handbook/prc/section2/prc241.htm

30. NIST/SEMATECH e-Handbook of Statistical Methods, 7.2.4.2 Sample sizes required. National Institute of Standards and Technology, viewed 24 October 2026. https://www.itl.nist.gov/div898/handbook/prc/section2/prc242.htm

31. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. Sebastian Raschka, arXiv 1811.12808, version 3, 11 November 2020. https://arxiv.org/abs/1811.12808

32. Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks. Curtis G. Northcutt, Anish Athalye and Jonas Mueller, NeurIPS 2021 Datasets and Benchmarks Track, arXiv 2103.14749, version 4, 7 November 2021. https://arxiv.org/abs/2103.14749

33. NLP Evaluation in trouble: On the Need to Measure LLM Data Contamination for each Benchmark. Oscar Sainz, Jon Ander Campos, Iker García-Ferrero and others, Findings of EMNLP 2023, arXiv 2310.18018, 27 October 2023. https://arxiv.org/abs/2310.18018

34. How is ChatGPT's behavior changing over time? Lingjiao Chen, Matei Zaharia and James Zou, arXiv 2307.09009, version 3, 31 October 2023. https://arxiv.org/abs/2307.09009

35. NIST/SEMATECH e-Handbook of Statistical Methods, 7.2.4 Does the proportion of defectives meet requirements? National Institute of Standards and Technology, viewed 24 October 2026. https://www.itl.nist.gov/div898/handbook/prc/section2/prc24.htm

About this report

Published by Rillor on 7 November 2026. Superintelligence (SI) tools assist research and drafting, and every figure cites its source. Rillor builds and secures SI systems, supplies the data they learn from and delivers the infrastructure they run on. We apply SI to financial and compute markets.

Cite as: Rillor. How to test document extraction accuracy before you buy. 7 November 2026. https://rillor.com/insights/testing-document-extraction-accuracy

Notices

Superintelligence (SI) outputs can be wrong. Important decisions should include human review.

Work with Rillor

Send a sample of the rulebooks, regulations, contracts or filings you work from and the fields you need from them. Contact us at rillor.com/contact?type=document-intelligence or contact@rillor.com. More on document and regulation intelligence: rillor.com/si/document-intelligence