Appendix Contents

A Data Collection and Processing 18
A.1 Trial-Court Scraping 18
A.2 The Charlotin Gold Set: Case Linking 18
A.3 Case and Document Inventory 20
A.4 Text Extraction and Preprocessing 21
B Detector Details 22
B.1 Text Processing 22
B.2 Detector Implementation Details 23
B.3 MLE Estimation Objective 24
C Factors Contributing to P/Q Divergence in Legal Writing 25
C.1 Data and Conventions 25
C.2 Is Legal Text Indistinguishable From Its Own AI Rewrites? 25
C.3 Content Anchoring 25
C.4 Boilerplate 27
C.5 Boilerplate Dilutes the Signal 28
C.6 Other Generators and Prompts 29
C.7 Boilerplate and EditLens on the Trial-Court Corpus 31
C.8 Realistic Use: Answers to Questions (Q') 33
C.9 Vocabulary Concentration and Divergence 34
D The Five-Domain Comparison 35
D.1 Human Documents 35
D.2 AI Text 36
D.3 Units 36
D.4 The Columns 37
D.5 What the Table Shows 38
D.6 Caveats 39
E Setting Sensitivity 39
E.1 EditLens 39
E.2 Pangram 40
E.3 MLE 40
E.4 Aggregation Rules on PhantomDocket 41
E.5 Pangram Aggregation Rules and Benchmark Tradeoffs 42
E.6 EditLens sensitivity to window geometry and score aggregation 43
E.7 Does Calibration Remove the Dependence on the Setting? 43
F EM algorithm for ScoreMix 44
G Hyperparameter Tuning 47
G.1 Classification sensitivity across EditLens specifications 47
H Benchmark Score Distributions and ROC Curves 47
I Additional Results on the Trial Courts 49
I.1 Case Mix by County 49
I.2 Representation Status 50
I.3 Labelling Paragraph Function 51

Year	Tulsa	OK	SF	SC
2020	3,921	5,819	1,409	2,995
2021	3,657	5,193	3,678	2,791
2022	3,912	5,807	3,669	3,023
2023	4,541	6,782	2,725	3,486
2024	4,868	7,695	3,267	4,233
2025	5,953	8,729	5,841	4,794
2026	4,291	6,715	4,545	3,179
Total	31,143	46,740	25,134	24,501

Table 4. Case-initiating complaints included in the detector corpus per jurisdiction and year: 31,143 Tulsa, 46,740 Oklahoma (OK), 25,134 San Francisco (SF), and 24,501 Santa Clara (SC) filings, 2020–2026.

A Data Collection and Processing

A.1 Trial-Court Scraping

We scrape case-initiating complaints from the trial courts of Tulsa County, Oklahoma County, San Francisco County, and Santa Clara County through three public digital records systems; Section 2.1 describes the case types in scope for each court.

Both Oklahoma courts are collected from the Oklahoma State Courts Network (OSCN) court-records search (Figure 4). OSCN limits a search response to approximately 500 records. We issue queries in parallel covering every county, filing weekday, and case class. For each query, we set the lower and upper filing dates to the same day and supply a case-type parameter from OSCN. This gives each county-day-class query its own 500-record allowance.

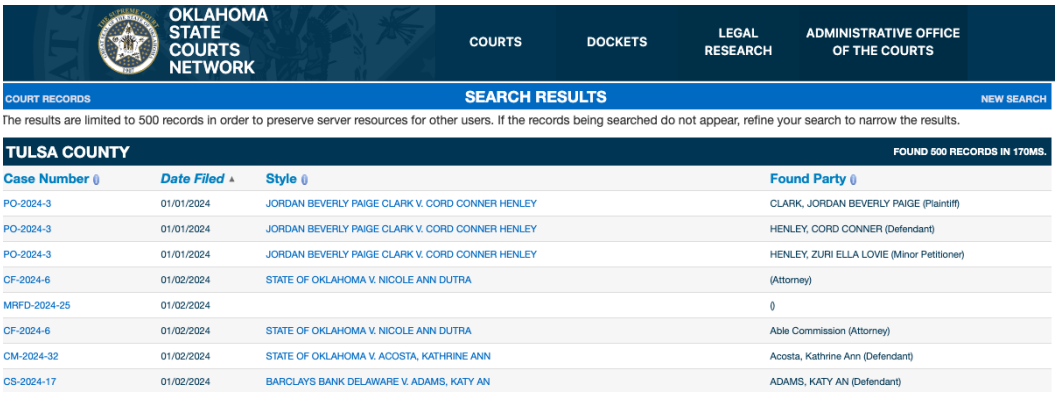
San Francisco complaints are collected from the court's New Filings search. We query one filing date at a time, expand the results table to include all returned cases, and retain cases with the CGC general-civil prefix. For each retained case, we collect the case metadata, parties, attorneys, complete register of actions, and linked docket documents. The typical runs we performed allowed at most two simultaneous case operations and six simultaneous document downloads per worker. We implemented session-level retries and intermediate failure results to accommodate Cloudflare challenges, session expiration, and unsuccessful case or document requests.

Santa Clara complaints are collected through the court's public portal JSON endpoints; these were performed after a browser session was established and the portal's required case token was acquired. The Civil search also returns details not related to CV, so we apply the substantive case-type inclusion and exclusion rules; these are described more in Section 2.1. Within each eligible case, we prioritize the original complaint, exclude amended, cross-, supplemental, and other non-initiating complaint references, require the complaint date to match the searched filing date, and attempt at most one document download per case. After running everything to completion, we re-ran to fille production runs and inserted a uniformly randomized delay of 1.2–2.5 seconds before portal API requests. The collection ran over several months under the system's rate limits.

Table 4 reports the per-year yields that make up the detector corpus, the filings for which an EditLens score exists.

A.2 The Charlotin Gold Set: Case Linking

Charlotin's AI Hallucination Cases database records rulings that identify hallucinated citations, but not the underlying filings. We recover the filings by joining three unique identifiers: the case or docket number, the jurisdiction, and the ECF/docket-entry number of the filing. Figure 5 shows



OKLAHOMA STATE COURTS NETWORK

COURTS | DOCKETS | LEGAL RESEARCH | ADMINISTRATIVE OFFICE OF THE COURTS

COURT RECORDS — SEARCH RESULTS — NEW SEARCH

The results are limited to 500 records in order to preserve server resources for other users. If the records being searched do not appear, refine your search to narrow the results.

TULSA COUNTY — FOUND 500 RECORDS IN 170MS.

Case Number	Date Filed	Style	Found Party
PO-2024-3	01/01/2024	JORDAN BEVERLY PAIGE CLARK V. CORD CONNER HENLEY	CLARK, JORDAN BEVERLY PAIGE (Plaintiff)
PO-2024-3	01/01/2024	JORDAN BEVERLY PAIGE CLARK V. CORD CONNER HENLEY	HENLEY, CORD CONNER (Defendant)
PO-2024-3	01/01/2024	JORDAN BEVERLY PAIGE CLARK V. CORD CONNER HENLEY	HENLEY, ZURI ELLA LOVIE (Minor Petitioner)
CF-2024-6	01/02/2024	STATE OF OKLAHOMA V. NICOLE ANN DUTRA	(Attorney)
MRFD-2024-25	01/02/2024		()
CF-2024-6	01/02/2024	STATE OF OKLAHOMA V. NICOLE ANN DUTRA	Able Commission (Attorney)
CM-2024-32	01/02/2024	STATE OF OKLAHOMA V. ACOSTA, KATHRINE ANN	Acosta, Kathrine Ann (Defendant)
CS-2024-17	01/02/2024	BARCLAYS BANK DELAWARE V. ADAMS, KATY AN	ADAMS, KATY AN (Defendant)

Fig. 4. Oklahoma State Courts Network (OSCN) search results for Tulsa County, the scraping source for Oklahoma case-initiating filings. Each record lists case number, date filed, style, and found party. The interface caps results at 500 records per query ("to preserve server resources"), which dictates the narrowed date-window scraping strategy described in Appendix A.

an example docket view from which target filings are identified. The rest of this section describes the retrieval process, its challenges, and our verification of each document; Section 4.2 gives the summary.

To create PhantomDocket, we first filtered down to deduplicated U.S. cases in the Charlotin Database (1,279 of 1,848 source cases as of 12 October 2026). For each case, we searched for the filing that includes the hallucination using these steps: (1) Extraction of case docket number from the ruling document's caption or document header. (2) Identifying the electronic filing number (ECF) or docket-entry number, filing title, and filing party of the implicated document. (3) Locating the case in Lexis CourtLink or Bloomberg Law and matching the candidate to an exact docket entry using the docket number, court, case caption, filing number, and filing party. (4) Retrieving the filing and verifying its identity using the internal CM/ECF stamp, caption, and title. If available, we also verified that the target filing includes the hallucination quoted in the ruling. This yields 980 hallucination filings drawn from 619 cases.

This process required addressing the challenges of docket resolution, filing identification, and data access. First, Charlotin does not supply a searchable docket number, which is used to identify the specific proceeding in court. This hence required extracting the court, litigants, and the docket number from the ruling documents' captions and headers. The extracted docket numbers may also differ in format, with variation in ease of lookup through PACER, Lexis CourtLink, and/or Bloomberg Law. We excluded any case for which we could not clearly identify and verify the correct docket number, litigant, and jurisdiction; this removed 115 of the 1,279 U.S. cases (9%).

Second, a ruling did not always cleanly or plainly identify the offending filing by its ECF. Judges often refer to just the title of the filing e.g. "the Plaintiff's Motion to Dismiss", cite several filings collectively e.g. "the Plaintiff's briefs", or discuss a filing written by the opposing counsel which uncovered errors in another document. We hence carefully mapped the opinion's discussion to a specific filing in the docket. Typographical errors were also possible, with at least two rulings citing the wrong ECF for an implicated document. This multi-hop identification required access not just to the full ruling, but to the full docket. We hence developed an agentic workflow that could search across a case docket, understand the ruling, and compare candidate entries given the rulings' descriptions of the hallucination. Even then, many opinions did not provide an explicit

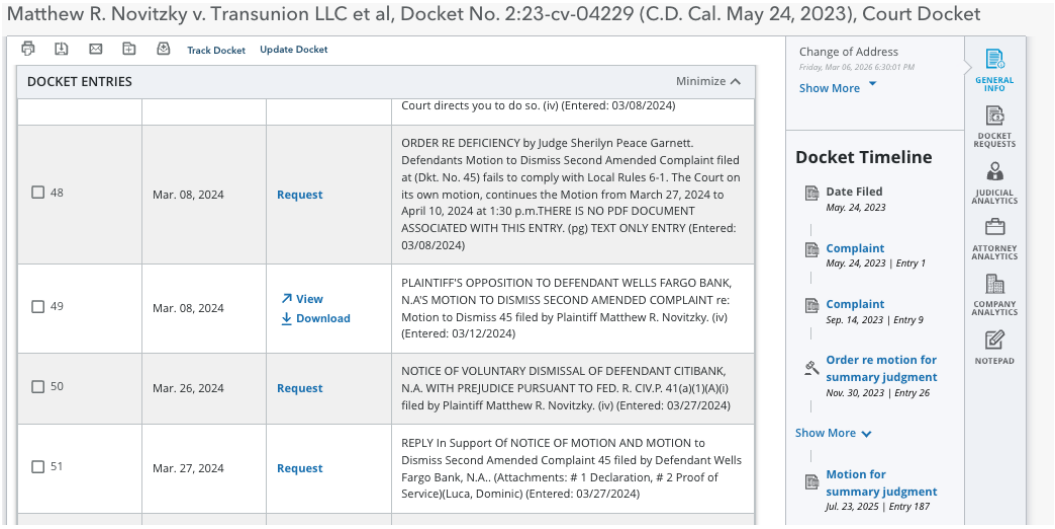


Fig. 5. Example court docket as viewed in Bloomberg Law, which we used to locate target entries (*Matthew R. Novitzky v. Transunion LLC et al.*, No. 2:23-cv-04229 (C.D. Cal.)). The target filing is identified by its ECF entry number and then requested from PACER where a document is attached.

filing number or identify the particular citation or proposition at issue, making targets impossible to recover with sufficient confidence. Of the 928 cases we located in a docket database, 249 had no ECF entry we could tie to the ruling; at the document level, 407 of 1,913 candidate filings (21%) were referenced only by name.

Third, many cases were not freely and publicly available from the Courtlistener RECAP archive [16]. Commercial legal databases may offer these, but terms of service may forbid publically sharing documents sourced from these services. These databases often had incomplete or stale dockets as well, forcing paid update requests to surface the relevant filing. Where Bloomberg Law surfaced the relevant docket entry, we downloaded the document from PACER directly rather than through Bloomberg Law, with five exceptions. The exceptions were obtained through Bloomberg Law; we include them in our analyses but exclude them from the public release of PhantomDocket (https://huggingface.co/datasets/Arifov/PhantomDocket). Numerous filings were also sealed, restricted, or stricken from the record. State-court coverage was particularly affected, as commercial docket coverage was substantially less complete compared to federal courts. In total, 236 cases (359 candidate filings) had dockets unavailable in either Lexis CourtLink or Bloomberg Law. An additional 167 filings could not be retrieved.

After retrieval, we verified the retrieved documents to check for workflow errors. Errors we encountered included: parent filings returned in place of requested attachments, docket entries with similar titles, corrected and amended briefs in lieu of the original offending briefs, and PDFs named under one ECF number but internally stamped with another. We verified each document using its internal CM/ECF stamp, caption, title, filing party, and, where possible, the challenged hallucinated material as described in the ruling.

A.3 Case and Document Inventory

Figure 6 shows how we retrieve documents from Charlotin and Figure 7 how we retrieve court documents. PhantomDocket reflects the Charlotin and court-cases at 12 October 2026 and holds

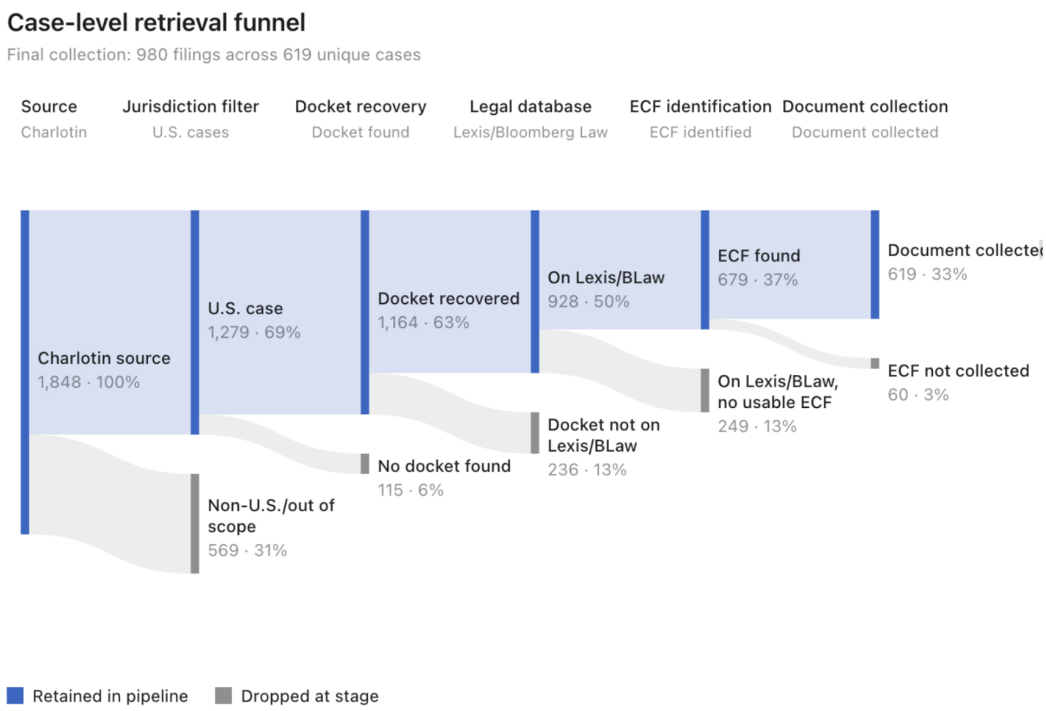


Fig. 6. Case inventory: the case-level funnel from Charlotin source to collected documents. Of 1,848 source cases, 619 (33%) result in a document; the largest areas where we drop cases are non-U.S./out-of-scope cases (569, 31%) and dockets not on Lexis/Bloomberg Law (236, 13%). Blue = retained in pipeline; gray = dropped at stage. Filings for cases located through Bloomberg Law were collected from PACER.

2,289 documents. These include 980 hallucination filings; 538 same-party filings; and 771 controls, 509 opposing-party filings and 262 judicial orders. 20 opposing-party filings were excluded because they reproduce the challenged citations (18) or contain a possible false citation of their own (2).

Common-detector score set. Not every detector scores every document. For example, MLE will drops emply documents (i.e. because, say, deduplication leaves them empty); 2,267 are used for the Oklahoma fit and 2,265 for San Francisco fit. This leaves out 27 hallucination filings. Table 2 and most figures like H are restricted to the 953 hallucination filings, 538 same-party filings and 771 controls (509 opposing-party filings and 262 judicial orders).

A.4 Text Extraction and Preprocessing

We collect the filings as either PDFs or TIFF images which are then converted into text with OCR. PaddleOCR 2.7.3 with PP-OCRv4 was used, with 300 DPI rendering on PyMuPDF. Multipage TIFFs were decoded frame-by-frame with Pillow. Documents had to have at least 200 characters and at least 40% alphabetic characters to be included. We also excluded Judicial Council form complaints (Section 2.1). This text can then be preprocessed depending on the detector setting. We have to segment text for EditLens, but the Pangram API segments it dynamically on its own. The MLE receives more heavily processed text: OCR artifacts are removed, sentences that occur in at least two documents of the reference or target corpus are dropped. Onle then the remainder is cut into

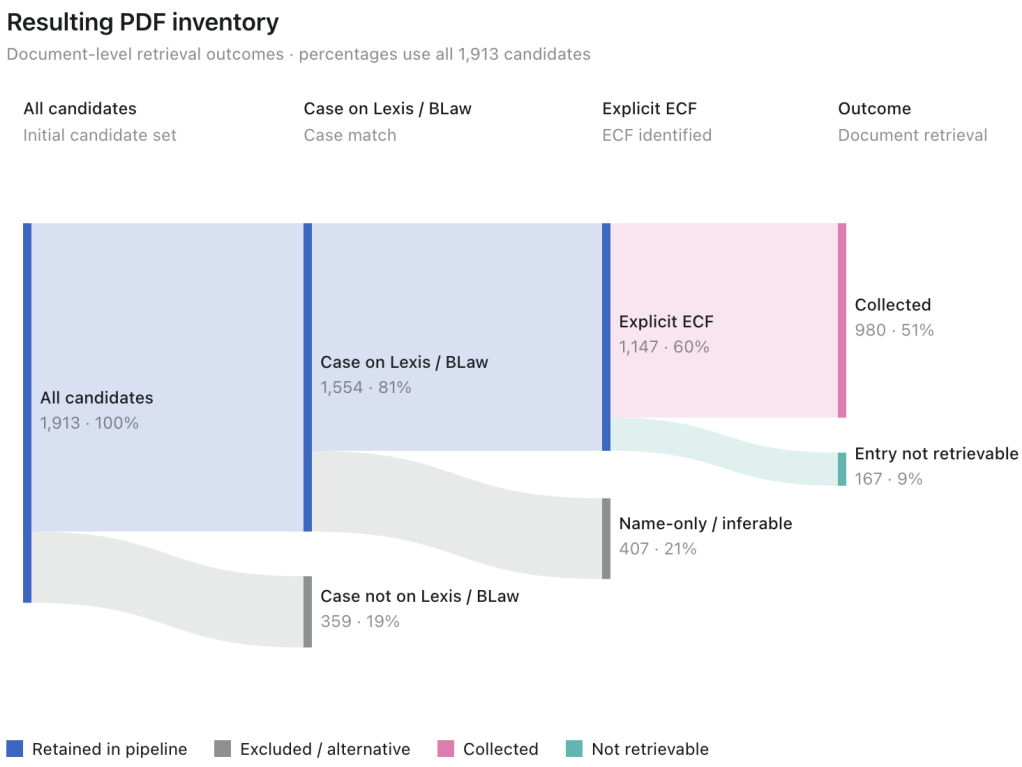


Fig. 7. Resulting PDF inventory: document-level retrieval outcomes. Of 1,913 candidate filings, only 980 (51%) are fully collected by our pipelines; 407 (21%) offer only name-only ECF entries, 359 (19%) belong to cases not on Lexis/Bloomberg Law, and 167 (9%) have explicit ECF entries that are not retrievable. Filings for cases located through Bloomberg Law were collected from PACER

40-token chunks. Appendix B.2 gives each pipeline in full, and Appendix E measures what the MLE's preprocessing changes.

B Detector Details

B.1 Text Processing

We now describe how we processed text before scoring them.

Deduplication. Before estimation, we tokenize and remove OCR artifacts, remove exact duplicate sentences appearing in the reference or target duplicate banks, and divide the remaining text into consecutive 40-token chunks. These preprocessing choices are our attempts to transform legal text into standard, more general-use text before training the MLE estimator. Legal documents repeated text across documents, ranging from forms, standard clauses, captions, and repeat templates. Each 40-token window contributes separately to the likelihood, and double counting sentences each time they appears assigns disproportionate influence to boilerplate. We remove any tokenized sentences that occur *verbatim* in at least two documents before training. Deduplication occurs for both human and synthetic reference texts.

Textual Units. Although researchers often calculate statistics and scored over sentences [25], sentence length varies considerably in our court cases. Because we applied OCR, we have a number of headings, citation fragments, footers, side bars run throughout the document. Some of our corpus documents were also poorly scanned images, misaligned, or generally unparseable. These factors make it difficult to draw sentence-level boundaries. On the synthetic reference side, rewrites have a substantially narrower sentence-length distribution. Word-occurrence likelihood is extremely sensitive to the number of observed samples, and variable-length sentences often make unit length and OCR segmentation *alone* predictive of membership in P or Q. We therefore divide the text into complete, consecutive, nonoverlapping 40-token chunks. We have not measured, but hope that the 40-token specification places units near the scale of an ordinary legal sentences. We treat this as a design choice rather than part of the original MLE method and report both sentence-level and 400-token specifications as robustness checks (Appendix E, Table 18).

B.2 Detector Implementation Details

This appendix gives the implementation details of the three detectors summarised in Section 2.2.

Word-Occurrence MLE. is a distributional method designed to estimate the prevalence of AI-like writing in a corpus from [24, 27]; we refer to it as the MLE. The model assumes text, x, comes from either a human distribution P or an AI distribution Q and the parameter of interest is α_{MLE}, which is the proportion of text coming from the AI distribution. α_{MLE} can estimated by directly maximizing the likelihood,

$$\hat{\alpha}_{\text{MLE}} = \arg \max_{\alpha_{\text{MLE}} \in [0,1]} \prod_i [(1 - \alpha_{\text{MLE}})P(x_i) + \alpha_{\text{MLE}} Q(x_i)]$$

or via the EM algorithm.

Following the original formulation, P is estimated using a randomly selected fitting subset of pre-ChatGPT filings: 2018–2022 for the Oklahoma fit, which pools Oklahoma and Tulsa counties, and 2020–2022 for the San Francisco and Santa Clara fits. The AI distribution Q is estimated using one synthetic rewrite generation for each filing in the P fitting set. We generate these rewrites using GPT-4o-mini in a two-stage procedure that: (1) first converts the source material into notes and (2) then redrafts it, using the notes, to be a matched length. A separate portion of the reference-period filings is held out from fitting for validation and calibration.

To compare document-level classification with Pangram and EditLens, we pool all 40-token chunks in a document and fit one mixture MLE to those chunks, yielding $\hat{\alpha}_{d,\text{MLE}}$. This document-level estimate is substantially noisier than the intended method, which is developed for corpus-level prevalence estimation.

EditLens. is an open-weight transformer-based detector that estimates the degree of generative-AI involvement in a text [48]. Authors train it using human-written texts, AI rewrites of those texts, and versions produced using AI editing instructions of varying intensity[8]. This is, in part, to capture a variety of use-cases and writing styles, as outlined in *Challenge #3* (Section 3). For each text input, EditLens produces four logits, each representing the likelihood of AI involvement being 25%, 50%, 75%, 100% . Our continuous window score is the expected class value normalized to the unit interval, $s = \frac{1}{3} \sum_{k=0}^{3} k\, p_k$. This score measures the model's estimated degree of AI involvement; it is not literally the probability that the text was written by AI. A window's discrete bucket is the class with the largest probability, and can also be used to score a window. The EditLens authors do not

[8]We use the released `pangram/editlens_Llama-3.2-3B` checkpoint, a LoRA adapter for Llama-3.2-3B, without further fine-tuning.

prescribe how to segment documents. In our original setting, we divide text into nonoverlapping 400-word windows, keeping any terminal window that may be shorter. The model's tokenization truncates inputs over 1,024-tokens. The Editlens checkpoint is distributed through a Hugging Face repository under a CC BY-NC-SA 4.0 noncommercial license.

Pangram. is a proprietary commercial detector that has drawn wide attention. Independent evaluations report strong accuracy [20] but also show that it can be evaded [6], and public reporting has questioned its reliability and the harm of its errors [47, 53]; further, it is very expensive, at $0.05 per 100 words for Pangram 4 and $0.05 per 1000 words for Pangram 3.3.2, making a systematic assessment of large corpora infeasible. Pangram has been applied to legal writing in several recent studies: 1,600 federal civil complaints [45], roughly 2,250 federal appellate opinions [33], and 571 articles from the flagship law reviews of the T14 [34]. Each bounds false positives by scoring a pre-ChatGPT period and finding almost nothing—one flagged document in 800 complaints before 2023, no flagged opinions from January 2022, no signal in 2020 law review articles. None, however, measures sensitivity against legal text of known AI authorship, or reports precision and recall on legal writing; the accuracy figures they rely on come from Pangram's model card or from general-domain benchmarks [20]. We use Pangram 3.3.2, a release of Pangram 3.3 [30], which uses architecture based on the EditLens model [48]. The Pangram API returns outputs at both the document and segment levels. At the document level, it returns a categorical prediction (`Human`, `Mixed`, or `AI`), a more detailed headline and prose prediction, estimated fractions of the document categorized as AI-generated, AI-assisted, or human-written, and counts of segments in each category. At the segment level, a confidence category, one of four labels (`human-written`, `lightly AI-assisted`, `moderately AI-assisted`, or `AI-generated`) and a normalized AI-assistance score between zero and one. These vendor-provided document fractions and predictions are distinct from the document-level scores that we construct below. Pangram states that the detector is calibrated to avoid false positives but cautions that templated writing, bullet lists, technical manuals, tables of contents, and artifacts introduced by PDF parsing are especially susceptible to false positives [15, 30]. For long documents, Pangram applies a segmentation procedure that returns variable-length segments. In PhantomDocket, the returned segments contained at most 500 tokens and had a median length of 475 tokens. We submit extracted text after we process using PDFs. We retain legal boilerplate; consequently, PDF-parsing artifacts are avoided, but OCR artifacts and the risks associated with templated legal language remain. For our equal-unit cross-detector comparison, we construct a continuous document score as the unweighted arithmetic mean of the returned segment-level AI-assistance scores. Maximum segment score and word-weighted means are alternative aggregation rules. We also track Pangram's native document prediction and estimated fractions as alternative scores.

B.3 MLE Estimation Objective

Unlike EditLens and Pangram, which detect AI usage on a document level, MLE detects AI prevalence on a population level. Its human reference P is the distribution of word occurrences in pre-ChatGPT court filings, its AI reference Q is the distribution in AI rewrites of those same filings, and the observed corpus, whether the Charlotin filings or one trial-court year, is modelled as $(1 - \alpha_{\mathrm{MLE}})P + \alpha_{\mathrm{MLE}}Q$. The estimate $\hat{\alpha}_{\mathrm{MLE}}$ is therefore the mixture weight that best explains the corpus under P and Q; it is not the share of documents written by AI. Appendix C.8 shows how far this quantity can fall below the true share when the AI text in the corpus does not resemble the AI rewrites from which Q was built.

C Factors Contributing to P/Q Divergence in Legal Writing

In Section 3, we explored theoretical limits to the accuracy of AI-generated text detection in legal documents. We explored how human and AI text distributions, P and Q, are affected by boilerplate, vocabulary range, and patterns of AI usage that occurs in legal writing.

In order to estimate AI adoption at scale, we compare legal filings with four other domains in which AI-detection methods have been tested: consumer complaints submitted to the CFPB, corporate press releases, United Nations press releases, and scientific abstracts [26]. We also examine AI text distributions that result from responses to realistic questions as opposed to being generated as a rewrite of an existing document (Appendix C.8). Finally, Appendix **??** provides definitions of the terms used throughout this analysis.

C.1 Data and Conventions

- **Legal.** In this section, we use 9,153 petitions filed in Tulsa and Oklahoma counties between 2018 and 2022. For each petition, we generate one AI-generated rewrite, as described in the main body.
- **CFPB complaints.** This dataset consists of 14,576 complaints from the 2018–2022 period. As with legal petitions, every complaint is paired with one AI rewrite, generated by [25].
- **Press releases.** We include 4,236 PR Newswire releases published in 2021, together with the AI rewrites provided by Liang et al. [26]. These were generated using GPT-3.5-turbo in two stages: First, the model compressed each release into its key points. It then expanded upon those points in order to produce a completed press release.

For the analyses in this section, we convert the text in all three corpora into lowercase word tokens. Each document is then divided into consecutive 40-token chunks. Then, for each chunk, we record whether a word appears at least once. In order to estimate the human and AI word-occurrence probabilities, P and Q, we randomly select half of the paired documents for fitting. We only incorporate words that occur in at least five human chunks and three AI chunks. We compute AUROC using our fitted model's log-likelihood ratio, as in [25]. We repeat this procedure over five random splits; all results we report are the average over these splits.

We measure detection performance at both the chunk and document levels. For chunk AUROC, every 40-token chunk is treated as a separate observation. For document AUROC, we first average the scores of the chunks that belong to each document. We refer to this fitted detector as the *oracle* detector, as it is trained using rewrites from the same generator on which it is evaluated.

C.2 Is Legal Text Indistinguishable From Its Own AI Rewrites?

Because legal text can use a highly specialized vocabulary, we first hypothesized that AI-written legal text would be unusually similar to human-written text: the register is formal and formulaic in both cases. We test this directly, using pairs of human documents and AI rewrites. If our hypothesis were correct, the oracle detector would have a higher error rate distinguishing rewritten petitions from real ones, compared to rewrites in the other domains. However, we find no difference between the corpora. Legal corpora, we find, are highly separable: AUROC is 0.986 for the Oklahoma petitions, 0.981 for CFPB complaints and 0.976 for press releases, (Table 5).

C.3 Content Anchoring

For each word w, we measure the extent to which its presence in an AI-generated rewrite depends on whether it appears in the original source document. We calculate this using the following expression:

$$\text{anchor}(w) = \Pr(w \in Q_d \mid w \in P_d) - \Pr(w \in Q_d \mid w \notin P_d).$$

Domain	Pairs	Vocab.	KL($Q\|\|P$)	Noise KL	Chunk AUROC [95% CI]	SD over splits	Doc. AUROC	TV ≥
Legal (Oklahoma petitions)	9,153	8,701	10.49 ± 0.13	0.84 ± 0.05	0.987 [0.986, 0.988]	0.001	1.000	0.83
Legal (Oklahoma petitions) (matched n)	4,236	6,042	9.99 ± 0.05	1.27 ± 0.13	0.983 [0.981, 0.984]	0.001	1.000	0.81
CFPB complaints	14,576	4,742	9.80 ± 0.02	0.34 ± 0.03	0.981 [0.980, 0.982]	0.000	0.997	0.81
CFPB complaints (matched n)	4,236	2,768	8.79 ± 0.20	0.63 ± 0.03	0.977 [0.975, 0.979]	0.001	0.997	0.78
Press releases (PR Newswire)	4,236	10,044	9.94 ± 0.05	0.85 ± 0.03	0.976 [0.973, 0.978]	0.001	0.996	0.78

Table 5. Divergence between human documents and their AI rewrites, by domain. Pairs: human documents with one AI rewrite each. KL($Q\|\|P$) is per 40-token chunk under the independent-word occurrence model; noise KL is the same quantity between two random halves of the human documents. Chunk AUROC is calculated on one split, with a document-cluster bootstrap 95% interval. TV ≥ inverts AUROC $\leq \frac{1}{2} + \mathrm{TV} - \mathrm{TV}^2/2$ at the mean chunk AUROC, meaning that rewrites of legal filings are at least as separable from the filings as rewrites of general-domain text are from their sources.

Here, d indexes the paired documents in the fitting half of the data. An anchor is a word that is required by the facts of a document, such as a lender's name or *foreclosure*; if it appears in the original, it is also likely to appear in the rewrite. In the case that the source document does not mention it, however, the word is unlikely to appear. In contrast, words associated with a particular writing style may appear in the rewrite regardless of whether they were present in the source. Consequently, their anchoring scores are closer to 0.

Length is an important confounder: longer petitions, for example, tend to produce longer rewrites. Since longer documents provide more opportunities for words to appear, the difference between the two probabilities used to calculate anchoring may be inflated. As a result, a word may receive a higher anchoring score simply because of document length, as opposed to actual dependence on the source material. In order to account for this, we calculate anchoring separately within five groups based on quintiles of source-document length. We then average the results across these groups, weighting each by the number of source documents that contain the word. We only calculate anchoring for words that appear in at least 20 source documents.

Across all three domains, we find a negative correlation between anchoring and divergence: $\rho = -0.43$ for legal text, -0.42 for complaints, and -0.30 for press releases (Table 6, Figure 8). This suggests that anchored words, whose presence is determined by the underlying facts, contribute very little to the divergence between P and Q. For legal text, words with anchoring scores of at least 0.6 account for 22% of human tokens, versus only 4% of KL($Q\|\|P$). The corresponding figures are 18% and 6% for complaints, and 10% and 1% for press releases.

This pattern is also present in the vocabulary used by petitions. In particular, the occurrence rates of topic-related words remain relatively similar. For example, *mortgage*, with an anchoring score of 0.97, appears in 11.2% of human chunks and 12.6% of AI chunks. Likewise, *county* (0.86) appears in 12.3% and 12.4%, *contract* (0.86) in 5.7% and 5.9%, *judgment* (0.76) in 6.3% and 6.2%, and *attorney* (0.82) in 5.8% and 5.9%, respectively.

On the other hand, words that contribute more substantially to the divergence tend to have low anchoring scores. The occurrence of *seeks* (anchoring 0.07) increases from 0.3% of human chunks to 9.5% of AI chunks. For *furthermore* (0.06), the increase is from 0.3% to 8.5%, while *additionally* (0.06) rises from 0.3% to 6.3%. The opposite pattern exists for *said* (0.09), which decreases from 13.0% of human chunks to 1.0% of AI chunks. Overall, legal text has the highest share of tokens belonging to highly anchored vocabulary, at 22%, compared with 18% for complaints and 10% for press releases. A detector that uses only these highly anchored words achieves an AUROC of just 0.70.

We also examine domain-specific vocabulary; in particular, we analyze words that are at least 4×, 8×, or 16× more frequent in human text. For instance, legal register terms such as *herein*, *wherefore*, and *prays* are grouped together with topic-specific vocabulary. Similarly, press-release words such

		Share of human tokens, by anchoring				Share of KL($Q\|\|P$), by anchoring				
Domain	ρ(anchor, KL)	<0.1	0.1–0.3	0.3–0.6	≥0.6	<0.1	0.1–0.3	0.3–0.6	≥0.6	AUROC, ≥0.6 only
Legal (Oklahoma petitions)	-0.43	34%	21%	19%	22%	27%	31%	23%	4%	0.700
CFPB complaints	-0.42	23%	39%	17%	18%	15%	39%	26%	6%	0.734
Press releases (PR Newswire)	-0.30	15%	30%	35%	10%	22%	33%	19%	1%	0.619

Table 6. Content anchoring by domain (length-stratified; means over three splits). Anchoring of a word is $\Pr(w \in Q_d \mid w \in P_d) - \Pr(w \in Q_d \mid w \notin P_d)$ across paired documents. ρ is the Spearman correlation between a word's anchoring and its contribution to KL($Q\|\|P$). The last column is the held-out chunk AUROC of a detector that uses only words with anchoring of at least 0.6. Proportions do not sum to 100% because words that are in <20 source documents are not scored.

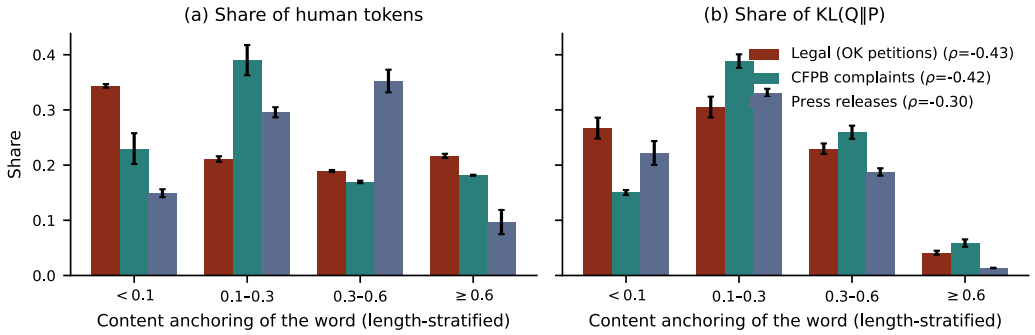


Fig. 8. (a) Share of human tokens in Q at each anchoring level. (b) Share of KL($Q\|\|P$) carried by those words. Error bars are SD over three cross-validation splits; ρ, in the legend, is the Spearman correlation between anchoring and KL contribution.

	KL share, register-specific words			Chunk AUROC (8×)	
Domain	4×	8×	16×	Register words only	Shared words only
Legal (Oklahoma petitions)	16.2%	11.7%	8.1%	0.833	0.984
CFPB complaints	22.7%	13.7%	6.3%	0.842	0.979
Press releases (PR Newswire)	38.1%	32.5%	28.7%	0.896	0.970

Table 7. Dataset split by domain specificity. Register-specific words are at least k× more frequent in the domain's human text than in either other domain's human text.

as *commitment*, *pivotal*, and *crucial*, which can also signal AI-generated text, are included within the same group. We report these results in Table 7.

C.4 Boilerplate

We classify an 8-gram as "boilerplate" in the case that it appears in at least 1% of the human documents within a particular domain. Any token that falls within one of these 8-grams is also considered boilerplate. Further, we examine how the results change when the length of the n-gram is varied. Specifically, we consider lengths of 6, 8, and 10 tokens, alongside document-frequency thresholds of 0.5%, 1%, and 2%.

The results show that boilerplate is much more prevalent in legal text than in the other two domains. For legal petitions, 21.1% of all tokens are classified as boilerplate language. Across the

Domain	Boilerplate token share		Rewrite suppression	Docs ≥25%	KL share, bound words	AUROC drop, masked	ρ(share, LR score)
	Human (range)	AI rewrite					
Legal (Oklahoma petitions)	21.1% (12.2%–32.4%)	1.2%	98.0%	35.0%	5.4%	0.0019	-0.49
CFPB complaints	0.2% (0.0%–1.2%)	0.0%	–	1.0%	0.0%	0.0000	0.06
Press releases (PR Newswire)	3.6% (1.0%–5.8%)	0.0%	99.9%	4.0%	0.8%	0.0007	-0.20

Table 8. Amount of boilerplate by domain, where boilerplate is defined as an 8-gram present in at least 1% of a domain's human documents. Rewrite suppression is calculated as one minus the ratio of boilerplate 8-gram rates. Bound words occur at least half the time inside boilerplate. The drop in AUROC is the change in held-out chunk AUROC when boilerplate spans are deleted from human documents. The last column is the Spearman correlation, among human documents, between boilerplate share and the likelihood-ratio score (↑ means more AI-like).

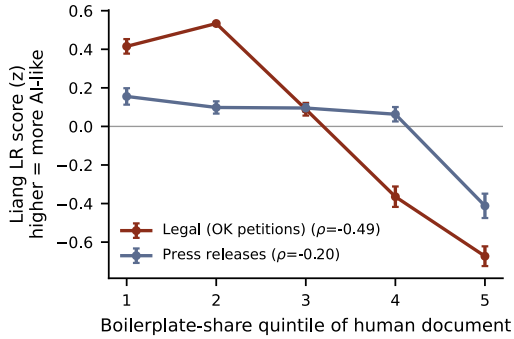


Fig. 9. Likelihood-ratio score of held-out human documents by boilerplate quintile (standardized; higher is more AI-like; error bars are SD over five splits). For the occurrence detector trained on rewrites, templated human filings look more human, not less.

nine combinations of n-gram lengths and frequency thresholds, this share ranges from 12.2% to 32.4%. By comparison, boilerplate accounts for only 3.6% of tokens in press releases and 0.2% in complaints (Table 8). However, the amount of boilerplate is also quite volatile across individual legal filings, with some containing substantially more than others. Although the median petition contains only 10.5% boilerplate, 35% of petitions have at least one-quarter of their tokens classified as boilerplate. At the 90th percentile, the share reaches 93%. Interestingly, the AI-generated rewrites retain very little of this repeated material. Boilerplate 8-grams occur 98.0% less frequently in the petition rewrites than in their corresponding human originals. For press releases, this reduction is even greater, at 99.9%.

Despite this difference, the word-occurrence detector does not rely on the absence of boilerplate language in the rewrites to distinguish AI from human text. When we remove boilerplate spans from held-out human petitions, chunk AUROC decreases by only 0.002. Furthermore, words that occur primarily within boilerplate account for just 5.4% of legal KL divergence. We also find that petitions containing more boilerplate tend to receive scores indicating that they are more likely to be human-written. In particular, the rank correlation between a petition's boilerplate share and its likelihood-ratio score is −0.49. The corresponding correlation for press releases is −0.20 (Figure 9).

C.5 Boilerplate Dilutes the Signal

In this section, we will show a more formal proof for how boilerplate, here defined as n-grams that repeat across documents, leads to lower AI-detection rates. Intuitively, if a large portion of text is

Domain	L	Detector	AUROC, $b = 0$	$\hat{\gamma}$	RMSE of predicted AUROC		
					$\hat{\gamma}$	$\gamma = 1$	$\gamma = \frac{1}{2}$
Legal (Oklahoma petitions)	40	boilerplate-masked fit	0.9838	0.90	0.0028	0.0179	0.0862
Legal (Oklahoma petitions)	40	natural fit	0.9861	0.98	0.0018	0.0039	0.1006
Legal (Oklahoma petitions)	120	boilerplate-masked fit	0.9989		saturated at $b = 0$		
Legal (Oklahoma petitions)	120	natural fit	0.9994		saturated at $b = 0$		
Press releases (PR Newswire)	40	boilerplate-masked fit	0.9744	1.05	0.0042	0.0083	0.1074
Press releases (PR Newswire)	40	natural fit	0.9744	0.99	0.0038	0.0042	0.0984
Press releases (PR Newswire)	120	boilerplate-masked fit	0.9922	0.89	0.0133	0.0286	0.0846
Press releases (PR Newswire)	120	natural fit	0.9926	0.86	0.0154	0.0339	0.0808

Table 9. The effects of templates in legal writing. $\hat{\gamma}$ is the least-squares exponent in $\Phi^{-1}(\text{AUROC}_b) = (1 - b)^{\gamma}\,\Phi^{-1}(\text{AUROC}_0)$ over $b \in \{0.1, \ldots, 0.9\}$. RMSE columns compare predicted and observed AUROC under $\hat{\gamma}$, $\gamma = 1$, and $\gamma = \frac{1}{2}$. The natural-fit detector is trained on unmodified human documents; the masked-fit detector is trained on human documents with boilerplate deleted.

repeated between documents, this "crowds out" text that can be used to distinguish text derived from an AI distribution, Q; and a human distribution, P.

Proposition. Let B be a distribution over text *shared* by human and AI authors, and suppose each token of a document comes from B with probability b; otherwise, it comes from either P (human) or Q (AI). Define $P_b = bB + (1 - b)P$ and $Q_b = bB + (1 - b)Q$. Under these mixtures,

$$\text{TV}(P_b, Q_b) = (1 - b)\,\text{TV}(P, Q) \quad \text{and} \quad D_f(Q_b \| P_b) \le (1 - b)\,D_f(Q \| P)$$

for any f-divergence D_f, including KL and χ^2. Substituting this into the detection bound gives $\text{AUROC} \le \frac{1}{2} + (1 - b)\,\text{TV} - (1 - b)^2\,\text{TV}^2/2$.

Proof. Subtracting the two mixtures gives $P_b - Q_b = (1 - b)(P - Q)$, establishing the first identity. For the second, we use the joint convexity of D_f to (B, B) and (Q, P), with weights b and $1 - b$. Since $D_f(B \| B) = 0$, the stated inequality follows. □

We test this relationship directly by constructing windows with known amounts of boilerplate text. From held-out documents we build windows of L tokens in which a fraction b is human boilerplate and the rest is either non-boilerplate human text or AI rewrite text from the same document. We score windows with the fitted detector and compute AUROC for b from 0 to 0.9 over three splits. For $L = 40$, we find that the relationship between boilerplate share and AUROC is described by

$$\Phi^{-1}(\text{AUROC}_b) = (1 - b)^{\gamma}\,\Phi^{-1}(\text{AUROC}_0).$$

The fitted exponent is $\hat{\gamma} = 0.98$ for legal text and 0.99 for press releases. The corresponding RMSE values are 0.002 and 0.004; this is in comparison with approximately 0.10 when $\gamma = \frac{1}{2}$ (Table 9, Figure 10). $\gamma = 1$ indicates that boilerplate contributes variance to the detector's scores at the same rate as authored content does. This means that separability decreases in proportion to the share of the window that is not boilerplate. In the case of legal text, AUROC falls from 0.986 when $b = 0$ to 0.870 when $b = 0.5$, and to 0.674 when $b = 0.8$. When $b = 0$, AUROC is already close to 1, though the fitted exponent for press releases ranges from 0.86 to 0.89.

C.6 Other Generators and Prompts

In our setup, rewrites of human-generated text to produce AI counterfactuals, Q entails a two-stage drafting process, where one model serves as a *drafting coordinator*. We construct variations to allow for more diversity: the drafting model may choose itself or a different model to generate the text,

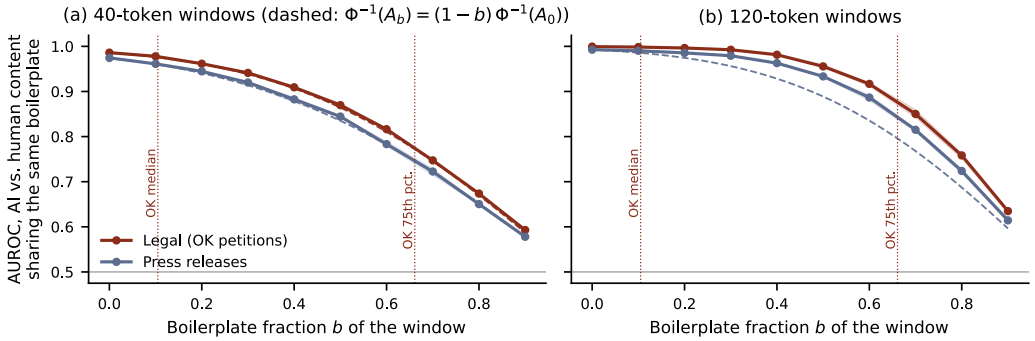


Fig. 10. Detection rates fall in proportion to the boilerplate fraction of the window. (a) 40-token windows, fitted with $\Phi^{-1}(\mathrm{AUROC}_b) = (1-b)\,\Phi^{-1}(\mathrm{AUROC}_0)$ dashed. (b) 120-token windows; the dashed line is the same law for press releases (legal AUROC is saturated at $b = 0$).

infill the conventions of the genre, or prompt the generator asking for text that sounds human. We generated new AI text for 1,500 documents per domain, drawn from the held-out half of the split behind Table 5.

Every source first goes through the same step: Llama-3.1-8B-Instruct lists its facts as bullet points, then, we experiment with the following variations:

- **Default draft.** A model writes the petition or press release from the bullet points: Llama-3.1-8B-Instruct, and for legal text also Mistral-7B-Instruct-v0.3 and SaulLM-7B-Instruct, a Mistral-based model with continued pretraining on legal text.
- **Register prompt.** We follow the default process, but the prompt asks for the draft "as an experienced Oklahoma litigation attorney" would write it, with the genre's formulas ("COMES NOW the Plaintiff," "alleges and states as follows," "WHEREFORE"); for press releases, the PR Newswire dateline, executive quotes, and an "About" paragraph.
- **Humanizing rewrite.** Llama-3.1-8B-Instruct rewrites its own default draft "as if it were written by an experienced human" professional, avoiding a list of stock AI words.

We score each arm against 500 held-out human documents, with two detectors. The *deployed* detector is shown in Table 5, fit on the original rewrites (GPT-4o-mini for legal text, GPT-3.5-turbo for press releases). The *matched* detector is fit on the other 1,000 documents.

On the original rewrites of the test documents, the detector reaches document AUROC 1.000 for legal text and 0.997 for press releases; on Llama default drafts of the same documents, it reaches 0.928 and 0.931. Part of the drop in performance is due to how the sources were prepared: the original legal rewrites were written from bullet points, whereas ours kept party names. Llama register-prompted petitions reach 0.772 AUC, against 0.834 for register-prompted press releases. The drop from default to register draft is 0.156 (95% CI 0.135–0.176) in legal text and 0.097 (0.080–0.113) in press releases, a difference of 0.059 (0.034–0.085). This drop is similar for Mistral (0.140) and SaulLM (0.112; or 0.131 excluding drafts that hit the length limit). A humanizing rewrite, we find, does not help a draft evade this detector. It raises document AUROC to 0.981 for petitions and 0.963 for press releases. The detector, fit on one open model's default drafts, is able to distinguish every generation variation from human text at the document level (AUROC 0.986 in legal text; 0.989 in press releases).

Prompts specifically requesting different registers, we find, are able to induce the models to reproduce more of the genre's boilerplate: register-prompted Llama rewrites contain 20.7 human

		Deployed detector		Matched	Refit	AI tells	Boilerplate
Domain	Generator, arm	Doc. AUROC [95% CI]	Chunk	doc. AUROC	chunk AUROC	per 1k	8-grams per 1k
Legal	human documents	–	–	–	–	0.62	162.9
	GPT-4o-mini, original rewrite	1.000 [1.000, 1.000]	0.987	–	–	–	–
	Llama-3.1-8B, default draft	0.928 [0.913, 0.941]	0.739	1.000	0.906	1.89	13.1
	Llama-3.1-8B, humanizing rewrite	0.981 [0.974, 0.986]	0.810	0.999	0.915	2.94	11.0
	Llama-3.1-8B, register prompt	0.772 [0.746, 0.796]	0.656	0.997	0.880	1.63	20.7
	Mistral-7B, default draft	0.970 [0.962, 0.978]	0.810	0.999	0.910	2.33	7.9
	Mistral-7B, register prompt	0.831 [0.811, 0.851]	0.667	0.997	0.889	1.59	13.3
	SaulLM-7B, default draft	0.896 [0.877, 0.914]	0.722	0.991	0.867	2.16	8.6
	SaulLM-7B, register prompt	0.784 [0.758, 0.807]	0.647	0.986	0.850	1.54	9.7
Press releases	human documents	–	–	–	–	3.15	19.4
	GPT-3.5-turbo, original rewrite	0.997 [0.993, 1.000]	0.977	–	–	–	–
	Llama-3.1-8B, default draft	0.931 [0.915, 0.943]	0.756	0.992	0.872	10.72	1.3
	Llama-3.1-8B, humanizing rewrite	0.963 [0.951, 0.971]	0.787	0.993	0.882	9.09	1.0
	Llama-3.1-8B, register prompt	0.834 [0.811, 0.850]	0.667	0.989	0.867	9.15	1.9

Table 10. The effects of using other generators and prompts. Deployed detector: is trained on the original rewrites (Table 5). Matched: is trained on 1,000 other documents against Llama-3.1-8B default drafts. Refit: two-fold cross-fit of human text against that arm on the 500 test documents. Document AUROC compares each arm with the same 500 held-out human documents; intervals are document-bootstrap 95% CIs.

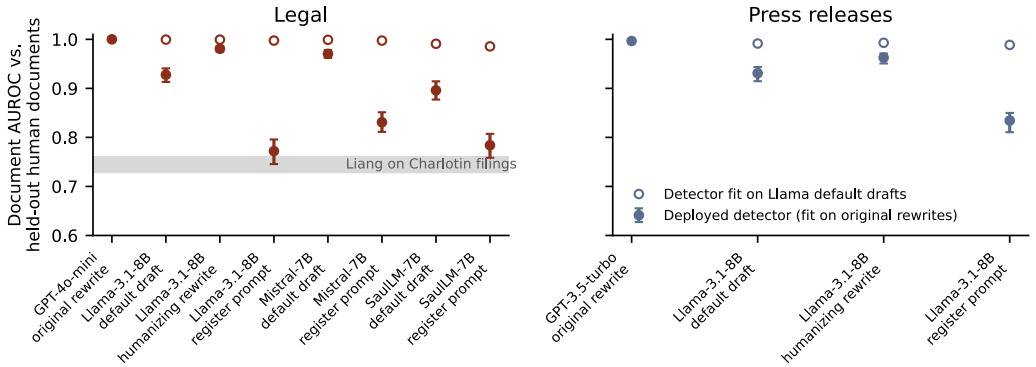


Fig. 11. Document AUROC of AI drafts compared with held-out human documents, grouped by generator and prompt. Solid points represent the deployed detector, fit on the original rewrites, with 95% CIs. Hollow points represent a detector fit on Llama-3.1-8B default drafts. The shaded band is the Liang detector's document AUROC on Charlotin hallucination filings. A detector fit on one generator's default register loses performance when the generator or the prompt changes.

boilerplate 8-grams per 1,000 positions vs. 13.1 for Llama default drafts and 162.9 for human-written petitions. Legal drafts, we find, repeat themselves more than press-release drafts (a 12-gram repeated at least three times appears in 4–24% of legal drafts and 2–4% of press-release drafts).

C.7 Boilerplate and EditLens on the Trial-Court Corpus

Our results explored in Appendix C.4 are specific to the word-occurrence detector, in particular, its fitted distribution Q is dependent on AI rewrites that omit 98% of boilerplate 8-grams. This does not prove a different detector would function in the same way. We repeat the analysis using EditLens across the full trial-court corpus. Unlike the word-occurrence detector, EditLens is based neither on word occurrence nor perplexity. Instead of computing a likelihood ratio, it estimates the magnitude of AI editing.

We use the same definition of boilerplate that was applied in the preceding analysis. Specifically, for each jurisdiction, we randomly select 5,000 documents and identify all 8-grams that appear in at least 1% of that sample. We then examine every 400-word window within the jurisdiction. For each window, we compute the fraction of tokens that fall within one of the identified boilerplate 8-grams.

No false-positive effect. Across a total of 555,905 scored windows, we analyze the effect of boilerplate language on EditLens scores. In Oklahoma, the correlation is +0.071 ($n = 190,010$), whereas in San Francisco, it is −0.074 ($n = 365,895$). In San Francisco, the correlation remains negative across each year, ranging from −0.02 to −0.19. Yet, in Oklahoma, a change appears around 2022. Specifically, the correlation rises to +0.186 in 2022 from −0.056 in 2020. Therefore, we find no evidence that EditLens systematically classifies templated human writing as being AI-generated. This concern was previously identified for distribution-based detectors [25].

The score variance compresses for AI-written text. Examining the average score alone does not suffice for dilution prediction. When a fraction b of a window is comprised of boilerplate or anchor content, the evidence from its authored portion becomes weaker. Since AI-assisted windows make up only a small minority, the overall mean score changes minimally. However, when moving from windows with no boilerplate to those containing 25–50% boilerplate, we observe a substantial reduction in variance of EditLens scores. In Oklahoma, the standard deviation decreases from 0.037 to 0.019; in San Francisco it falls from 0.057 to 0.027. Likewise, the 99th percentile decreases from 0.179 to 0.092 in Oklahoma, and from 0.295 to 0.130 in San Francisco. The maximum observed score also declines from 0.88 to 0.40 in Oklahoma and from 0.99 to 0.47 in San Francisco, respectively. Over this same range, the proportion of windows scoring above 0.3 falls by factors of 14 in Oklahoma and 16 in San Francisco. This trend reverses when boilerplate exceeds 75%. In Oklahoma, this category has the greatest score variance of any boilerplate group, with a standard deviation of 0.052. Additionally, 7.9% of its windows score above 0.1, while only 4.7% score near zero, versus 11–18% in the other groups. This suggests that the highest-boilerplate windows may represent a separate population of nearly duplicate form documents.

Why this does not identify performance drops. Although the compression in scores is consistent with dilution, it could also be attributed to differences in the composition of the text. In particular, windows containing more boilerplate language may simply contain less AI-assisted writing. This is plausible because certain materials, such as an attached promissory note, are generally not rewritten by the filer. Our analysis, however, cannot distinguish between these two explanations. One reason for this limitation is that EditLens provides a one-sided measure of AI editing. While the model reports the magnitude of editing, it provides no distinction between "no evidence of editing" and "confidently unedited" text. Both explanations would therefore produce similar patterns in the scores. Specifically, each predicts a reduction in the upper tail, a decrease in dispersion, as well as a greater concentration of scores near zero. We observe this last pattern in both jurisdictions: In Oklahoma, the share of windows scoring below 0.005 increases from 11.5% to 18.1%; in San Francisco, the corresponding share increases from 11.5% to 17.7%.

Caveats to our Boilerplate Detection Method: Unique text is not non-boiplerplate text. Now, we examine the windows with the lowest EditLens scores. Among windows in the lowest 2% of EditLens scores, 43.5% contain no boilerplate in Oklahoma, versus 41.8% of randomly selected windows. In San Francisco, the corresponding figures are 42.5% and 41.6%, respectively. Upon examining these windows more closely, we find that much of their content consists of reproduced third-party material. The following are examples of such content: a shopping-center lease exclusivity rider; an insurer's underwriting inspection manual; credit-card payment terms; a homeowners' association

		λ, share registered		Caught at Q-calibrated cut		Document AUROC	
Model	AI text	est.	95% CI	as pasted	length-matched	deployed	matched
Llama-3.1-8B	Q, Liang et al. prompts	0.65	[0.55, 0.76]	90%	90%	0.955	0.952
	Q′, attorney, "draft this section"	0.33	[0.24, 0.42]	62%	70%	0.816	0.887
	Q′, pro se, "help me write this"	0.35	[0.26, 0.44]	61%	55%	0.799	0.937
	Q′, pro se, "what do I need to say?"	0.27	[0.15, 0.38]	52%	53%	0.776	0.998
Mistral-7B	Q, Liang et al. prompts	0.85	[0.73, 0.95]	90%	90%	0.983	0.972
	Q′, attorney, "draft this section"	0.51	[0.40, 0.63]	54%	72%	0.895	0.918
	Q′, pro se, "help me write this"	0.46	[0.37, 0.56]	45%	59%	0.880	0.973
	Q′, pro se, "what do I need to say?"	0.40	[0.31, 0.50]	38%	64%	0.873	0.998
Qwen3-8B	Q, Liang et al. prompts	0.95	[0.86, 1.03]	90%	90%	0.997	0.992
	Q′, attorney, "draft this section"	0.48	[0.37, 0.58]	29%	51%	0.921	0.926
	Q′, pro se, "help me write this"	0.36	[0.28, 0.46]	24%	37%	0.843	0.967
	Q′, pro se, "what do I need to say?"	0.09	[0.03, 0.16]	1%	16%	0.574	0.999

Table 11. 378 passages from 115 held-out petitions are scored on three open models. Q is generated by [25]-style compress then rewrite protocol. Q' is generated by the model's answer to a realistic question. λ is share of a true 10% AI fraction the deployed estimator reports. Caught is the share of documents above the threshold that flags 90% of the same model's rewrites, as pasted and with each generation cut to its passage's length.

declaration amendment; a laboratory price schedule; a scaffolding contractor's terms; and an owner-controlled insurance program manual. Although these materials are unique within our corpus, none of them was composed by a party to the case. Consequently, their measured boilerplate share is zero. This highlights an important limitation of our definition of boilerplate. The n-gram criterion only identifies text that appears repeatedly *across* documents in the sampled corpus. Material that appears in only one document (i.e., an attached agreement) is therefore excluded from this measure. However, this does not mean that it does not contain boilerplate. As a result, the forced-content shares reported in Table 14 are lower bounds.

C.8 Realistic Use: Answers to Questions (Q')

The MLE estimator in [25] constructs its AI reference Q from rewrites of human documents. Actual users may, instead, ask a chatbot a legal question and paste part of its answer. These two generation techniques may have very different distributions. To test this, we extracted 378 passages from 115 held-out petitions covering factual allegations, causes of action, party descriptions, citations, prayers for relief, and jurisdiction or venue. For each passage, agents wrote three questions that a pro se filer or attorney might ask to obtain comparable text. We then used Llama-3.1-8B-Instruct, Mistral-7B-Instruct-v0.3, and Qwen3-8B to produce an answer to each question. We evaluate in three ways. First, λ measures how much of a AI share the deployed rewrite-based estimator recovers. Second, we apply the threshold that detects 90% of each generator's rewrites. Third, we refit the word-occurrence detector on the answers themselves, to determine whether they are intrinsically difficult to separate from human text.

When the generate text consists of rewrites, the estimator registers 65–95% of the AI share, but only 9–51% with questions (Table 11). Similarly, a threshold that detects 90% of rewrites detects only 16–72% of answers, even with matched length. Refitting a detector on answers specifically improves performance to AUROC 0.89–1.00. Therefore, a detector calibrated on just rewrites often will look for the wrong signal out of its distribution.

AI text (Oklahoma petitions)	AI units	Chunk AUROC	TV ≥
GPT-4o-mini rewrites, oracle fit on 4,576 pairs	64512	0.987	0.84
GPT-4o-mini rewrites, cross-fit on 500 pairs	7019	0.965	0.74
Llama draft from notes, default prompt	5158	0.898	0.55
Llama draft from notes, register prompt	4600	0.874	0.50
Llama rewrite of a passage, Liang et al. prompts	4456	0.855	0.46
Llama answer, attorney "draft this section"	2659	0.824	0.41
Llama answer, pro se "help me write this"	2293	0.878	0.51
Llama answer, pro se "what do I need to say?"	2970	0.971	0.76

Table 12. Separability of legal AI text by production process. AUROC of the best word-occurrence detector for each text: the oracle fit for GPT-4o-mini rewrites, and a two-fold cross-fit on the arm's own text. TV ≥ inverts the bound of Sadasivan et al. [42].

	$H(P)$	Effective	Types	Human/AI unigram TV				TV from rare	Chunk
Domain	(nats)	vocab.	≥ 5	top 50	top 300	all	no redact.	human words	AUROC
Oklahoma petitions	6.38	587	3,606	0.146	0.208	0.277	0.277	28%	0.895
Press releases	7.33	1,528	5,672	0.089	0.125	0.211	0.211	44%	0.869
CFPB complaints	6.12	455	2,183	0.177	0.226	0.286	0.271	24%	0.962
UN press releases	6.65	770	3,226	0.107	0.146	0.217	0.217	31%	0.895
arXiv CS abstracts	6.92	1,013	2,650	0.124	0.165	0.230	0.230	32%	0.914

Table 13. Vocabulary concentration and human/AI divergence, one generator (Llama-3.1-8B-Instruct drafting from fact notes). $H(P)$ is the entropy of the human unigram distribution; the effective vocabulary is $\exp H(P)$. The TV columns are the total variation between the human and AI unigram distributions over the whole vocabulary. "TV from rare human words" is the share of the full distance between words with human frequency below 10^{-4}. AUROC is the cross-fit word-occurrence detector from Table 14.

C.9 Vocabulary Concentration and Divergence

A restricted vocabulary results in a narrow set of possibilities for either a model or a human for the next word. We test whether vocabularly narrows the gap between human and AI text in two ways.

Narrowing's impact within a domain. A prompt that asks the model to follow the typical conventions of a genre narrows the writing and moves the two distributions together. We see that the per-unit total variation falls from 0.542 to 0.488 in petitions, from 0.725 to 0.688 in complaints, from 0.564 to 0.522 in UN releases and from 0.488 to 0.471 in press releases. It, however,rises against the pattern in abstracts from 0.586 to 0.636 (Table 14). As see in (Table 13), petitions and complaints have the most concentrated vocabularies, at of 587 and 455 word types against 1,528 for press releases, yet they sit further from their AI text: unigram TV is 0.28 and 0.27 against 0.21.

Factors contributing to text divergence. In a restricted vocabulary, a model cannot use words outside the register. For example, the word *plaintiff*, we find, rises from 17.2 to 28.1 occurrences per 1,000 tokens between petitions and drafts, *court* from 2.8 to 11.6, *oklahoma* from 7.1 to 15.6, while *or* falls from 9.6 to 3.0. In press releases, 44% of the divergence is due to words that are rare in human text, however (against 28% in petitions and 24% in complaints). Word-occurrence detectors estimate common-word rates with far less variance, which is consistent with complaints and petitions being at least as separable as press releases in Table 14, and with the anchored-word result. *Content*, not just a small vocabulary, brings P and Q closer alike (Appendices C.3 and C.4).

		Forced tokens		Units over	Ceiling AUROC		Chunk AUROC (cross-fit)			TV ≥	Doc. AUROC	Task
Domain	Docs	boiler.	anch.	half forced	T=1	T=0.5	rewrite	draft	register	(chunk)	(min)	entropy
Oklahoma petitions	500	26%	9%	28%	0.88	0.76	0.884	0.895	0.869	0.49–0.54	0.992	0.35
Press releases	500	3%	22%	5%	0.96	0.80	0.867	0.869	0.860	0.47–0.49	0.986	0.43
CFPB complaints	500	1%	15%	2%	0.98	0.83	0.953	0.962	0.951	0.69–0.73	0.999	0.57
UN press releases	395	6%	14%	6%	0.97	0.82	0.905	0.895	0.886	0.52–0.56	0.992	0.45
arXiv CS abstracts	500	0%	16%	0%	0.98	0.83	0.874	0.914	0.934	0.50–0.64	0.987	0.53

Table 14. Five domains and one generator. 500 human documents per domain from before ChatGPT (395 UN releases), with Llama-3.1-8B-Instruct producing a rewrite with Liang et al.'s compress-and-expand prompts ("rewrite"), a draft from fact notes, and a register-prompted draft of each. Forced tokens: boilerplate is the share of tokens inside 8-grams that recur in at least 1% of the domain's documents; anchored is the share, outside boilerplate, in words whose presence in the rewrite is forced by their presence in the document. Ceiling: unit average of $\frac{1}{2} + sT - s^2T^2/2$ over the authored share s, at $T = 1$ and $T = 0.5$. Chunk and document AUROC: two-fold cross-fit word-occurrence detector, human documents against each kind of AI text; TV ≥ inverts the bound at the chunk AUROC (range over the three kinds); Doc. AUROC is the minimum over the three. Task entropy: the model's mean next-token entropy in nats while drafting from notes. Each column is defined in Appendix D.

D The Five-Domain Comparison

Table 14 compares legal petitions with four domains in which distributional estimation of AI use has been applied at scale [26, 27]: corporate press releases, consumer complaints, United Nations press releases, and scientific abstracts. Elsewhere in this appendix, comparisons mix generators (i.e., GPT-4o-mini for the legal rewrites, GPT-3.5-turbo for the press-release rewrites) and sample sizes. This makes cross-domain differences very difficult to attribute. Here, every domain receives the same treatment. Specifically, this entails the following conditions: the same number of human documents from before ChatGPT are incorporated (except for UN press releases); the same open model is used to produce the same three kinds of AI text from each document; and every column is computed by the same code. This section describes the data, the generation, and each column of the table in turn.

D.1 Human Documents

Each domain contributes 500 human-written documents from before the release of ChatGPT, except UN press releases, where 395 met the length filter. We use the first 1,200 words of each document.

- **Oklahoma petitions and press releases.** The 500 held-out documents of Appendix C.6: petitions from the 2018–2022 reference corpus and 2021 PR Newswire releases, both drawn from the half of the paired corpus that the deployed detector was never fit on.
- **CFPB complaints.** Consumer complaint narratives received in 2021, taken from the Consumer Financial Protection Bureau's bulk file as archived by the Internet Archive on 2022-02-08. The Bureau's live database stopped providing narrative text in August 2026, so the archived file is the only remaining public source. Narratives of 150–900 words were sampled evenly across the twelve months with a cap per product category within each month (700 collected, 500 used). The Bureau's redaction tokens (XXXX) are left in place.
- **UN press releases.** English releases dated 2021, enumerated from press.un.org's sitemap (1,854 candidate pages). The article pages sit behind a bot check; bodies were thus fetched from Internet Archive captures of the same pages, with the publication date read from each page's own metadata. Releases of 150–1,500 words were kept (401 collected, 395 used). The length filter removes the long meeting-coverage transcripts, so the sample leans toward

shorter series (i.e., Secretary-General statements and appointments, biographical notes, and Security Council summaries).

- **arXiv abstracts.** Computer-science abstracts from the 2021 monthly files distributed with Liang et al. [27], sampled evenly across months and restricted to 120–400 words (700 sampled, 500 used).

Job postings, the remaining domain in Liang et al. [26], are licensed data and are not included. Tokens are lowercased words. Petitions use the tokenization of the deployed legal detector (words of letters only, single characters other than *a* and *i* dropped); the other domains use Liang et al.'s rule (`\b\w+\b`, all-digit tokens dropped).

D.2 AI Text

We generated three types of AI text for each document. In order to do this, we used Llama-3.1-8B-Instruct (bfloat16, vLLM, temperature 0.7, top-p 0.95, and a maximum of 1,200 output tokens). Prior to prompting, we truncated the documents to 1,200 words and 3,000 tokens.

- **Rewrite.** In order to construct the first version, we followed the two-stage generation procedure of Liang et al. [26]. We retained their published prompts while also altering the document type to match each domain: "civil petition filed in an Oklahoma district court", "press release", "consumer complaint", "United Nations press release", "abstract of a scientific paper". In the first stage, the model reduced the source document to a set of bullet points. Here, the temperature was set to 0.3, with a maximum of 700 output tokens. The model then used those bullet points to produce a complete document. This procedure follows the construction of the reference distribution Q that was produced in Liang et al. [26].
- **Draft.** The second version was generated via a separate prompt that describes the source document's facts using brief bullet points. Personal names were replaced with role labels. Given these notes, the model generated a new document in the corresponding genre. We specified a target length equal to the original document's word count, clipped to 150-900 words. Five samples were generated for every document. In the separability analyses, we used the first sample; we included all five when computing the task entropy.
- **Register.** The third version used the same fact notes as the Draft condition, but also included instructions to follow the conventions of the relevant genre. The prompt directed the model to write as an experienced professional would, and also identified characteristic elements of that document type. For legal petitions, example attributes included the opening "COMES NOW the Plaintiff," phrases such as "alleges and states as follows," numbered paragraphs, as well as a concluding "WHEREFORE" paragraph. For press releases, the instructions specified a dateline, executive quotations, and an "About" paragraph. Likewise, prompts for other domains specified their respective writing conventions.

Before evaluating the generated documents, we removed any introductory chat framing (e.g., removing phrases such as, "Here is the petition:" when present). Across generations, between 0.5% and 3% of outputs reached the specified output-token limit.

D.3 Units

For all per-unit measures, the deployed estimator's 40-token chunks are used. Specifically, each document is divided into consecutive 40-token spans. Chunks that are shorter than 10 tokens are excluded from computations. Each chunk is represented by the words that appear within it, regardless of how often they appear. The number of human units is 6,910 for petitions, 6,809 for press releases, 3,912 for complaints, 4,347 for UN releases, and 2,543 for abstracts. Note that because abstracts are shorter, with a median length of 183 words, they have the fewest units.

D.4 The Columns

Boilerplate share, b. We define an 8-gram of tokens as ‘ boilerplate” language if it occurs in at least 1% of the corresponding domain’s human documents. Specifically, we require a minimum of five documents from the domain’s own 500-document sample; a token is then classified as boilerplate if it falls within any such 8-gram. For each human unit, we calculate the fraction of tokens that are covered by boilerplate. We then average these fractions across all units. This is a per-unit average, meaning that each unit will receive equal weight as opposed to each document.

Anchored share, a. We measure the extent to which a word’s presence in an AI rewrite is dependent on its presence in the source document. For each word w that occurs in at least 20 human documents, we calculate

$$\text{anchor}(w) = \Pr(w \in Q_d \mid w \in P_d) - \Pr(w \in Q_d \mid w \notin P_d)$$

over document–rewrite pairs, within quintiles of document length. We then average across these quintiles, weighting by the number of documents that contain w in each. This is described further in Appendix C.3. Words that receive an anchoring score of at least 0.6 are classified as anchored. The column reports the mean, across human units, of the fraction of tokens that are anchored but have not already been counted as boilerplate. Thus, a and b do not overlap, and $1 - a - b$ represents the authored share. Between 158 and 258 distinct words meet the anchoring threshold per domain. The anchoring process also depends on the model used to generate the rewrite. For example, press releases contain 22% anchored tokens in Llama rewrites, but only 8% in Liang et al.’s GPT-3.5-turbo rewrites (Table 6). This is in line with Llama retaining more of the source vocabulary in its rewrites.

Units over half forced. This measure indicates the share of human units in which boilerplate and anchored tokens together comprise more than half of all tokens. It captures the concentration of forced content, which cannot be determined from the two average shares alone. For instance, a domain that averages 35% forced content could have every unit contain roughly one-third forced tokens. Alternatively, it is possible for a quarter of its units to consist almost entirely of forced content. Note that petitions follow the latter pattern.

Ceiling AUROC. Suppose that some fraction s of a unit’s tokens is authored, while the remaining tokens are instead drawn from a distribution that is shared by human and AI writers. Under this assumption, the bound of Sadasivan et al. [42] becomes $\text{AUROC} \leq \frac{1}{2} + s\,T - \frac{1}{2}s^2T^2$. Here, T denotes the total variation between the human and AI distributions of the authored portion (Appendix C.5). For every human unit, we set $s = 1 - a - b$, and then compute the bound. The column provides the average of the unit-level bounds at two levels of separation between human- and AI-authored text. First, we use $T = 1$; in this case, the authored portions are perfectly separable, so the resulting ceiling depends only on the structure of the domain. At $T = 0.5$, the calculation instead uses a value close to the distance measured for realistic AI text. This can be seen in the following columns, as well as in Appendix C.8. The ceiling assumes that AI-generated text retains the material shared with human writing. However, this assumption does not generally hold for the rewrites in our experiment. Llama’s petition rewrites contain 17 boilerplate 8-grams per 1,000 positions, versus 181 in the original petitions. For press releases, the corresponding figures are 0.9 and 23, respectively. Since the rewrites largely omit this material, it is possible for the measured AUROC to exceed the conditional ceiling.

Chunk AUROC, cross-fit. For each type of AI text, we fit and evaluate a word-occurrence detector using two-fold cross-fitting across documents. We randomly divide the documents into two halves using a fixed seed. Using the first half, we estimate per-word occurrence probabilities $P(w)$ and $Q(w)$ from both human and AI units. We restrict the vocabulary to words that appear in at least

five human units and three AI units. Each unit in the held-out half is then scored using the log-likelihood ratio $\sum_w \log \frac{Q(w)}{P(w)}$. The score comprises the contributions from present words given by the preceding sum; also included are the corresponding contributions from absent words. Next, we swap the training and evaluation halves and pool their scores. The column reports the resulting AUROC for distinguishing AI units from human units. Importantly, the detector is fitted on the same type of AI text used for evaluation, but on different documents. This means that we measure the separation the word-occurrence method can achieve for that text at the given sample size. The resulting values are directly comparable across domains and the three types of AI text.

TV $\geq$. We invert the bound using the chunk-level AUROC: $\mathrm{TV} \geq 1 - \sqrt{2(1 - \mathrm{AUROC})}$. This provides a lower bound on the per-unit distance between the human and AI distributions. The rationale for this is two-fold: (1) The underlying bound is tight only for an optimal detector; (2) The word-occurrence model captures only a coarsened representation of the text. The column reports the range across the three types of AI text.

Document AUROC. We average the same cross-fitted scores across the units of each document. We then compare the resulting scores between the AI and human documents. The column reports the minimum AUROC across the three types of AI text. In particular, it represents the weakest document-level separation observed within each domain.

Task entropy. For each domain, we use 200 documents with five draft samples per document. We process each draft's prompt and generated tokens through the same model using Hugging Face Transformers. At each generated token position, we record the entropy of the model's full next-token distribution; this distribution is obtained by applying the softmax to its logits at temperature 1. Entropy is measured in nats. We first average entropy across token positions within each draft. We then calculate the mean across the five drafts for each document, and finally average across documents. This measures how many choices the model faces per token when producing a document in a given genre independently of any detector. The same run also records the model's negative log-likelihood for each human document. This includes two conditions: first, when the drafting task is provided as context; second, when no context is supplied. These results are reported in Appendix C.9.

D.5 What the Table Shows

From the results in Table 14, we can gauge two major differences between legal petitions and other domains, alongside two measures that remain relatively similar. First, petitions contain substantially more forced content. Boilerplate accounts for 26% of their tokens, versus only 0–6% in other domains. Moreover, 28% of petition units are comprised of more than half forced tokens. Meanwhile, no other domains exceed 6%. This pushes down the conditional AUROC ceiling for AI text that preserves shared material. For example, at $T = 1$, the ceiling is 0.88 for petitions versus 0.96–0.98 elsewhere; at $T = 0.5$, it is 0.76 as opposed to 0.80–0.83. This is the chief respect in which legal text differs from the other domains. Petition drafting also has the lowest task entropy, at 0.35 nats per token. The corresponding values range from 0.43 to 0.57 for the other domains; these values increase from press releases to UN releases, abstracts, and consumer complaints.

In spite of these discrepancies, the measured separability of human and AI text is broadly comparable across domains. Across all fifteen domain-and-generation combinations, chunk AUROC ranges from 0.86 to 0.96. Petitions fall near the middle of this range, at 0.87–0.90, while consumer complaints exhibit the highest separability. At the document level, AUROC is at least 0.986 in every case. The three generation methods also produce similar chunk-level results within each domain; the results differ by no more than 0.03, with the exception of abstracts, where the register prompt

augments separability. Thus, the main differences between domains are not simply differences in the observed separation of human and AI text. Instead, domains vary in the degree of material shared between human and AI writing; this limits detection when AI retains that material. They also differ in vocabulary concentration, which affects where the divergence between the distributions occurs (Appendix C.9).

D.6 Caveats

It is important to recognize the limitations and caveats of our experimental process in arriving at the five-domain comparison and results. First, our experiment only has one 8B open model, and we only use one prompt for each type of AI text. The results could be different for a larger model or a differently trained model. Despite this restriction, we used the same generator for all five domains. This was done to maintain consistency across any calculations. Even if the measured values differed under another model, each domain would still be assessed under the same experimental conditions.

Indeed, there are several other limitations that stem from the content anchoring process we used. In particular, we used Llama rewrites to guide our calculations of anchored shares. Thus, the anchored values we calculated are specific to the way Llama in particular processes the source material. Therefore, we are unable to deduce a more general property–one that is consistent across different language models–from these calculations alone. There is also an assumption underlying the conditional AUROC ceiling: We assume that anchored words have the same distribution under P and Q; while the two distributions are similar, they are not identical.

Some of the underlying datasets have their own limitations: (1) The length filter for UN press releases leads to a sample biased toward shorter series; (2) Consumer complaints were pulled from an archived file; (3) Abstracts are relatively short compared to the other text types and thus contribute the fewest units. We must also be cautious about the total-variation values we compute. These are lower bounds obtained from the word-occurrence detector, which does not capture every distinction between human and AI text.

A final consideration concerns the dates of the documents. Our human samples are all derived from the period before ChatGPT was officially released; the AI counterparts, on the other hand, were generated in 2026. The conventions of these genres may have changed in the intervening years, and any such changes would be reflected in the measured distance between the human and AI distributions. Thus, the measured distance may not be explained entirely by differences in authorship.

E Setting Sensitivity

Table 15 lists every value considered for each detector setting (aggregation rules and model hyperparameters). Here we provide additional detail and results for some of these choices.

E.1 EditLens

For EditLens, the stride fraction is how far the next window moves forward relative to the window length. So a stride fraction of 1 means that the window moves an entire window length forward and there is no overlap between consecutive windows. A stride fraction of 0 means total overlap. The window signal controls how the EditLens results are turned into a score, including. The aggregation rules control how those window scores are converted into a document score, including whether short windows are dropped and whether a threshold is used. Table 19 reports the performance for different aggregation rules and Section 3.2 and Figure 2 evaluate the full grid of 216 rules on every county-year.

Method	Setting Choice	Values considered (setting used in bold)
Pangram	text submitted	**OCRed text**; PDF
	document score	**mean of window `ai_assistance_score`**; maximum window `ai_assistance_score`; mean of window `label`; document `prediction_short` label; document `fraction_ai`; document `fraction_ai_assisted`; document 1 − `fraction_human`; document `num_ai_segments`
EditLens	window length	75–500 words in 25-word increments; **250**
	stride fraction	0.50, **0.60**, 0.70, 0.75, 0.80, 0.90, 1.00
	terminal-window rule	**retain all**; retain only if at least 75 words; retain full windows only
	window signal	expected normalized bucket; normalized argmax bucket; $\mathbf{1 - p_0}$; $p_2 + p_3$; p_3; indicators for argmax bucket ≥ 1, ≥ 2, or $= 3$
	document score	**mean**; word-weighted mean; median; 10% trimmed mean; maximum; 75th, 90th, or 95th percentile; mean of top 10%, 25%, or 50%; softmax pooling ($\tau = 0.05$), $\tau = 0.10$, or $\tau = 0.20$; count or longest-run fraction for discrete signals
MLE	estimation unit	**40-token chunks** (final remainder dropped; documents under 40 tokens kept as one unit); sentences; 400-token windows
	vocabulary	minimum occurrence counts in the human and AI references (# in human text, # in AI): (3, 2); **(5, 3)**; (8, 3); (12, 3); (20,3); optional part-of-speech filter
	human reference P	**same-jurisdiction filings from the 2020–22 reference period**; another jurisdiction's filings
	AI reference Q	**LLM rewrite of a redacted skeleton**; two-step summarize-and-redraft; answers to realistic questions; persona-prompts; hallucination documents
	preprocessing	**processed text + exact-sentence deduplication**; raw OCR + deduplication; raw OCR without deduplication
	aggregation to an estimate	**pooled corpus-level** $\widehat{\alpha}_{\text{MLE}}$; per-document fits $\widehat{\alpha}_{d,\text{MLE}}$ summarized by their median, unweighted mean, or unit-weighted mean; share of documents with $\widehat{\alpha}_{d,\text{MLE}}$ above a threshold

Table 15. All values considered for each detector setting, with the values we use in **bold**. Table 1 summarizes these as counts. Rows colored green are *aggregation rules*, the choices made *after* inference about how a document is labeled as AI-assisted; rows colored purple are *model hyperparameters*, which change the scores the model produces.

E.2 Pangram

For Pangram, Table 19 compares the aggregation rules in terms of classification performance, which is discussed in Appendix E.4. Appendix E.5 compares different approaches to creating document-level scores.

E.3 MLE

The word-occurrence MLE estimator has five model hyperparameter choices detailed below.

Vocabulary. We count a word as a member of the vocabulary if it appears at least k times in the human documents and m in the AI documents (referred to as HCk/AICm from here). In Table 16, we demonstrate how the 2025 Oklahoma estimate can vary for different chosen values of HCk/AICm.

Human reference. In Figure 13, we use two versions of the MLE estimator, one trained on Oklahoma filings and one on San Francisco filings as reference P. The AUROC is practically unchanged, although individual detector scores differ vastly.

AI reference. How the AI reference, Q, is generated plays a major role and is connected to Challenge #3 in Section 3. Results for the AI reference are included in Appendix C.8 and Appendix C.6.

Preprocessing. In Table 17, we report how results change when we use either processed text or raw OCR text and when sentences are deduplicated. Raw, unprocessed OCR text actually degrades performance and raises the held-out floor from 1.4% to 7.8–8.2% in Oklahoma, from 1.0% to 1.9–2.3% in San Francisco and from 0.6% to 3.4–3.6% in Santa Clara.

Threshold (HC/AIC)	Estimated α_{MLE} (%)
HC3/AIC2	4.09
HC5/AIC3	4.28
HC8/AIC3	4.59
HC12/AIC3	5.36
HC20/AIC3	6.79

Table 16. MLE vocabulary sensitivity: the 2025 Oklahoma estimate $\hat{\alpha}_{\text{MLE}}$ (%), after excluding the influential filing, across minimum occurrence counts in the human (HC) and AI (AIC) references. Estimates range 4.09–6.79% and stay positive at every threshold.

Jurisdiction	Input channel	Held-out floor	Slope	AUROC
Oklahoma	processed + dedup	1.37%	0.968	0.773
	raw OCR + dedup	7.77%	0.927	0.698
	raw OCR, no dedup	8.15%	0.924	0.720
San Francisco	processed + dedup	1.03%	0.978	0.759
	raw OCR + dedup	2.29%	0.976	0.756
	raw OCR, no dedup	1.87%	0.983	0.762
Santa Clara	processed + dedup	0.61%	0.983	0.765
	raw OCR + dedup	3.60%	0.958	0.740
	raw OCR, no dedup	3.37%	0.965	0.755

Table 17. MLE preprocessing sensitivity. The held-out floor is $\hat{\alpha}_{\text{MLE}}$ on a human half-sample (lower is better; 95% intervals are within ±0.5 points); the slope is the spike-in recovery slope (ideal 1.0); AUROC is on the PhantomDocket documents common to each variant and its baseline. This robustness run used an earlier pre-exclusion snapshot of the controls; the current inventory is reported in Appendix A.3. Paired bootstrap intervals on the change are roughly ±0.02.

Estimation unit	2023	2024	2025	2026
Sentence	1.70%	2.15%	2.91%	2.74%
40-token chunk (used throughout)	1.26%	1.48%	2.12%	1.95%
400-token window	0.60%	0.38%	0.27%	0.11%

Table 18. MLE unit-length sensitivity on the Santa Clara complaints: the pooled annual estimate $\hat{\alpha}_{\text{MLE}}$ under three estimation units, with the reference split, the Q rewrites and the vocabulary held fixed. The 40-token chunk is the prespecified unit used throughout the paper (95% CI 1.76–2.15% in 2026); its 2020–2022 held-out floor is 0.61%.

Unit construction. In Table 18, we compare 3 unit constructions, sentences, 40-token chunks, and 400-token windows. Sentence units are 0.4–0.8 points higher than 40-token chunks in every year. For the 400-token window, the estimates decline and that is one reason we use the 40-token unit.

E.4 Aggregation Rules on PhantomDocket

In Table 19, we report results for the aggregation rules for all three detection methods. For each method and aggregation rule, we report the percentage of documents flagged as AI for the three PhantomDocket groups. The flagged rate for controls is the false positive rate and the flagged

Table 19. Aggregation-rule tradeoffs across PhantomDocket filing groups

Aggregation rule	Same-party flagged (%)	Hallucination flagged (%)	Control flagged (%)
Panel A: Pangram 3.3.2			
Pangram AI-or-Mixed Label	24.3%	47.7%	0.6%
Mean segment score ≥ 0.10	28.3%	54.4%	1.2%
Mean segment score ≥ 0.05	39.4%	68.9%	1.7%
Mean segment score ≥ 0.01	51.5%	84.6%	4.3%
Maximum segment score ≥ 0.10	47.0%	78.9%	3.2%
AI-generated fraction > 0 [45]	24.7%	48.5%	1.0%
Panel B: EditLens			
Mean window score ≥ 0.10	16.2%	33.7%	1.2%
Mean window score ≥ 0.05	37.7%	63.0%	8.0%
Mean window score ≥ 0.01	92.8%	98.0%	84.2%
Maximum window score ≥ 0.10	55.2%	75.7%	31.1%
Any window assigned a nonzero native bucket	26.2%	41.8%	4.7%
Panel C: MLE (Santa Clara reference fit)			
$\widehat{\alpha}_{d,\mathrm{MLE}} \geq 0.10$	32.7%	40.7%	10.1%
$\widehat{\alpha}_{d,\mathrm{MLE}} \geq 0.05$	49.6%	56.2%	19.5%
$\widehat{\alpha}_{d,\mathrm{MLE}} \geq 0.01$	68.4%	75.1%	40.5%

Percentage of filings in each group satisfying the stated flagging threshold. The same-party filings ($n = 538$) are unlabeled, so their true flagging rate unknown, but illustrative of sensitivity. The hallucination rate flag rate for 953 court-identified hallucination filings. The control flagged rate is the false-positive rate.

rate for the hallucination documents is the sensitivity or recall. The flagged rate for the same party documents are shown for reference.

For Pangram, we see that the native labels are conservative on the hallucination documents. It flags only 0.6% of the controls but also only 47.7% of the hallucination filings. This is similar to the any AI-generated rule used by [45]. However, rules based on aggregating the segment scores can recover more of the hallucination set and keep the false positive rate low.

EditLens is more sensitive than Pangram to the aggregation rule. A threshold of 0.10 on the mean window score is highly conservative, with 33.7% sensitivity and a 1.2% false-positive rate. Maximum-window aggregation similarly raises both sensitivity and false detections: at 0.10 it flags 75.7% of hallucination filings and 31.1% of controls.

The document-level MLE coefficients exhibit the same pattern. Using the Santa Clara reference fit, lowering the threshold on $\widehat{\alpha}_{d,\mathrm{MLE}}$ from 0.10 to 0.01 raises benchmark sensitivity from 40.7% to 75.1%, but also raises the control false-positive rate from 10.1% to 40.5%.

E.5 Pangram Aggregation Rules and Benchmark Tradeoffs

For a given document, Pangram returns multiple results. This means that there are many degrees of freedom in constructing a document level score. At the window-level, Pangram returns both a continuous score (`ai_assistance_score`) and a categorical label (`label`). We encode the levels of `label`, `Human Written`, `Lightly AI-Assisted`, `Moderately AI-Assisted`, and `AI-Generated` as 0, 1/3, 2/3, and 1, respectively. Pangram also returns document-level scores. `prediction_short` is a document-level label and we encode the levels as `Human`= 0, `Mixed`= 0.5, and `AI`= 1. Finally,

there are three continuous document-level scores, `fraction_human`, `fraction_ai_assisted`, and `fraction_ai`, which sum to 1.

In Table 20, we compare eigth document-score rules using the PhantomDocket classification documents. Seven of the rules involving a threshold and we find the threshold that maximizes Youden's J = TPR − FPR over four folds and then report results for the fifth fold. The eight rule (`fraction_ai`> 0) does not require a fitted threshold.

We see that the average and max continuous window scores have the highest sensitivities (88.6% and 89.3%) but also the highest false-positive rates (7.2% and 6.9%). Overall, we see that the aggregation rules matter and that this variability comes from how the information is utilized, not a common cutoff.

Table 20. Sensitivity of Pangram classifications to the choice of document score

Document score	Share flagged	TPR	FPR
Document `prediction_short`	26.7%	47.6%	0.9%
Average window `ai_assistance_score`	52.1%	88.6%	7.2%
Max window `ai_assistance_score`	52.4%	89.3%	6.9%
Average window `label`	32.7%	58.0%	1.4%
Document `fraction_ai`	27.0%	47.9%	1.3%
Document `fraction_ai_assisted`	22.3%	39.8%	0.6%
Document 1 − `fraction_human`	32.7%	58.0%	1.4%
Fixed rule: document `fraction_ai` > 0	27.1%	48.0%	1.3%

Out-of-fold classifications using different outputs given by Pangram API output. The optimal Youden's J operating threshold was selected on four folds for each scoring method. The displayed scores are from the out-of-fold fifth fold. The final row implements the rule where a flag is whenever `fraction_ai` field is strictly positive, as used by [45].

E.6 EditLens sensitivity to window geometry and score aggregation

In Tables 23 and 24, we repeat the sensitivity study in E.6 for EditLens.

E.6.1 Sensitivity to window geometry. Table 23 varies window length and stride. TPR ranges from 69.6% to 78.6% and FPR from 6.8% to 14.6%. At 400 words, strides of 0.60 and 0.75 are similar while TPR and FPR are higher when the stride is 0.90 raises.

E.6.2 Sensitivity to window signal and pooling. In Table 24, we keep the window length fixed at 400-word and the stride at 1.00, varying just the score aggregation across windows. The combination of window geometries and score aggregation rules results in a whopping total of 13,452 EditLens pipelines. Among these pipelines, the share of PhantomDocket documents classified as positive ranges from 0.0% to 64.5%. The results in Tables 23 and 24 show one-factor slices of the larger joint grid.

E.7 Does Calibration Remove the Dependence on the Setting?

We compare prevalence estimates across our counties for 216 different EditLens settings. The interquartile range of the raw 2026 flag rate is 65 percentage points in Tulsa, 64 in Oklahoma County, 71 in San Francisco and 77 in Santa Clara. We remove settings that do not have Youden's J on PhantomDocket of at least 0.15, reducing the IQR to 21, 22, 37 and 41, respectively. In Figure 2, we also apply a Rogan-Gladen correction to the prevalence estimates, where the sensitivity is estimated using the hallucination filings and the specificity comes from the county's own 2022-2022 data. We find that the interquartiles ranges for the corrected 2026 estimates, using Rogan-Gladen, are 13, 16, 16 and 21. Validation and correction remove a source of the variation.

Table 21. Reported prevalence given aggregation rules

Aggregation rule	Result
Panel A: Pangram aggregation rules for same-party filings in PhantomDocket (n = 538)	
Unweighted mean segment score ≥ 0.01	51.5%
Unweighted mean segment score ≥ 0.05	39.4%
Unweighted mean segment score ≥ 0.10	28.3%
Maximum segment score ≥ 0.10	47.0%
Native AI-or-Mixed document verdict	24.3%
"Any AI-generated text" rule used by Shah and Levy: vendor AI-generated fraction > 0 [45]	24.7%
Any AI involvement: vendor AI-generated fraction plus AI-assisted fraction > 0 [12]	29.6%
At least 25% of the filing classified as AI-generated or AI-assisted	16.2%
Panel B: EditLens aggregation rules on Santa Clara complaints, 2026 (n = 3,179)	
Unweighted mean window score ≥ 0.01	97.5%
Unweighted mean window score ≥ 0.05	33.7%
Unweighted mean window score ≥ 0.10	9.7%
Maximum window score ≥ 0.10	56.5%
Any window assigned a nonzero native bucket	13.5%
Panel C: MLE aggregation rules on Santa Clara complaints, 2026 (n = 3,168)	
Documents with $\widehat{\alpha}_{d,\mathrm{MLE}} \geq 0.01$	34.6%
Documents with $\widehat{\alpha}_{d,\mathrm{MLE}} \geq 0.05$	16.4%
Documents with $\widehat{\alpha}_{d,\mathrm{MLE}} \geq 0.10$	7.4%

Varied rules and estimator outputs for unknown AI-use document sets. Compare within, rather than across, panels, for model aggregation rule sensitivity.

F EM algorithm for ScoreMix

ScoreMix (Section 4.1) estimates the prevalence of AI usage using raw detector scores. Let s_i be the EditLens score for document i; we model it as drawn from the mixture

$$(1-\alpha)\,P_S(s_i) + \alpha\,Q_S(s_i),$$

where α is the mixture parameter and P_S and Q_S are the densities of s_i for human-written and for AI-assisted documents. For the EM algorithm, W_i is the latent indicator variable for whether s_i was drawn from Q_S. For the E-step we calculate

$$E(W_i \mid s_i, \alpha^{(t)}) = \frac{\alpha^{(t)} R(s_i)}{\alpha^{(t)} R(s_i) + (1-\alpha^{(t)})},$$

where $R(X_i) = Q_S(X_i)/P_S(X_i)$. We note that

$$\begin{aligned} Q_S(s_i) &= f(s_i \mid W_i = 1) \\ &= \frac{\Pr(W_i = 1 \mid s_i)\,\alpha'}{f(s_i)} \end{aligned}$$

by Bayes' rule. α' is the proportion of AI documents across the documents used to define P_S and Q_S. Implying that

$$R(s_i) = \frac{\Pr(W_i = 1 \mid s_i)\,\alpha'}{\Pr(W_i = 0 \mid s_i)\,(1-\alpha')}.$$

Table 22. Sensitivity of corpus-level summaries to aggregation rules

Corpus-level aggregation rule	**Result**
Panel A: Pangram summaries for same-party filings in PhantomDocket (n = 538)	
Unweighted mean AI-generated fraction	7.7%
Word-weighted mean AI-generated fraction	6.5%
Unweighted mean AI-generated-plus-assisted fraction	9.7%
Word-weighted mean AI-generated-plus-assisted fraction	8.1%
Unweighted mean maximum segment score	0.284
Panel B: EditLens summaries for Santa Clara complaints, 2026 (n = 3,179)	
Median of filing-level mean window scores	0.038
Unweighted mean of filing-level mean window scores	0.050
Equal-window mean score	0.049
Word-weighted mean window score	0.049
Unweighted mean of filing-level maximum window scores	0.165
Panel C: MLE summaries for Santa Clara complaints, 2026 (n = 3,168)	
Median filing-level $\widehat{\alpha}_{d,\mathrm{MLE}}$	0.00%
Unweighted mean filing-level $\widehat{\alpha}_{d,\mathrm{MLE}}$	2.46%
Unit-weighted mean filing-level $\widehat{\alpha}_{d,\mathrm{MLE}}$	2.43%
Pooled corpus-level $\widehat{\alpha}_{\mathrm{MLE}}$	1.95%

We fix estimator outputs and corpus and vary the aggregation rule. Note that not all of the estimates share a common estimand. The MLE's pooled $\widehat{\alpha}_{\mathrm{MLE}}$ is a mixture estimate of the share of text units that resemble the AI distribution in its reference. The Pangram panel uses the same-party PhantomDocket corpus, but the EditLens and MLE panels use 2026 Santa Clara complaints. Magnitudes should only be compared within panels.

Table 23. Sensitivity of EditLens classifications to window geometry

Window length	Stride fraction	Share flagged	TPR	FPR
75	0.50	43.7%	72.2%	8.7%
100	0.75	47.6%	75.6%	13.1%
150	0.75	43.2%	70.9%	9.1%
225	0.70	46.2%	75.3%	10.3%
250	0.60	48.3%	78.2%	11.5%
300	0.50	46.2%	74.6%	11.2%
375	0.90	46.4%	73.2%	13.5%
400	0.60	41.7%	70.0%	6.8%
400	0.75	41.9%	70.3%	6.9%
400	0.90	47.5%	74.1%	14.6%
400	1.00	42.7%	69.6%	9.6%
450	0.60	46.0%	73.1%	12.7%

We estimate $\Pr(W_i = 1 \mid s_i)$ with a generalized additive model (GAM) fit on the documents used to define P_S and Q_S: the court's own 2020–2022 filings for P_S and the hallucination filings in PhantomDocket for Q_S. For the M-step,

$$\alpha^{(t+1)} = \frac{1}{n} \sum_i E(W_i \mid s_i, \alpha^{(t)}),$$

Table 24. Sensitivity of EditLens classifications to score aggregation at a fixed geometry

Aggregation rule	Share flagged	TPR	FPR
Expected bucket, mean	42.7%	69.6%	9.6%
Drop terminal windows under 75 words	41.9%	69.6%	7.7%
Full windows only	44.5%	72.1%	10.5%
Normalized argmax bucket, mean	23.0%	39.9%	2.1%
$1 - p_0$, mean	47.2%	74.6%	13.5%
$p_2 + p_3$, mean	43.1%	68.1%	12.4%
p_3, mean	42.5%	65.7%	14.0%
Indicator(argmax ≥ 1), mean	22.9%	39.8%	2.1%
Indicator(argmax ≥ 2), mean	7.0%	12.2%	0.6%
Indicator(argmax $= 3$), mean	4.9%	8.4%	0.5%
Expected bucket, word-weighted mean	42.6%	70.4%	8.3%
Expected bucket, median	46.0%	71.5%	14.6%
Expected bucket, maximum	45.5%	68.7%	17.0%
Expected bucket, 75th percentile	43.5%	69.9%	11.0%
Expected bucket, 90th percentile	40.3%	65.8%	8.8%
Expected bucket, top-25% mean	41.2%	66.4%	10.1%
Expected bucket, softmax ($\tau = .05$)	45.9%	71.3%	14.5%

Window geometry remains fixed at 400 words with stride fraction 1.00. For each aggregation, the operating threshold maximized Youden's J and was applied only on the withheld fold.

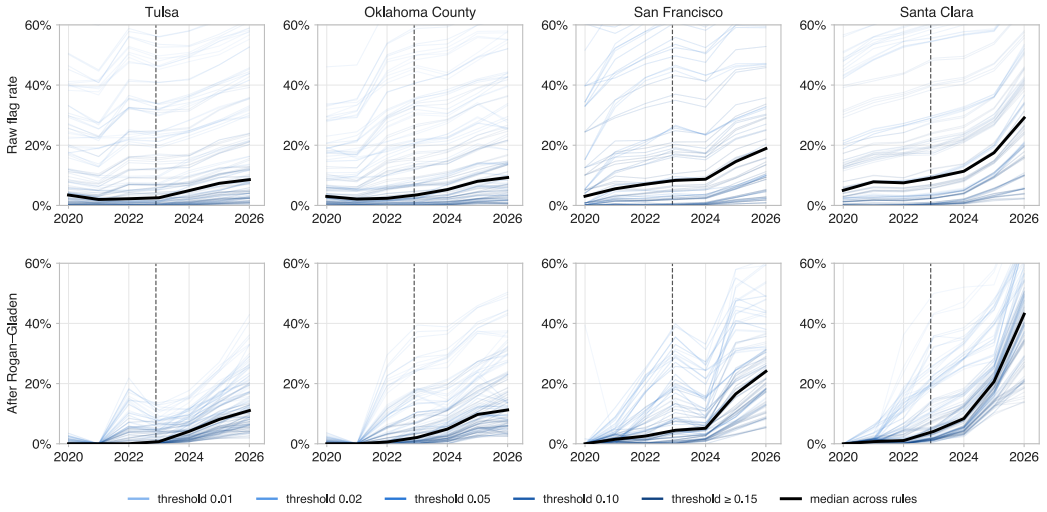


Fig. 12. Every setting before (top) and after (bottom) Rogan–Gladen correction with rule-specific sensitivity from PhantomDocket and county-specific pre-ChatGPT false-positive rates. The thick line is the median across rules.

where n is the number of complaints. We bootstrap both the datasets used to define P_S and Q_S and the complaints for bootstrap intervals.

This is equivalent to the probabilistic classify and count approach with the EMQ algorithm used in [4, 5, 43]. Others, such as [1], have explored this approach, but we are the first to apply it to the AI detection setting by grounding an AI detector with a benchmark.

G Hyperparameter Tuning

We employ PhantomDocket to help validate a choice of EditLens setting from those in Table 1. We use a Tree-Structured Parzen Estimator (TPE) hyperparameter tuning procedure with 244 of our hallucination filings, 132 of our opposing-party controls, and 66 judicial orders held out as a test. The process discovered an optimal setting of: 250-word windows, stride of 0.60, window scores using the probability assigned to any-AI bucket ($1 - p_0$), and averaging across windows. We optimize on a partial AUROC (pAUROC) with the fpr at less than 0.10, so as to find a reliable classifier setting that isn't overly rewarded for false positives. The search began with 13 geometries where window lengths ranged from 75 to 500 words and stride fractions from 0.50 to 1.00. TPE would then propose the next geometries to try, resulting in a total of 40 explored.

Table 25. Selection of the EditLens classification geometry

Window words	Stride fraction	Nested-CV pAUROC	Validation pAUROC
225	.60	.836 (.016)	.800
225	.50	.824 (.011)	.794
250	**.60**	**.821 (.003)**	**.820**
100	.75	.817 (.021)	.784
75	1.00	.816 (.019)	.782

Results of our TPE optmization. Among the five geometries, 250/.60 had the highest selection-validation partial AUROC. Standard errors in parentheses.

Table 26. Locked-test performance of the selected and original EditLens pipelines

Pipeline	Opposing-party pAUROC	Judicial-order pAUROC	Macro pAUROC
Selected: 250/.60, retain, $1 - p_0$, mean	.849	.833	**.841**
Original: 400/1.00, retain, expected bucket, mean	.817	.753	.785

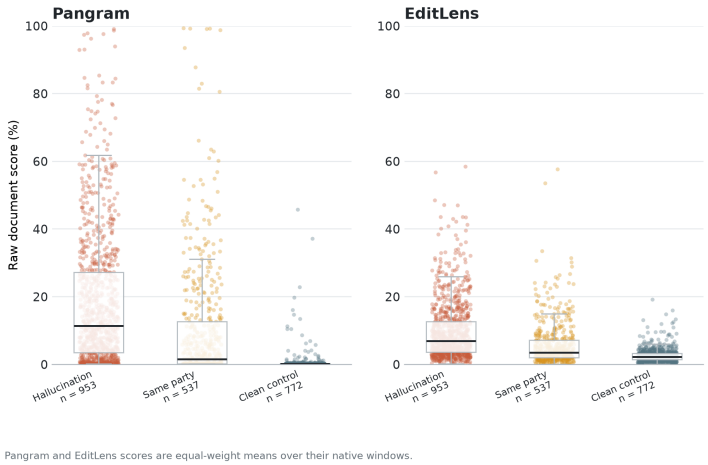
Comparing the default geometry we began with with the one discovered as a result of optimization. The paired pAUROC difference was .056 (95% CI, .017–.093) on the locked test set.

G.1 Classification sensitivity across EditLens specifications

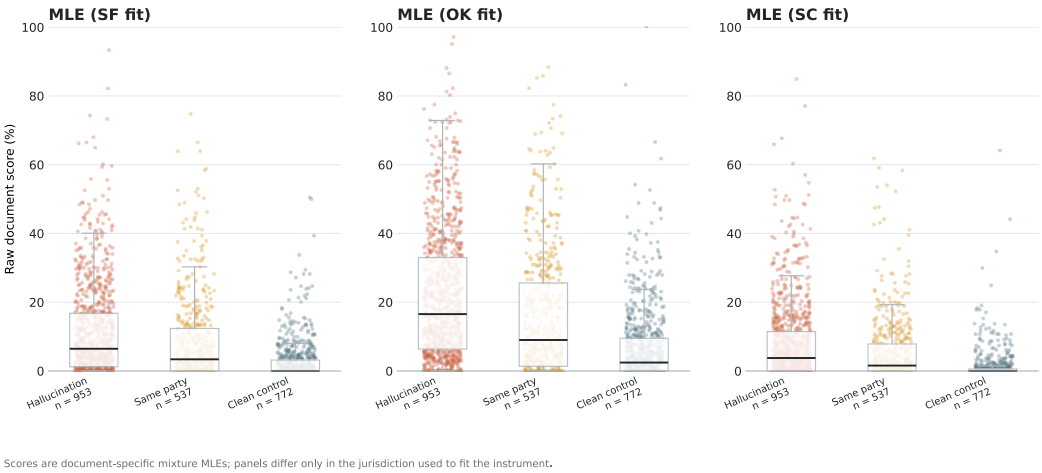
To understand whether the classification performance was more sensitive to the aggregation rule or the window geometry, we looked at the combination of 354 aggregation rules and 38 geometries. Across these 13,452 combination, the aggregation rules explained 90.6% of the variation the geometries explained 4.0%, with the remainder being explained by the interaction.

H Benchmark Score Distributions and ROC Curves

Figure 13 demonstrates the PhantomDocket document-level scores for each of the detection methods in a boxplot. Hallucinated filings are not perfectly separable, but score over higher ranges than the control documents. We see that, for even Pangram, some control documents receive higher



(a) Neural-network-based detectors.



(b) MLE estimators fit on San Francisco, Oklahoma, and Santa Clara filings.

Fig. 13. **Raw document-level detector scores by filing group.** Each point represents one filing in PHANTOMDOCKET: 953 filings with documented AI hallucinations, 538 other filings submitted by the same party, and 771 controls. The controls include 509 opposing-party filings and 262 judicial orders. Boxes show medians and interquartile ranges. The top row compares document scores using the mean Pangram segment score and mean EditLens window score. The bottom row shows the document-level mixture estimate, $\hat{\alpha}_{d,\mathrm{MLE}}$, fitted on the Oklahoma, San Francisco or Santa Clara references.

scores akin to the hallucination group and some hallucinated documents receive low scores akin to the control group. The same-party filings are used as an unknown set, where we suspect an unknown mix of AI and human writing. The resulting ROC curves are in Figure 14 for classification performance. Pangram is the best classifier, followed by EditLens, and the MLE methods are further behind the two.

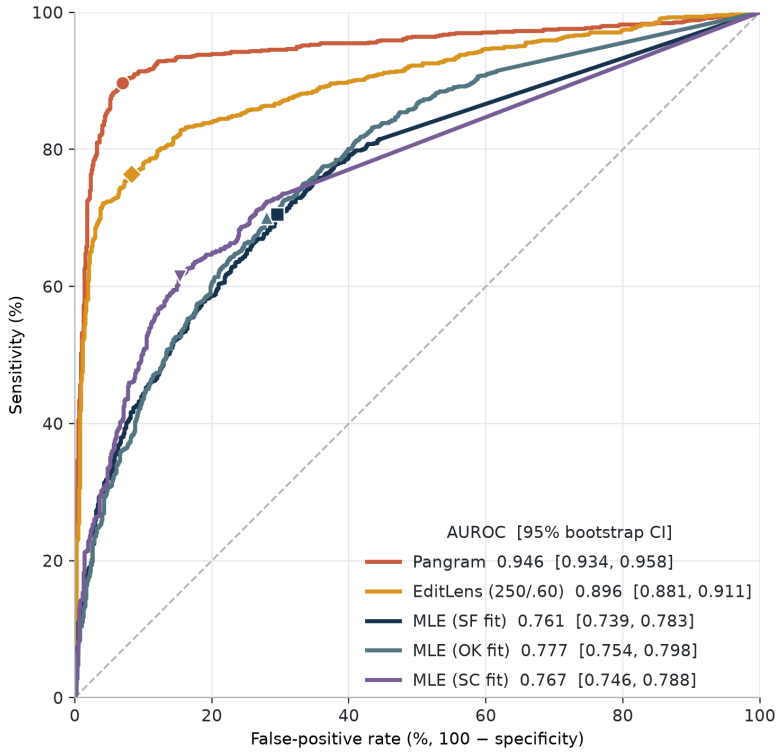


Fig. 14. **Detector performance on PhantomDocket.** Receiver operating characteristic curves compare 953 filings with documented AI hallucinations against 771 controls. The controls include 509 opposing-party filings and 262 judicial orders. Pangram and EditLens scores are unweighted means of their native window scores; each MLE score is a document-level mixture estimate, $\hat{\alpha}_{d,\mathrm{MLE}}$, obtained using the indicated jurisdiction-specific fit. Markers identify the threshold maximizing sensitivity plus specificity. Curves connect operating points at unique score thresholds; linear interpolation across tied scores corresponds to assigning positive–negative ties half credit. The legend reports AUROC estimates with 95% confidence intervals from 5,000 document-stratified bootstrap draws.

I Additional Results on the Trial Courts

I.1 Case Mix by County

Table 28 shows 2026 cases by issue family, alongside the share of filings with a mean EditLens score ≥ 0.05 in that family. The issue families labels are determined by deterministically mapping the native county labels to a harmonized set. The Oklahoma counties are densely populated by contract and collections cases; San Francisco and Santa Clara have far more employment and personal injury cases. Composition explains some, but not all of the gap between the two states. Reweighting each county within-family to San Francisco's family mix moves Oklahoma County filings flagged from 7.3% to 8.5% and Tulsa from 8.4% to 10.7%. San Francisco's rate is at 19.1%. Within most issue families, the California counties sit at two to four times the Oklahoma flag rate, but the counties are close in contract cases. Composition makes a difference across the California counties: reweighted to San Francisco, Santa Clara falls from 33.7% to 25.5%. Santa Clara's employment cases, which tend to be flagged higher, may be bringing the rate up.

Table 27. Pro se complaint counts and shares by jurisdiction and year

Year	Oklahoma County	Tulsa	San Francisco	Santa Clara	All jurisdictions
2020	193 (3.32%)	76 (1.94%)	50 (3.55%)	142 (4.74%)	461 (3.26%)
2021	134 (2.58%)	83 (2.27%)	91 (2.47%)	132 (4.73%)	440 (2.87%)
2022	118 (2.03%)	81 (2.07%)	75 (2.04%)	119 (3.94%)	393 (2.39%)
2023	141 (2.08%)	96 (2.11%)	37 (1.36%)	159 (4.56%)	433 (2.47%)
2024	186 (2.42%)	129 (2.65%)	86 (2.63%)	167 (3.95%)	568 (2.83%)
2025	156 (1.79%)	193 (3.24%)	233 (3.99%)	290 (6.05%)	872 (3.44%)
2026	239 (3.56%)	82 (1.91%)	193 (4.25%)	266 (8.37%)	780 (4.16%)
Total	1,167 (2.50%)	740 (2.38%)	765 (3.04%)	1,275 (5.20%)	3,947 (3.10%)

Table 28. Case mix and estimated AI-assisted drafting prevalence by issue family, 2026

	Oklahoma County		Tulsa		San Francisco		Santa Clara	
Issue family	Share	$\hat{\alpha}$	Share	$\hat{\alpha}$	Share	$\hat{\alpha}$	Share	$\hat{\alpha}$
Business/commercial	0.8%	—	1.3%	—	2.6%	83.4%	5.1%	81.5%
Civil rights	0.1%	—	0.4%	—	1.9%	—	1.7%	—
Collections/debt	48.5%	0.9%	46.3%	1.2%	45.8%	0.0%	0.0%	—
Contract	20.2%	11.8%	27.2%	8.5%	10.8%	21.1%	30.0%	15.8%
Employment	0.2%	—	0.1%	—	4.2%	98.3%	26.0%	59.2%
Family/probate/trust	0.0%	—	0.0%	—	0.0%	—	0.0%	—
Insurance	7.1%	—	0.5%	—	0.6%	—	1.8%	—
Other/unspecified	1.7%	29.9%	0.3%	—	12.3%	22.9%	3.7%	49.6%
Personal injury/tort	19.6%	21.4%	21.4%	28.1%	17.5%	30.2%	25.6%	26.1%
Procedural/equitable	1.2%	—	1.5%	—	0.0%	—	0.0%	—
Real property/housing	0.5%	—	1.0%	—	4.3%	27.0%	6.1%	46.1%
All filings	100.0%	8.3%	100.0%	9.3%	100.0%	15.4%	100.0%	37.8%

Share is the percentage of each court's 2026 complaints assigned to respective issue family, while $\hat{\alpha}$ is our ScoreMix estimate of AI-assisted drafting prevalence. Dashes indicate either fewer than 100 complaints in the pooled 2020–2022 human-reference period or fewer than 100 complaints in 2026. Santa Clara collections/debt is structurally under collected. The all-filings estimates use each court's complete 2020–2022 complaint distribution as the human reference.

Table 29 lists every Judicial Council case type with at least 30 complaints filed in 2026 in the two California courts, pooled. It extends Table 3b by reporting threshold-free mean-score contrasts separately for 2020–22 and 2026, together with each type's complaint count and share. Table 30 separately breaks the thresholded flag rates down by county. Representation does not explain the gap between the states either. Pro se plaintiffs file 3.2% of the 2026 Oklahoma County complaints, 1.9% in Tulsa, 4.2% in San Francisco and 8.4% in Santa Clara, and among attorney-filed complaints alone the 2026 flag rates are 6.9%, 8.1%, 18.7% and 31.9%.

I.2 Representation Status

Case metadata may not represent situations in which a litigant lost or gained representation after filing a petition/complaint. So, we classify representation status from the initiating document itself. We deterministically search for affirmative evidence of self-representation or attorney-representation, including keywords like "pro se", "pro per", attorney signature blocks, law-firm names, and/or bar numbers. Each document was then assigned a label `attorney`, `pro_se`, `conflicting`, or `unknown`. Any document tagged with `conflicting`, or `unknown` was reviewed by GPT-4o mini. Anything still uncategorized received human review. Among 127,518 documents, 123,525 were labeled as attorney-represented, 3,947 as pro se, 25 conflicting, and 21 unknown.

	2020–22			2026			
Case type	*n*	%	Pr(higher)	*n*	%	Pr(higher)	Log-odds$_{>.20}$
Wrongful termination	657	3.7%	73.4% ± 1.8	499	6.5%	82.3% ± 1.7	+1.17 ± 0.74
Business tort / unfair practice	280	1.6%	62.1% ± 3.1	178	2.3%	78.2% ± 3.5	+0.73 ± 1.92
Defamation	75	0.4%	56.4% ± 6.1	31	0.4%	69.3% ± 10.0	+2.19 ± 1.79
Product liability	200	1.1%	70.4% ± 2.7	94	1.2%	66.5% ± 5.1	+2.55 ± 0.88
Fraud	297	1.7%	56.1% ± 3.1	89	1.2%	68.8% ± 5.6	+1.96 ± 1.24
Professional negligence	112	0.6%	51.4% ± 4.6	50	0.6%	67.9% ± 7.6	+1.69 ± 1.76
Other contract	195	1.1%	52.9% ± 3.7	56	0.7%	51.8% ± 7.9	+1.94 ± 1.91
Civil rights	301	1.7%	61.4% ± 2.8	139	1.8%	69.6% ± 4.2	+0.63 ± 1.73
Wrongful eviction	302	1.7%	73.3% ± 2.6	98	1.3%	69.4% ± 4.3	+0.99 ± 1.74
Other employment	1,044	5.9%	72.9% ± 1.3	516	6.7%	70.9% ± 1.6	−0.74 ± 1.72
Other personal injury	1,459	8.3%	58.7% ± 1.5	548	7.1%	60.6% ± 2.4	+0.77 ± 0.79
Other tort	197	1.1%	57.0% ± 3.8	43	0.6%	62.2% ± 8.4	+0.16 ± 0.41
Medical malpractice	195	1.1%	52.0% ± 3.9	80	1.0%	63.3% ± 5.6	−0.46 ± 0.35
Other real property	620	3.5%	49.2% ± 2.1	169	2.2%	58.8% ± 3.9	−0.10 ± 1.30
Quiet title	312	1.8%	54.2% ± 3.0	117	1.5%	54.4% ± 4.8	+0.81 ± 1.73
Other complaint	1,230	7.0%	61.0% ± 1.5	675	8.7%	57.2% ± 2.1	−0.41 ± 1.24
Mass tort	318	1.8%	72.3% ± 2.1	112	1.5%	66.9% ± 2.1	−0.80 ± 0.32
Breach of contract / warranty	4,002	22.8%	48.3% ± 0.9	1,385	17.9%	47.8% ± 1.6	+0.00 ± 0.70
Auto injury	1,451	8.3%	42.0% ± 1.6	596	7.7%	44.4% ± 2.2	−2.53 ± 0.26
Insurance coverage	348	2.0%	42.7% ± 2.7	76	1.0%	34.8% ± 6.2	+0.72 ± 1.30
Asbestos	172	1.0%	57.6% ± 2.3	47	0.6%	45.0% ± 3.5	+0.07 ± 0.40
Collections	189	1.1%	34.6% ± 4.1	96	1.2%	26.2% ± 4.9	−0.64 ± 0.32
Collections (Rule 3.740)	3,384	19.3%	23.9% ± 0.9	1,987	25.7%	19.2% ± 1.0	−3.96 ± 0.26
All California complaints	17,565	100%		7,724	100%		

Table 29. **California case types before and after ChatGPT.** For each period, Pr(higher) is the probability that a randomly selected complaint of that type has a higher EditLens document score than a randomly selected complaint of any other type in that year, half credit for ties. 50% indicates no separation. For 2026, Log-odds$_{>.20}$ compares the odds that the document score exceeds 0.20 for the indicated type against all other California complaints. Only 74 complaints exceeded 0.20 in 2026; finite estimates use a 0.5-cell correction. Values are accompanied by the larger half-width of their 95% intervals from 5,000 jurisdiction-and-year-stratified document bootstrap draws. We use 400-word non-overlapping windows, the expected-bucket window score, and the mean across windows.

I.3 Labelling Paragraph Function

We label paragraphs by a function that they serve in the document for paragraphs scoring above 0.10 on EditLens and an equal number scoring ≤ 0.10 in the same county and year. GPT-5.6 proposes labels for the paragraphs, which may range from "alleges breach of duty," "recounts the collision," to "reproduces a promissory note." We do not force the model to follow any particular taxonomy. We embed and cluster the labels into 72 groups using `bge-small-en-v1.5` and k-means. Those functions are then assigned to a standard section of the complaint following FRCP 8(a) and 10(b). Tables 31 and 32 list the resulting functions and the log-odds that the EditLens score is above 0.20.

Case type	*n* (2026)	% flagged 2026	% flagged 2020–22
San Francisco			
Product liability	37	73.0%	6.4%
Wrongful termination	187	71.1%	15.9%
Business tort / unfair practice	73	64.4%	12.8%
Civil rights	85	61.2%	8.0%
Fraud	36	55.6%	5.8%
Medical malpractice	38	47.4%	6.8%
Wrongful eviction	78	41.0%	22.1%
Other personal injury	262	34.7%	8.8%
Other complaint	558	28.7%	9.9%
Quiet title	117	27.4%	6.4%
Mass tort	112	21.4%	7.5%
Breach of contract / warranty	490	21.2%	5.3%
Auto injury	253	14.6%	4.3%
Collections	96	3.1%	5.3%
Collections (Rule 3.740)	1,987	2.8%	3.3%
Asbestos	47	2.1%	0.0%
All San Francisco complaints	4,545	19.1%	6.1%
Santa Clara			
Wrongful termination	312	71.2%	15.5%
Business tort / unfair practice	105	66.7%	9.7%
Fraud	53	54.7%	9.0%
Other complaint	117	43.6%	6.0%
Civil rights	54	40.7%	5.6%
Other tort	43	39.5%	4.6%
Other employment	516	38.6%	10.9%
Other personal injury	286	36.4%	9.3%
Medical malpractice	42	35.7%	7.4%
Other real property	169	34.3%	4.2%
Other contract	56	23.2%	6.2%
Product liability	57	21.1%	18.2%
Breach of contract / warranty	895	17.5%	4.0%
Auto injury	343	16.9%	6.1%
Insurance coverage	53	11.3%	1.9%
All Santa Clara complaints	3,179	33.7%	6.7%

Table 30. **Flag rates by case type and county, California.** Every Judicial Council case type with at least 30 complaints filed in 2026, with the share of those complaints flagged (mean EditLens window score of at least 0.05) and the share flagged among the same type's 2020–2022 complaints, where every flag is a false positive. Case types are read from the civil cover sheet; Santa Clara's limited and unlimited divisions of a type are pooled. Santa Clara's collection excluded collections, Rule 3.740, unlawful-detainer, mass-tort and asbestos cases (Section 2.1).

Section / function	log-odds (95% CI)	*n*
Caption & parties		
Identifies Case And Parties	−0.76 [−0.87, −0.64]	4,781
Identifies and Defines Parties	−1.06 [−1.24, −0.87]	2,339
Jurisdiction & venue		
Establishes Jurisdiction and Venue	−0.55 [−0.70, −0.40]	2,336
Statement of facts		
Recounts Prelitigation Factual History	+0.91 [+0.78, +1.05]	1,138
Describes Injury and Harm	+0.85 [+0.73, +0.98]	1,368
Quotes External Statements And Authorities	+0.83 [+0.68, +0.97]	1,015
Describes Employment Relationships and Performance	+0.73 [+0.58, +0.88]	1,002
Explains Claims And Supporting Context	+0.64 [+0.54, +0.74]	2,312
Alleges Defects And Failed Repairs	+0.46 [+0.31, +0.60]	1,168
Alleges Unsafe Property Conditions	+0.39 [+0.24, +0.55]	1,093
Presents Factual Allegations	+0.37 [+0.30, +0.45]	5,045
Recounts Case Background	+0.24 [+0.07, +0.40]	1,129
Alleges Debt and Payment Default	−0.25 [−0.39, −0.12]	2,458
Describes Property And Interests	−1.11 [−1.41, −0.82]	989
Causes of action		
Alleges Access and Privacy Violations	+1.22 [+1.09, +1.34]	1,118
Alleges Employment Discrimination	+1.06 [+0.93, +1.19]	1,182
Alleges Harassment Assault and Retaliation	+0.96 [+0.84, +1.08]	1,351
Alleges Unlawful Employment Practices	+0.87 [+0.77, +0.98]	1,980
States Termination Rights And Claims	+0.72 [+0.57, +0.87]	1,017
Pleads Injury And Damages	+0.58 [+0.51, +0.65]	6,105
Alleges Legal Violations And Misconduct	+0.51 [+0.42, +0.59]	3,665
Alleges Failures and Violations	+0.45 [+0.35, +0.54]	3,188
Alleges Fraudulent Or Deceptive Conduct	+0.44 [+0.32, +0.56]	1,982
Alleges Negligence Causing Harm	+0.40 [+0.31, +0.50]	3,232
Incorporates and Cites Supporting Authority	+0.40 [+0.32, +0.48]	4,696
Alleges Supporting Liability Elements	+0.39 [+0.29, +0.48]	3,112
Alleges Landlord and Tenancy Violations	+0.38 [+0.22, +0.55]	992
Alleges Wage Payment Violations	+0.32 [+0.18, +0.46]	1,417
Pleads Claims And Liability	+0.16 [+0.07, +0.25]	4,274
Alleges Safety Failures And Knowledge	+0.15 [+0.03, +0.26]	2,638
Asserts Claims Liens and Remedies	+0.08 [−0.04, +0.20]	2,222
States Governing Rules and Terms	−0.00 [−0.14, +0.14]	1,765
Pleads Breach of Contract	−0.09 [−0.24, +0.07]	1,528
Enumerates Claims and Case Categories	−0.13 [−0.33, +0.07]	964
Asserts Liability Theories	−0.14 [−0.27, +0.00]	2,003
Establishes Governing Legal Framework	−0.33 [−0.47, −0.19]	2,259
States Warranty Rights and Breaches	−0.60 [−0.81, −0.38]	1,181

Table 31. **Functions within each section of a complaint (part 1: caption through causes of action).** Every paragraph function (the 72 named clusters of Appendix I.3) nested under the section it was assigned to, with the log-odds that a window carrying it scores above 0.20 on EditLens rather than at or below it, and the number of labelled windows carrying it. Cells are tinted by the magnitude of these thresholded log-odds. Each function is assigned to exactly one section (Appendix I.3).

Section / function	log-odds (95% CI)	*n*
Prayer for relief		
Demands Damages and Relief	+0.12 [+0.01, +0.24]	2,489
Requests Judgment And Relief	−0.10 [−0.23, +0.02]	2,392
Demands Judgment And Remedies	−0.56 [−0.69, −0.43]	3,086
Seeks Fees And Costs	−0.61 [−0.78, −0.43]	1,777
Invokes Foreclosure Rights and Remedies	−0.78 [−1.06, −0.50]	820
Verification, signature, service		
Provides Document Administration Material	−0.24 [−0.39, −0.09]	1,783
Authenticates and Certifies Filing Matters	−1.20 [−1.42, −0.98]	1,951
Attachments & exhibits		
States Available Remedies	+0.16 [+0.05, +0.28]	2,524
Provides Required Notice Terms	+0.11 [−0.01, +0.22]	2,517
States Security and Privacy Terms	−0.03 [−0.18, +0.12]	1,585
Waives Procedural Rights	−0.05 [−0.19, +0.10]	1,771
Defines Terms and Introduces Claims	−0.08 [−0.19, +0.02]	3,485
Preserves Contractual Rights and Duties	−0.20 [−0.43, +0.03]	736
Reproduces Financial or Contract Records	−0.26 [−0.36, −0.17]	5,356
Imposes Contractual Duties	−0.30 [−0.46, −0.14]	1,643
Itemizes Account Financial Activity	−0.31 [−0.47, −0.14]	1,641
Defines Payment Duties And Procedures	−0.35 [−0.48, −0.23]	2,854
Allocates Contractual Costs and Obligations	−0.38 [−0.56, −0.20]	1,383
States Insurance Terms And Obligations	−0.47 [−0.66, −0.29]	1,475
Grants And Reserves Contractual Rights	−0.52 [−0.66, −0.38]	2,489
Provides Credit Disclosures and Representations	−0.55 [−0.70, −0.39]	2,131
States General Contract Provisions	−0.63 [−0.81, −0.45]	1,762
Grants Contractual Creditor Authorizations	−0.64 [−0.78, −0.50]	2,845
Provides Dispute Procedures	−0.66 [−0.82, −0.49]	2,186
Sets Contractual Payment Terms	−0.67 [−0.91, −0.44]	1,070
Defines Loan Terms and Obligations	−0.68 [−0.83, −0.53]	2,667
Presents Supporting Exhibits	−0.72 [−0.82, −0.63]	6,963
Addresses Disclosures And Information Sharing	−0.77 [−1.03, −0.50]	911
Sets Arbitration Requirements	−0.82 [−1.00, −0.63]	1,944
Imposes Contractual Restrictions and Limitations	−0.87 [−1.05, −0.68]	2,003
States Financial Agreement Terms	−0.89 [−1.10, −0.69]	1,656
States Financial Rights and Obligations	−0.96 [−1.17, −0.74]	1,629
Sets Lease Rights And Duties	−1.06 [−1.31, −0.81]	1,252
Establishes Account Terms And Status	−1.18 [−1.44, −0.92]	1,313
Explains Financial Charge Calculations	−1.29 [−1.49, −1.08]	2,318

Table 32. **Functions within each section of a complaint (part 2: prayer for relief through attachments).** Every paragraph function (the 72 named clusters of Appendix I.3) nested under the section it was assigned to, with the log-odds that a window carrying it scores above 0.20 on EditLens rather than at or below it, and the number of labelled windows carrying it. Cells are tinted by the magnitude of these thresholded log-odds. Each function is assigned to exactly one section (Appendix I.3).

	2020–22				2026			
Section of case	*n*	%	Pr(higher)	Log-odds$_{>.20}$	*n*	%	Pr(higher)	Log-odds$_{>.20}$
Causes of action	2,834	57.5%	56.8% ± 2.1	−0.30 ± 0.23	9,833	58.5%	68.9% ± 1.4	+1.05 ± 0.16
Statement of facts	1,639	33.2%	54.7% ± 1.9	−0.38 ± 0.25	5,586	33.2%	65.7% ± 1.2	+0.92 ± 0.11
Definitions	27	0.5%	53.0% ± 11.0	−0.40 ± 1.36	90	0.5%	63.4% ± 4.8	+0.30 ± 0.65
Prayer for relief	895	18.1%	47.3% ± 2.3	−0.30 ± 0.34	3,123	18.6%	47.2% ± 1.3	−0.28 ± 0.14
Jurisdiction & venue	263	5.3%	46.0% ± 3.4	−0.65 ± 0.74	776	4.6%	47.6% ± 2.0	−0.50 ± 0.25
Verification, signature	507	10.3%	38.3% ± 5.1	−0.07 ± 0.56	1,237	7.4%	36.5% ± 2.9	−0.82 ± 0.34
Attachments & exhibits	2,658	53.9%	38.6% ± 1.9	+0.01 ± 0.23	9,258	55.1%	30.6% ± 1.2	−1.01 ± 0.13
Caption & parties	739	15.0%	41.4% ± 2.2	−0.92 ± 0.45	1,784	10.6%	46.2% ± 1.4	−0.62 ± 0.17
Venue	76	1.5%	38.0% ± 5.5	−0.58 ± 1.93	242	1.4%	38.4% ± 3.1	−1.45 ± 0.80
All labeled windows	4,932	100%			16,813	100%		

Table 33. **Complaint sections before and after ChatGPT.** Pr(higher) is the probability that a randomly selected window has a higher expected-bucket EditLens score than a randomly selected window not tagged as that section type. Ties are half credit. 50% indicates no separation. Log-odds$_{>.20}$ is the log odds ratio comparing whether the window's expected-bucket score exceeds the 0.20 threshold inside versus outside the section, so zero indicates no association there. Red denotes values above 50% or zero, respectively, and green denotes values below. Estimates are accompanied by the larger half-width of their 95% intervals from 5,000 bootstrap draws. The 2020–22 sample contains San Francisco and Oklahoma County windows; the 2026 sample also contains some Tulsa windows. No Santa Clara section labels were available at the time.

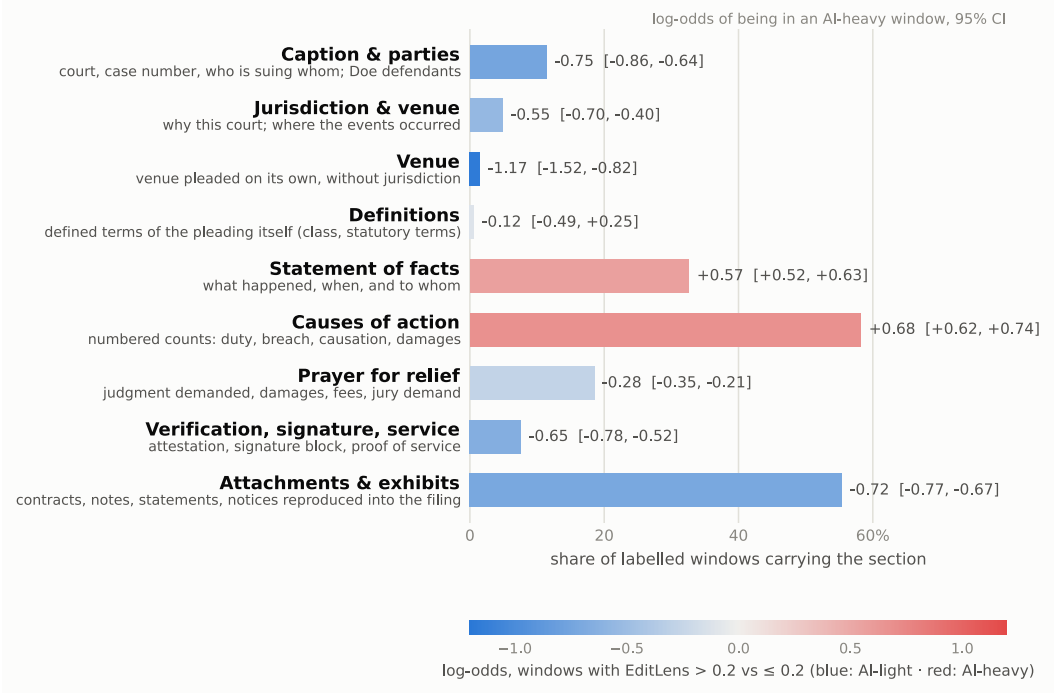


Fig. 15. Supplementary thresholded view of complaint anatomy (Section 5.3). Bar length is the share of windows carrying each section; color is the log-odds of scoring above 0.20 (blue is AI-light, red is AI-heavy). Table 3a instead reports threshold-free probabilities of superiority.