ReLeVAnT: Relevance Lexical Vectors for Accurate Legal Text Classification

Anonymous ACL submission

Abstract

Classifying legal documents by relevance to court filings is a foundational task for downstream applications including motion drafting, docket summarisation, retrieval, litigation surveillance, and training data curation. Existing approaches depend on structured metadata, indicative filenames, or large transformer models. These assumptions break down on noisy, unstructured firm-scale corpora and incur substantial compute cost. The authors propose ReLeVAnT, a lightweight framework for binary legal document classification that models relevance as a discriminative phrase-signal task. ReLeVAnT combines n-gram extraction, contrastive score matching, corpus-specific weighting, and a gradient-boosted tree classifier (XGBoost) into a one-time, interpretable keyword extraction stage followed by a shallow classifier. On the TREC Legal Track, ReLeVAnT achieves 92.7% accuracy and 87.0% Precision; on LexGLUE, 98.9% accuracy and 98.8% Precision. ReLeVAnT runs end-to-end on CPU, classifying documents over 100× faster at inference than competitive embedding baselines while matching or exceeding fine-tuned transformer Precision.

1 Introduction

Binary classification of legal documents by relevance underpins a wide range of downstream tasks, including legal workflow automation (Grossman and Cormack, 2010), docket summarisation (Bhattacharya et al., 2021; Saravanan et al., 2008), retrieval systems (Chalkidis et al., 2022; Pipitone and Alami, 2024), litigation surveillance (Ashley, 2017; Katz et al., 2017), and training data curation for larger ML systems (Chalkidis et al., 2019; Zhong et al., 2020). Within the scope of this work, relevance is defined as relatedness to court filings and proceedings. The authors approach the problem from the perspective of robustness, coverage, and computational efficiency, motivated by the practical constraints of deploying classification at a firm scale.

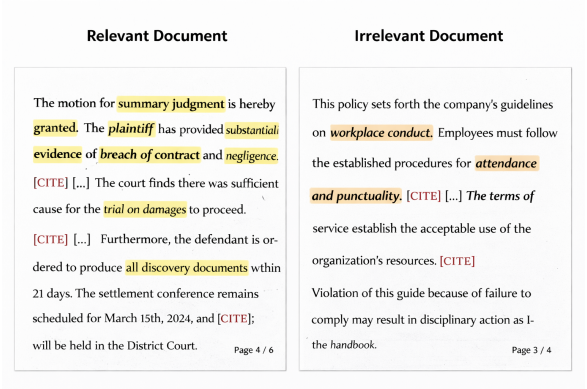

Figure 1: An example of the keywords found in excerpts of relevant and irrelevant documents. The '[CITE]' placeholder is left behind during clause filtering. The relevant document contains stronger and more frequent signals of relevance than the irrelevant document.

Existing methods are constrained by document type, compute cost, or metadata assumptions. Undavia et al. (2018) achieves 72.4% accuracy on SCOTUS classification but requires text in opinion format, limiting use beyond appellate cases. Li et al. (2025) obtain strong results at high speed using filename signals, but break on noisy or out-of-scope filenames common in real corpora. Limsopatham (2021) report up to 20.6% accuracy gains on long legal documents using transformers, at compute costs infeasible at firm scale. Methods such as de Queiroz Santos Filho et al. (2025), Wang et al. (2022), and Watson et al. (2023) report strong results but require curated metadata or structured judgments, which are neither reliably available even at major law firms.

To counter these limitations, the authors model classification as a discriminative task driven by phrase-level signals, building on the observation that relevance in legal corpora is often determined by the presence of highly indicative phrases (Ash-

ley, 2017) (Fig. 1) rather than broad document-level similarity. ReLeVAnT explicitly contrasts phrase frequencies between relevant and irrelevant documents, captures lexical markers that distinguish court filings from non-filings, and feeds these into a gradient-boosted classifier, thereby avoiding metadata dependence, structural assumptions, and expensive representation learning.

The contributions of this work are as follows:

- A formulation of relevance classification as a contrastive phrase-signal task, combining document-level and corpus-level frequency through a contrastive scoring function that admits interpretable keyword extraction.
- A lightweight classifier for legal document relevance that depends solely on document text, free of metadata, filename, or structural assumptions, making it deployable on noisy firm-scale corpora.
- A comprehensive evaluation across 11 baselines spanning classical, embedding-based, fine-tuned transformer, and recent LLM (GPT-5.4-Nano) methods, mapping the accuracy–cost frontier for binary legal document classification.
- State-of-the-art Precision (98.8% on LexGLUE, 87.0% on TREC Legal Track) at 1.06ms per-document inference on CPU, about 100-800× faster than embedding and LLM baselines, matching or exceeding fine-tuned transformers.

2 Related Works

2.1 Classical Text Classification & Retrieval Methods

The heuristics central to ReLeVAnT are grounded in proven foundations in the Language Modelling space. Term-frequency signals for document analysis were popularised by (Sparck Jones, 1972), and the idea of combining term frequency with term rarity is central to ReLeVAnT. However, classical TF-IDF weighs terms globally rather than contrasting classes, and does not explicitly model diminishing returns for repeated occurrences, an important consideration in long documents.

BM25 (Robertson and Zaragoza, 2009) similarly leverages term-frequency importance and lexical matching, but its relevance is determined by an external query, whereas ReLeVAnT implicitly models class-dependent relevance without one. Like SVMs (Cortes and Vapnik, 1995), ReLeVAnT assumes linear separability, but instead of learning weights implicitly, it pre-selects discriminative features (keywords, see Section 3), reducing noise and preserving interpretability. Contrastive learning, whether across modalities (Radford et al., 2021) or via noise contrast (Matsuda et al., 2021), is typically used to learn semantically meaningful representations; ReLeVAnT instead uses contrastive scoring as a lightweight heuristic to isolate discriminative phrases. To our knowledge, no publicly available method implements explicit discriminative filtering to align lexical triggers with legal relevance.

2.2 Legal Relevance and Document Filtering

Identifying relevant documents within large legal corpora has been studied extensively in Technology-Assisted Review (TAR) and the TREC Legal Track (Cormack et al., 2010; Grossman et al., 2011), which formalise relevance as responsiveness to a given matter and rank documents accordingly. These frameworks are inherently query-driven, requiring a predefined topic description to guide relevance estimation. The authors repurpose the TREC dataset for the task by labelling certain sets as relevant and irrelevant (Section 4). Continuous Active Learning (Cormack and Grossman, 2015) extends this paradigm with human-in-the-loop refinement, achieving high recall but relying on iterative labelling, seed queries, and interactive workflows that are operationally complex. In contrast, ReLeVAnT formulates legal relevance as a fixed classification problem, independent of external queries or iterative feedback, making it better suited to large, noisy, unstructured corpora where metadata or human supervision may be unavailable.

2.3 Metadata and Structure-based Methods

Several top-performing methods in legal classification rely on metadata, structured datasets, or curated features. Filename signals combined with TF-IDF and lightweight models (Li et al., 2025) offer a cheap solution similar in spirit to ReLeVAnT but limit generalisability across corpora. The approach of Sebastiani (2002) assumes clean metadata on authors, keywords, and tags, which is often inconsistent or missing in legal data (Ismaylovna, 2024). Domain-specific work in animal protection

(Watson et al., 2023) pioneered keyword-based relevance signals but depends on structured judgments and header cues unavailable in many court filings, while graph-based methods such as (Wang et al., 2022) require more document structure than real-world data typically provides and are sensitive to extraction errors. ReLeVAnT relies solely on document text, generalises beyond narrow domains, and uses NER filtering only to remove non-legal entities.

2.4 Deep Learning-based Methods

Deep learning has found application in legal text classification, but a common drawback is the computational and data costs. These factors are largely independent of the proposed method. Transformer-based approaches (Limsopatham, 2021) model contextual and long-range dependencies, neither of which is strictly necessary for relevance estimation, and incur substantial scalability overhead. SCOTUS classification using Word2Vec with autoregressive architectures Undavia et al. (2018) captures token-level patterns but requires large labelled datasets, optimises a global semantic representation rather than relevance-specific signals, and is restricted to structured court opinions. The shallow n-gram model of Joulin et al. (2017) is closer to ReLeVAnT in spirit but operates on character-level n-grams, which carry less significance than word-level n-grams from a legal-relevance perspective (e.g., *complaint* & *compliance*, *dismiss* & *dismissal*). The use of discriminatory word-level n-grams is a novel feature of the proposed method.

3 Methodology

This section details the proposed methodology of ReLeVAnT. Building on the idea that discriminatory words are strong signals of relevance, the authors propose a 2-stage pipeline comprising the Keyword Extraction pipeline and the XGBoost-based classification pipeline (CLS).

As shown in Fig. 2, the Keyword Extraction pipeline (KE) begins with NER-based filtering (Nadeau and Sekine, 2007). This removes all person names, but preserves tags of locations, clauses and organisations. This is particularly important in longer documents where entity names, such as names of plaintiffs, defendants, petitioners, respondents, appellants, appellees, occur very often, and can affect the keywords due to the extraction being frequency-based, as explained further. The remaining clauses and constitutional references are removed to filter out any residual entities left by NER. Then, n-grams are constructed from these filtered texts. These n-grams are key for representing relevance since the signal "Motion to Dismiss" is stronger than individual words like "Motion" or "Dismiss". With these extracted n-grams, contrastive scores are computed. This can be formalised as:

$$\text{CSM}(t) = \frac{f_t^+}{f_t^+ + (f_t^-)^p + \epsilon} \quad (1)$$

where t is the candidate term (extracted n-gram), f_t^+ is the total frequency of t across all relevant documents, f_t^- is the total frequency of t across all irrelevant documents, ϵ is the smoothing constant and p is the penalty exponent applied to negative frequency to control the aggressiveness of punishing terms also in irrelevant docs. Finally, CSM(t) is the contrastive score for t in the range [0, 1], where 1 is highly specific to relevant documents, and 0 is common in irrelevant documents. Using these scores, the document-level frequency is computed. This is formalised as:

$$\text{DF}(t) = \frac{r_t^+}{r_t^+ + r_t^- + \epsilon} \quad (2)$$

where $r_t^+ = d_t^+ \,/\, N^+$ represents the fraction of relevant documents containing the term t, $r_t^- = d_t^- \,/\, N^-$ is the fraction of irrelevant documents containing the term t, d_t^+ and d_t^- are the number of positive and negative documents containing t, and N^+ and N^- are the total number of positive and negative documents in the corpus. Moreover, a hard filter is implemented to reject terms that appear in any irrelevant document. DF(t) counts the fraction of relevant vs irrelevant documents containing the term t. This captures terms that might appear many times in a few irrelevant documents (inflating the negative frequency), but are actually spread across many relevant documents. Here, it should be noted that CSM(t) does not indicate absence, but rather a lack of lexical signal related to legal relevance in that case.

The combined score is computed by averaging the term-level CSM scores and document-level DF scores.

$$\text{S}(t) = w \cdot CSM(t) + (1 - w) \cdot DF(t) \quad (3)$$

where w is the weighing hyperparameter discussed in the next section. This combined how 'often' and

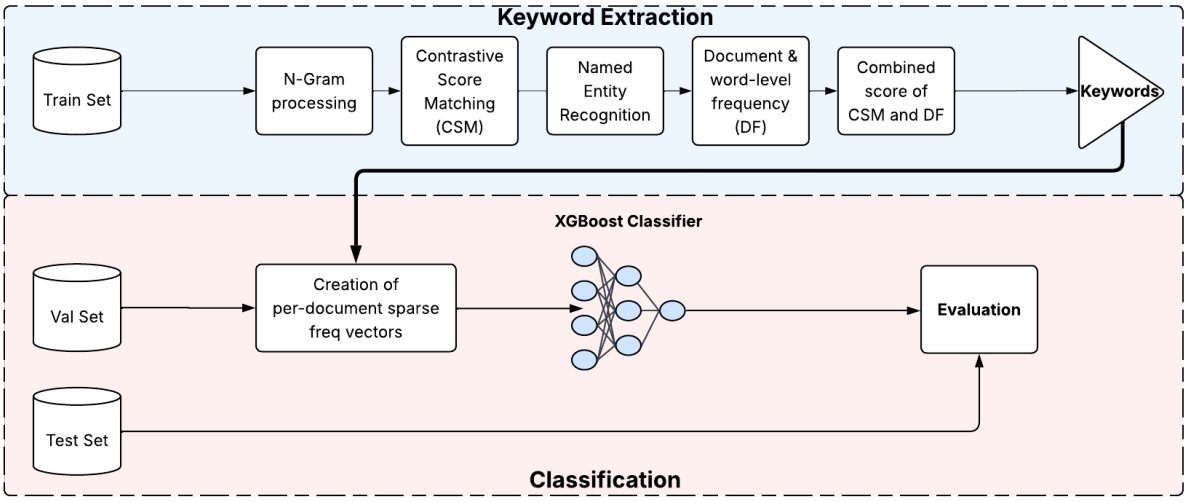

Figure 2: Illustration of the proposed method. The section in Blue highlights the KE stage, and the section in Pink highlights the CLS stage. The given classifier is only for visual purposes, and its exact architecture is detailed in Section 4.

how 'widely' a term t appears in the corpus. This holistic score determines the absolute importance of t across the entire corpus.

This combined score is weighted by the frequency of the specific keyword to determine the value of the respective index in the resulting vector. This is expressed as:

$$\vec{x_j} = S(t_j) \cdot count(t_j, doc) \tag{4}$$

These scores are used to construct a (sparse) vector per document, which is then used by XGBoost to classify documents as relevant or irrelevant. For a given document d and a set of K keywords $t_1, t_2, ..., t_K$, the feature vector is formally defined as:

$$\mathbf{v}(d) = \begin{bmatrix} x_1(d) \\ x_2(d) \\ \vdots \\ x_K(d) \end{bmatrix} \tag{5}$$

where each dimension is the score-weighted frequency $x_j(d) = S(t_j) \cdot count(t_j, doc)$.

These feature vectors are constructed for each document, and the documents are classified by an XGBoost gradient-boosted tree ensemble with a logistic objective. The construction of these feature vectors using a well-weighted, relevance-focused discriminative frequency paradigm ensures accurate and reliable classification.

4 Experiments

To validate the proposed method, the LexGLUE (Chalkidis et al., 2022) benchmark and TREC Legal Track (Grossman et al., 2011) dataset were used. For TREC, Topic 401 (Online Financial Trading) and Topic 402 (Legality of Derivatives Trading) were included, while Topic 403 (Environmental impact of company activities) was excluded. Topic 403 has substantially lower corpus coverage (1300 judged documents vs 1,444–1,559 for Topics 401 and 402), and with only 44 positive test documents, all methods (including ReLeVAnT, TF-IDF, and fine-tuned LegalBERT) produce 0% relevant-class Precision. It is also thematically distinct from the financial-trading focus of the Enron corpus, making it an outlier. The dataset was split into train, val, and test sets of 2101, 151, and 434 documents (2746, 414, and 1122 pages) respectively, with a relevant-irrelevant ratio of approximately 26.8%-73.2% across all splits. For LexGLUE, ECtHR A and SCOTUS were labelled as relevant (European court cases and US Supreme Court opinions) while EUR-LEX and UNFAIR-ToS were labelled as irrelevant (European legislation and Terms of Service clauses). CaseHOLD and LEDGAR were excluded as they are QA-task-centric and contract-provision-centric, respectively. The dataset was split into 29532, 9675, and 9007 documents (201666, 74538, and 85441 pages) for train, val, and test, with relevant-irrelevant ratios of 47%-53%, 25%-75%, and 27%-73%.

These ratios reflect realistic scenarios where law

Method	LexGLUE		TREC	
	Acc↑	Pre↑	Acc↑	Pre↑
Always positive	26.6	26.6	26.8	26.8
Manual keywords	79.9	64.7	81.7	55.2
Chi^2+LR	99.8	99.8	90.0	68.4
BM25	41.3	79.5	77.1	45.3
MiniLM+LR	99.4	99.4	85.4	61.9
TF-IDF+LR	99.9	100.0	90.8	92.3
DistilBERT	73.4	0.0	25.8	24.4
LegalBERT	31.5	11.7	37.8	56.6
DistilBERT*	100.0	100.0	90.8	80.0
LegalBERT*	100	100.0	93.2	83.2
GPT-5.4-Nano	98.5	98.5	80.5	51.0
Longformer	–	–	92.8	80.9
ReLeVAnT	**98.9**	**98.8**	**92.7**	**87.0**

Table 1: Comparison of methods on LexGLUE and TREC. Accuracy and weighted Precision are reported for both datasets. The '*' denotes results with fine-tuning.

Method	LexGLUE setup (s)↓	TREC setup (s)↓	Inf. (/doc)↓
TF-IDF+LR	34.8	1.2	<1ms
Chi^2+LR	37.3	1.2	<1ms
BM25	81.8	25	4.6ms
MiniLM+LR	474	44.7	150ms
DistilBERT*	1445	965	2.2ms
LegalBERT*	6420	1440	5.7ms
GPT-5.4-Nano	–	–	870ms
Longformer	–	5220	148.7ms
ReLeVAnT	109.2	369.7	1.06ms

Table 2: Wall-clock setup and inference cost across methods on LexGLUE and TREC. Setup denotes one-time training (or KE+CLS for ReLeVAnT) per corpus; inference is per-document classification cost after setup. The '*' denotes fine-tuned transformer baselines, which require GPU access. ReLeVAnT is the only competitive method that runs end-to-end on CPU. GPT-5.4-Nano has no setup time since inference is through an API call.

firms hold large volumes of irrelevant data (discovery documents, bills, news reports). ReLeVAnT is invariant to class imbalance and performs well in both regimes, as shown in Table 3. Performance is evaluated using weighted Accuracy and weighted Precision, and all experiments are conducted on an Intel Core Ultra 9 H-class CPU. As described in Section 3, eyecite (Cushman et al., 2021) is used to remove clauses and references, the SpaCy core-small model is used for entity filtering, and CSM uses a smoothing constant of 0.01.

All baselines, including ReLeVAnT, are compared in Table 2, which demonstrates ReLeVAnT's feasibility. Each design choice is ablated thoroughly. Excluding lemmatisation and stemming improves performance on both datasets as seen in Table 4.

The n-gram range sweep (Table 5) selects 4-grams for LexGLUE and 5-grams for TREC as optimal.

Class Split	Accuracy↑	Pre↑
Realistic	99.3	98.7
Original	98.9	98.1

Table 3: ReLeVAnT's invariance to class imbalance seen in consistent results between less and more aggressive class imbalance scenarios on LexGLUE. The 'Realistic' scenario has a relevant-irrelevant class ratio of 47%-53%, whereas the 'Original' ratio in the dataset is 19%-81%.

Method	LexGLUE		TREC	
	Acc↑	Pre↑	Acc↑	Pre↑
ReLeVAnT+LS	95.8	92.1	85.1	70.3
ReLeVAnT	**98.4**	**97**	**87.0**	**73.1**

Table 4: Comparison of ReLeVAnT with and without lemmatisation and stemming on LexGLUE and TREC. The $+LS$ suffix indicates ReLeVAnT with lemmatisation and stemming at KE and CLS.

The Minimum Term Frequency (MTF) sweep (Table 6) yields optimal thresholds of 30 for LexGLUE and 10 for TREC, reflecting the differing scales of the two datasets.

The penalty exponent ((1)) is central to scoring and is swept in Table 7, while the document frequency weight (Table 8) balances document-level and cross-document presence.

XGBoost tree depth is swept in Table 9, with depth 4 selected.

The final XGBoost classifier uses 400 estimators, max tree depth of 4, learning rate 0.05, subsample ratio 0.8, and column subsample ratio 0.8 per tree, trained with logloss as the evaluation metric and a per-topic decision threshold tuned on a held-out validation set.

To validate that performance is not driven by a few dominant features or underlying bias, three additional configurations were tested (Table 10): using only the top 3 keywords by frequency (LexGLUE: 'remanded', 'testimony', 'congres-

N-gram	LexGLUE		TREC	
	Acc↑	Pre↑	Acc↑	Pre↑
2-gram	97.9	96.0	86.5	76.5
3-gram	98.0	96.3	85.7	78.5
4-gram	**98.1**	**96.4**	87.1	86.5
5-gram	98.1	96.4	**87.7**	**88.9**
6-gram	98.1	96.4	86.4	87.5
7-gram	98.1	96.3	68.1	68.2
15-gram	98.1	96.4	86.6	88.0

Table 5: Comparison of ReLeVAnT across n-gram ranges on LexGLUE and TREC.

MTF	LexGLUE		TREC	
	Acc↑	Pre↑	Acc↑	Pre↑
10	98.4	97.1	**87.3**	**84.0**
20	98.4	97.0	85.0	79.8
30	**98.4**	**97.0**	84.8	70.1
50	98.1	96.4	84.0	78.3
100	97.4	95.0	82.6	81.5
250	97.0	94.2	80.5	71.1
400	96.3	92.7	74.2	36.8
1000	93.4	86.5	73.2	0.0

Table 6: Comparison of ReLeVAnT across MTF thresholds on LexGLUE and TREC.

PenEx	LexGLUE		TREC	
	Acc↑	Pre↑	Acc↑	Pre↑
1	98.4	97.0	**87.7**	**83.3**
2	98.4	97.0	87.4	83.1
10	**98.6**	**97.3**	84.2	82.7
50	98.5	97.1	83.9	83.3
∞	98.1	96.4	83.4	73.9

Table 7: Comparison of ReLeVAnT across penalty exponent (PenEx) thresholds on LexGLUE and TREC.

DocFreq	LexGLUE		TREC	
	Acc↑	Pre↑	Acc↑	Pre↑
0.25	98.1	96.4	87.5	78.6
0.33	98.9	97.9	87.7	85.4
0.50	99.2	98.5	87.7	83.3
0.66	99.3	98.6	**88.4**	**86.7**
0.75	**99.3**	**98.7**	87.9	85.3
0.90	98.9	98.0	87.7	79.6

Table 8: Comparison of ReLeVAnT across document frequency thresholds on LexGLUE and TREC.

Max Depth	LexGLUE		TREC	
	Acc↑	Pre↑	Acc↑	wPre↑
2	98.6	98.6	92.2	84.7
3	98.7	98.7	92.6	86.8
4	**98.9**	**98.8**	**92.7**	**87.0**
6	98.8	98.8	92.6	86.1
9	98.5	98.5	92.2	86.0
12	98.4	98.4	92.1	85.0

Table 9: Comparison of ReLeVAnT (XGBoost) across tree depths on LexGLUE and TREC. Max depth of 4 achieves optimal precision on both datasets, with deeper trees yielding diminishing returns on the small TREC corpus.

sional'; TREC: 'CME', 'CBOE', 'Mini'), the full keyword set excluding those top 3 (ReLeVAnT-K), and random feature vectors. The full ReLeVAnT pipeline outperforms all three, confirming that the classifier draws signal from the broader discriminative keyword set rather than memorising a handful of cues.

5 Results

Experimental results in Table 1 reveal the strong performance of ReLeVAnT relative to baselines such as majority-class prediction, always-positive selection, manual keyword selection, TF-IDF with Logistic Regression, Chi-sq with Logistic Regression, BM25 (Robertson and Zaragoza, 2009), MiniLM embeddings (Wang et al., 2020) with Logistic Regression, DistilBERT (Sanh et al., 2019), LegalBERT (Chalkidis et al., 2020) and GPT-5.4-Nano (OpenAI, 2026). 'Always positive' refers to a dummy classifier that classifies everything as relevant; this is the anchor of the results, where any value less than this would be considered erroneous. Manually-chosen keywords with an MLP demonstrate subpar classification at <85% accuracy on both datasets (Appendix B). On TREC, ReLeVAnT achieves higher accuracy than TF-IDF (92.7% vs 90.8%), but TF-IDF achieves marginally higher Precision (92.3% vs 87.0%); this is attributed to TF-IDF's conservative classification behaviour, which yields high Precision at the cost of substantially lower coverage of the relevant class, a tradeoff poorly suited to the legal setting where missed filings carry real cost. Similarly, Chi^2+LR matches TF-IDF on accuracy but trails ReLeVAnT by 18.6% in Precision on TREC, reinforcing that statistical feature selection alone is insufficient when class-discriminative phrases are sparse. BM25, designed for retrieval rather than classification, underperforms across both datasets (41.3% accuracy on LexGLUE, 77.1% on TREC), reinforcing the limitation of query-driven scoring in the absence of a query. MiniLM with Logistic Regression performs strongly on LexGLUE (99.4% accuracy) but drops to 85.4% on TREC with only 61.9% Precision, showing that semantic embeddings struggle when the discriminative signal is lexical and topic-specific rather than stylistic. Moreover, Zero-shot transformer inference fails because pretrained objectives, such as masked language modelling, confer no knowledge of task-specific discriminative

Vector Config	LexGLUE		TREC	
	Acc↑	Pre↑	Acc↑	wPre↑
Top 3 keywords	87.9	70.8	78.7	70.4
Random vectors	73.4	0	28.9	21.5
ReLeVAnT-K	98.4	97.0	87.8	75.3
ReLeVAnT	**98.9**	**98.8**	**92.7**	**87.0**

Table 10: Comparison of ReLeVAnT across specific vector configurations on LexGLUE and TREC. ReLeVAnT-K refers to the entire keyword list except the top 3 keywords.

features. Fine-tuning addresses this by providing a supervised signal that adjusts representations toward class-separating directions in the embedding space; on TREC, fine-tuned DistilBERT trails ReLeVAnT on both metrics (90.8% vs 92.7% accuracy and 80.0% vs 87.0% Precision), while fine-tuned LegalBERT marginally surpasses ReLeVAnT on accuracy (93.2% vs 92.7%) but still trails on Precision (83.2% vs 87.0%). ReLeVAnT achieves a similar goal by explicitly contrasting frequencies upstream, allowing a lightweight gradient-boosted classifier to operate on already-discriminative features rather than learning them end-to-end from raw text. To evaluate whether a modern lightweight LLM can match contrastive lexical signals out-of-the-box, the authors include OpenAI's GPT-5.4-Nano (OpenAI, 2026) as a zero-shot baseline. GPT-5.4-Nano is the smallest, lowest-latency variant of the GPT-5.4 family, marketed by OpenAI for short-turn classification, data extraction, and ranking tasks, making it a natural fit for binary relevance prediction. It performs well on LexGLUE because the binary framing reduces to genre identification (court opinions vs legislation/ToS), a distinction salient from pretraining alone. It fails on TREC because relevance there requires distinguishing documents within a single corpus that share genre, register, and vocabulary; a discrimination that depends on corpus-specific lexical contrasts the model has no access to in a zero-shot setting.

Beyond accuracy and Precision, Table 2 highlights the practical cost advantages of ReLeVAnT. At inference, ReLeVAnT classifies a document in 1.06ms on CPU, roughly 141× faster than MiniLM+LR (150ms), 5.4× faster than fine-tuned LegalBERT (5.7ms on GPU), and 820× faster than GPT-5.4-Nano (870ms), while operating on commodity hardware rather than requiring sustained GPU access. Setup costs reveal an even sharper contrast: on LexGLUE, ReLeVAnT completes setup in 109.2s on CPU, 4.3× faster than MiniLM+LR (474s), 13.2× faster than DistilBERT fine-tuning (1445s on GPU), and 58.8× faster than LegalBERT fine-tuning (6420s on GPU). On TREC, ReLeVAnT (369.7s on CPU) completes 2.6× faster than DistilBERT (965s on GPU) and 3.9× faster than LegalBERT (1440s on GPU). Across both datasets, the cost asymmetry is compounded by the hardware difference: ReLeVAnT runs end-to-end on commodity CPU while the transformer baselines require sustained GPU access. ReLeVAnT does incur a higher setup cost than the lightest baselines, TF-IDF+LR and Chi-sq+LR, which finish setup in under 40s on LexGLUE and ∼1s on TREC, but those baselines pay this back in classification quality, with TF-IDF lagging ReLeVAnT by 1.9% on TREC accuracy and Chi-sq+LR by 18.6% on TREC Precision. ReLeVAnT thus occupies a favourable position on the accuracy–cost frontier: it matches or exceeds fine-tuned transformer Precision on both datasets while running at one to two orders of magnitude lower compute cost, and it materially outperforms the cheapest classical baselines without straying far from their cost regime.

ReLeVAnT is robust to class imbalance on LexGLUE (Table 3), with a 0.4% accuracy and 0.6% Precision difference between regimes, owing to the near independence of negative frequency at CSM (Eq. 1). Tables 5–9 show single-hyperparameter sweeps; the final configuration values appear in Table **??**.

To explore the impact of vocabulary scope, the first set of experiments focused on lemmatisation and stemming. As evident in Table 4, ReLeVAnT without this processing step performs considerably better on both datasets, by upto 2.4% in accuracy and 4.3% in Precision on LexGLUE and 1.1% in accuracy and 2.8% in Precision on TREC. Legal language is precise, and morphological variants carry distinct procedural or substantive meaning that lemmatisation and stemming erode, reducing the contrastive signal CSM relies on.

Table 5 justifies the optimal choice of n-grams for each dataset. For LexGLUE, performance improves up to 4-grams, then stagnates, while the keyword count remains stable. TREC demonstrates optimal results at 5-grams, after which performance steadily declines. The disagreement over the optimal n-gram choice between the datasets stems from corpus size: TREC has fewer documents than LexGLUE, leading to longer n-grams in TREC that are less sparse, allowing vectors to capture just enough semantic information for reliable classification.

Table 6 demonstrates optimal performance at MTF 30 for LexGLUE and MTF 10 for TREC. The MTF threshold resolves a tension inherent to the CSM: low-frequency terms can produce extreme contrast scores, but those scores may reflect either genuine but rare topic indicators or statistical coincidences from a single repeated mention. Setting MTF too low admits the latter as features, while setting it too high discards the former. The

disagreement between LexGLUE and TREC is explained by the discriminative signal in LexGLUE being stylistic; formal legal documents distinguish themselves through standardised, repeated phraseology that occurs many times across the corpus, so terms that fail to clear MTF 30 are almost certainly noise. In contrast, TREC's discriminative signal is topic-specific terminology within a uniform conversational genre, where genuinely informative terms may appear in only a small fraction of the relevant set, demanding a lower MTF to be retained.

Table 7 points to the choice of the penalty exponent in the CSM modelling, as explained in Eq. 1. A higher penalty exponent imposes stricter punishment on keywords that also occur in negative documents. Tuning this to 10 produces the best results for LexGLUE, while tuning to 1 gives maximal returns for TREC. The disagreement is because LexGLUE's irrelevant documents share a large overlapping vocabulary in legal language, sentence structure, and writing style with the relevant set, hence such terms need to be penalised more. In TREC, which has a smaller corpus, an excess penalty discards relevant but rarer terms, as seen in the decline in performance.

Table 8 showcases the best performance of ReLeVAnT at 0.75 weight for LexGLUE and 0.66 for TREC. This indicates that preference for in-document frequency results in better performance over frequency across documents for both datasets, owing to the size of the dataset and the length of documents. This reveals an interesting finding: local frequency is more influential than global frequency across corpora with longer documents, and vice versa.

Table 9 details the tree depths explored for the XGBoost classifier. A maximum depth of 4 achieves optimal precision on both datasets. Shallower trees (depth 2–3) underfit the feature space, while deeper trees (depth 6–12) progressively overfit on TREC's smaller corpus. LexGLUE is largely insensitive to depth given the corpus's size and stylistic separability.

To investigate dependency on top keywords, the authors isolated the top 3 most influential keywords from each dataset. The exact same setup with just these 3 keywords demonstrates passable performance of 87.9% accuracy and 70.8% Precision on LexGLUE, and 78.7% accuracy and 70.4% Precision on TREC. This supports the initial assumption of discriminatory words being a strong, separable signal of legal relevance: 'remanded', 'testimony', 'congressional' for LexGLUE and 'CME', 'CBOE', 'Mini' for TREC (see Appendix A.6 for glosses), all intuitively strong markers of legal relevance. Removing these 3 to assess generalizability reveals that XGBoost can compensate for the lack of very strong signals and performs very similarly to the full keyword list, with only a 0.9% drop in accuracy and 1.7% drop in Precision on LexGLUE, and 0.4% drop in accuracy and 1.4% drop in Precision on TREC. To assess the classifier's randomness, random feature vectors were generated for each document, yielding an exact class-split accuracy of 73.9% and a Precision of 0. These results further reinforce the hypothesis about legal relevance being measurable through the frequency of discriminatory signals.

These ablations isolate three findings. First, the contrastive keyword selection carries most of the signal: three keywords alone recover 87.9% accuracy on LexGLUE, while random vectors collapse to 0% Precision. Second, the classifier's role is to integrate redundant signals across the keyword set, which is why removing the top three keywords results in only a 0.9% drop in accuracy. Third, the optimal hyperparameter configuration tracks the discriminative signal in each corpus: LexGLUE's stylistic signal rewards higher MTF thresholds, stronger penalty exponents, and shorter n-grams, while TREC's topic-specific signal rewards the opposite. Across both corpora, in-document frequency dominates cross-document frequency, suggesting local repetition outweighs corpus-wide spread when documents are long. Combined with the cost-frontier results, these findings support the broader claim that explicit upstream feature engineering remains competitive with end-to-end transformer learning when the underlying signal is lexical rather than semantic.

6 Conclusion and Future Works

This work introduced ReLeVAnT, a lightweight, contrastive, phrase-driven framework that leverages a shallow classifier to achieve state-of-the-art results on binary classification tasks for court filings. It is completely independent of metadata, filenames, document and directory structure, allowing for massive coverage and robustness at an exponentially lower cost than other methods in the legal text field.

7 Limitations

The authors outline several limitations of ReLeVAnT and its evaluation, and encourage future work to address them.

Performance on small, imbalanced corpora. ReLeVAnT, along with the baselines, performs poorly on TREC Topic 403 (Environmental impact of company activities), where the relevant class comprises only 44 documents (10.6%) of the test set. As discussed in Section 4, this topic produced near-zero relevant-class Precision across nearly all hyperparameter configurations, motivating its exclusion from the main evaluation. This points to a real failure mode of the method: when the relevant pool is too small to populate the contrastive frequency table reliably, CSM scores become dominated by noise rather than signal. Methods designed for low-resource settings (few-shot fine-tuning, active learning) may be more appropriate in this regime. This problem is generally not encountered with top-level law firms.

Binary classification only. ReLeVAnT is formulated as binary relevance classification (relevant vs irrelevant). Multi-class and multi-label legal classification can be applied similarly. For example, identifying the specific filing type, jurisdiction, or legal issue is not addressed. The contrastive scoring formulation in Eq. 1 extends naturally to one-vs-rest schemes, but the trade-offs of doing so (keyword overlap across classes, threshold calibration, computational cost scaling with class count) have not been studied here.

Multi-seed results. All reported numbers reflect training runs with results averaged across 10 seeds. The authors do not report variance across seeds since it is ≤ 0.1 for each result.

Transformer baselines under input-length constraints. DistilBERT and LegalBERT were fine-tuned with documents truncated to 512 tokens, following standard practice. Legal documents in both datasets frequently exceed this length, which limits the context available to the transformer baselines. Methods that handle long documents (e.g., Longformer, hierarchical BERT) exist but introduce additional architectural and computational overhead beyond the scope of this comparison; ReLeVAnT processes documents at full length without truncation. This means the gap between ReLeVAnT and the transformer baselines partly reflects truncation rather than purely modelling capacity, and the comparison would need to be revisited against long-context models.

Dataset scope. Both datasets are English-language. TREC corresponds to the Enron corpus with two financial-trading topics, and LexGLUE in the binary framing reduces to coarse domain classification (US/European court opinions vs EU legislation and Terms of Service). Neither tests ReLeVAnT against a corpus where relevant and irrelevant documents share a single domain, language, and style. For example, distinguishing motions from briefs within a single case. Generalisation to such fine-grained, intra-domain relevance distinctions is not established.

Interpretability vs explanation. ReLeVAnT produces an inspectable keyword list, which the authors treat as a form of interpretability. The authors do not provide per-document explanations (e.g., SHAP-style feature attributions for individual classifications), nor have the authors evaluated whether the extracted keywords align with how legal practitioners actually judge relevance. User studies with domain experts would be needed to validate the method's practical interpretability.

References

Kevin D Ashley. 2017. *Artificial intelligence and legal analytics: new tools for law practice in the digital age*. Cambridge University Press.

Paheli Bhattacharya, Soham Poddar, Koustav Rudra, Kripabandhu Ghosh, and Saptarshi Ghosh. 2021. Incorporating domain knowledge for extractive summarization of legal case documents. In *Proceedings of the eighteenth international conference on artificial intelligence and law*, pages 22–31.

Ilias Chalkidis, Emmanouil Fergadiotis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2019. Large-scale multi-label text classification on eu legislation. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 6314–6322.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Katz, and Nikolaos Aletras. 2022. Lexglue: A benchmark

dataset for legal language understanding in english. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4310–4330.

Gordon V. Cormack and Maura R. Grossman. 2015. Autonomy and reliability of continuous active learning for technology-assisted review. *arXiv preprint arXiv:1504.06868*.

Gordon V. Cormack, Maura R. Grossman, Bruce Hedin, and Douglas W. Oard. 2010. Overview of the trec 2010 legal track. In *Proceedings of the Text REtrieval Conference (TREC)*. National Institute of Standards and Technology (NIST).

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

Jack Cushman, Matthew Dahl, and Michael Lissner. 2021. eyecite: A tool for parsing legal citations. *Journal of Open Source Software*, 6(66):3617.

José Jorge de Queiroz Santos Filho, Filipe Araújo Dantas, Melquezedeque da Silva Lima, Shirley Barbosa dos Santos, Galileu Genesis, Maria Gabriely Lima da Salva, Álvaro Farias Pinheiro, and Eraylson Galdino da Silva. 2025. Comparing machine learning and an expert system for legal document classification. In *Conference on Digital Government Research*, volume 26.

Maura R Grossman and Gordon V Cormack. 2010. Technology-assisted review in e-discovery can be more effective and more efficient than exhaustive manual review. *Rich. JL & Tech.*, 17:1.

Maura R. Grossman, Gordon V. Cormack, Bruce Hedin, and Douglas W. Oard. 2011. Overview of the trec 2011 legal track. In *Proceedings of the Text REtrieval Conference (TREC)*. National Institute of Standards and Technology (NIST).

BJ Ismaylovna. 2024. Problems of admissibility and reliability of metadata as evidence. international journal of. *Law and Policy*, 2(8):1.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomáš Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th conference of the European chapter of the association for computational linguistics: volume 2, short papers*, pages 427–431.

Daniel Martin Katz, Michael J Bommarito II, and Josh Blackman. 2017. A general approach for predicting the behavior of the supreme court of the united states. *PloS one*, 12(4):e0174698.

Zhijian Li, Stefan Larson, and Kevin Leach. 2025. Document classification using file names. In *Proceedings of the 2025 ACM Symposium on Document Engineering*, pages 1–10.

Nut Limsopatham. 2021. Effectively leveraging bert for legal document classification. In *Proceedings of the natural legal language processing workshop 2021*, pages 210–216.

Takeru Matsuda, Masatoshi Uehara, and Aapo Hyvarinen. 2021. Information criteria for non-normalized models. *Journal of Machine Learning Research*, 22(158):1–33.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

OpenAI. 2026. Introducing gpt-5.4 mini and nano. https://openai.com/index/introducing-gpt-5-4-mini-and-nano/. Accessed: 2026-05-07.

Nicholas Pipitone and Ghita Houir Alami. 2024. Legalbench-rag: A benchmark for retrieval-augmented generation in the legal domain. *arXiv preprint arXiv:2408.10343*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Stephen Robertson and Hugo Zaragoza. 2009. *The probabilistic relevance framework: BM25 and beyond*, volume 4. Now Publishers Inc.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Murali Saravanan, Balaraman Ravindran, and S Raman. 2008. Automatic identification of rhetorical roles using conditional random fields for legal document summarization. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. In *ACM Computing Surveys*, volume 34, pages 1–47. ACM.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Samir Undavia, Adam Meyers, and John E Ortega. 2018. A comparative study of classifying legal documents with neural networks. In *2018 Federated conference on computer science and information systems (FedCSIS)*, pages 515–522. IEEE.

Qiqi Wang, Kaiqi Zhao, Robert Amor, Benjamin Liu, and Ruofan Wang. 2022. D2gclf: Document-to-graph classifier for legal document classification. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2208–2221.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Proceedings of the 17th International Conference on Machine Learning (ICML)*.

Joe Watson, Guy Aglionby, and Samuel March. 2023. Using machine learning to create a repository of judgments concerning a new practice area: a case study in animal protection law. *Artificial Intelligence and Law*, 31(2):293–324.

Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. How does nlp benefit legal system: A summary of legal artificial intelligence. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 5218–5230.

A Detailed Ablation Analysis

A.1 Lemmatisation and Stemming

The loss of semantic meaning from lemmatisation and stemming is concrete in legal text. For example, 'futures' and 'future' are both lemmatised to 'futur'; however, 'futures' is a domain-specific term in securities trading cases, whereas 'future' could refer to plans or an event in a news report. Similarly, stemming collapses 'proceeding' (a legal noun indicating regulatory action) and 'proceed' (a general verb) into the same form. These collisions remove precisely the morphological distinctions CSM uses to separate relevant from irrelevant documents.

A.2 N-gram Range

An additional experiment with 15-grams was conducted to test for edge cases of very long repeated sentences (e.g., boilerplate clauses or signature blocks). As shown in the final row of Table 5, performance is statistically indistinguishable from the 4-gram and 5-gram configurations, indicating that very long phrases offer no additional signal beyond what mid-length n-grams already capture.

A.3 Minimum Term Frequency

The topic-specific terms that drive TREC classification at MTF 10 include regulator acronyms (e.g., 'CFTC', 'SEC'), trading-instrument names (e.g., 'CME', 'CBOE', 'E-mini'), and Enron-internal project codenames. These appear in only a small fraction of the relevant set but are highly discriminative when present. Raising MTF above 10 progressively discards these terms, explaining the steep accuracy decline on TREC visible in Table 6 from MTF 100 onward.

A.4 Penalty Exponent

The vocabulary overlap that motivates LexGLUE's higher penalty exponent comes from multiple sources: shared legal terminology across court opinions and legislation, common sentence-structural patterns ('the court holds', 'pursuant to'), and overlapping stylistic register across formal legal writing. Without aggressive penalisation, these shared terms produce high CSM scores despite being non-discriminative. TREC's smaller, narrower corpus has less such overlap, and an excess penalty instead discards genuinely relevant but rarer terms

A.5 XGBoost Tree Depth

The underfit-overfit behaviour across depths is visible in Table 9: at depth 2 to 3, the model fires too broadly across keyword dimensions, producing lower precision; at depth 6 to 12, the trees memorise the training noise on TREC's 2101-document training set rather than generalising, causing precision to decrease by 1 to 2%. LexGLUE plateaus between depths 3 and 6, consistent with its larger corpus and stronger stylistic separation between classes. Extended per-ablation analysis with worked examples appears in Appendix A.

A.6 Top-3 Keyword Glosses

The top three keywords by frequency in each corpus relate to legally salient entities and concepts as follows. For LexGLUE: 'remanded' refers to a higher court sending a case back to a lower court for further proceedings, and is characteristic of appellate opinions; 'testimony' refers to evidence given by a witness under oath, central to court records; 'congressional' relates to the United States Congress, frequently cited in SCOTUS opinions. For TREC: 'CME' refers to the Chicago Mercantile Exchange, a major US derivatives marketplace; 'CBOE' refers to the Chicago Board Options Exchange, a major US options market; 'Mini' refers to the E-mini futures contract, a class of electronically-traded derivatives. All six terms are highly diagnostic of the relevant document classes in their respective corpora.

B Manual Keyword Baseline

The following keywords were used with an OR-match rule: a document is classified as relevant if any keyword appears in the text.

applicant, court, judgment, petition, certiorari, affirmed, reversed, held, constitutional, amendment, justice, appeal, verdict, plaintiff, defendant, proceedings, statute, ruling, tribunal, conviction

TREC Legal Track

Topic 401 (EnronOnline / Online Financial Trading Systems):

enron online, enrononline, eon, online trading, online system, trading system, electronic trading, online platform, trading platform

Topic 402 (Legality of OTC Derivatives and Financial Instrument Trading):

derivative, derivatives, otc, over the counter, over-the-counter, financial instrument, swap, swaps, futures contract, options contract, hedge, hedging, notional, credit default

All keywords were selected by a subject matter expert without consulting the training data. Classification uses an OR rule: a document is labelled relevant if any keyword appears verbatim in the document text.