POLITECNICO DI TORINO
Repository ISTITUZIONALE

Exploring In-Context Learning Strategies for Temporal Ordering of Legal Events using Large Language Models.

Original
Exploring In-Context Learning Strategies for Temporal Ordering of Legal Events using Large Language Models / Cacioli, A., Cagliero, L., Tarasconi, F.. - 4192:(2026). (EDBT/ICDT-WS 2026 EDBT/ICDT 2026 Workshops Tampere (Finland) March 24, 2026.).

Availability:
This version is available at: 11583/3015321 since: 2026-09-09T16:30:41Z

Publisher:
CEUR

Published
DOI:

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

18 November 2026

Exploring In-Context Learning Strategies for Temporal Ordering of Legal Events using Large Language Models

Andrea Cacioli[1,2,*], Luca Cagliero[1,*] and Francesco Tarasconi[2]

[1]Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Turin, Italy
[2]Aruba AI Srl, Corso Francia 2/bis, 10143 Turin, Italy

Abstract

Large Language Models (LLMs) are increasingly adopted for legal document understanding by attorneys and legal consultants. Despite advances in adapting LLMs to their legal terminology and domain-specific linguistic nuances, the LLMs' ability to reason about temporal relations in legal documents remains largely underexplored. In this work, we explore the capabilities of LLMs to verify the correctness of a legal temporal ordering clause and to classify the type of temporal relationships between two legal entities. The results achieved on a public English-written benchmark show that (1) instruction-based models generally perform better than the corresponding chat versions; (2) LLMs reasoning capabilities are, typically, marginally useful to address the specific temporal reasoning tasks; (3) LLMs under a Few-Shot Learning (FSL) setting turn out to be the most effective, with Grok 4 surpassing the state of the art.

Keywords

Legal AI, Temporal Reasoning, Large Language Models

1. Introduction

Large Language Models (LLMs) have demonstrated remarkable legal document understanding and generation capabilities [1]. Within the legal domain, the most established tasks encompass (1) content search [2, 3], (2) document review [4, 5], and (3) prediction [6, 7]. The latter category of tasks also includes the deep understanding of complex semantic relations in text, such as legal entailment types, rhetorical roles, and temporal relations.

Reconstructing temporal relationships is known to be particularly challenging for LLMs [8]. Specifically, previous studies have shown that most LLMs fall short when they are asked to either update a knowledge base or adapt their responses to time-evolving scenarios [9].

So far, limited research efforts have been devoted to addressing temporal reasoning in the legal domain. For example, in LexTime [10] the authors address a prediction task which entails predicting whether a temporal ordering relationship between a pair of events mentioned in the document text (e.g., *event A precedes event B*) is true or false.

The main limitations of state-of-the-art works on temporal reasoning for legal document understanding are enumerated below.

- **Lack of Deep Reasoning**: They analyze classical textual LLMs belonging to the LlaMA [11], GPT [12], and Mistral [13] families while ignoring the LLMs that have been specifically pretrained with deep reasoning capabilities.
- **Binary Verification**: They analyze the zero-shot and few-shot LLM capabilities to verify whether a given statement is correct or not [10], leaving open more challenging legal understanding tasks, such as the automatic detection of the type of event ordering.
- **Limited exploration of the models' efficiency**: They do not deepen into the analysis of relevant technical aspects, such as context length, and model inference costs.

Published in the Proceedings of the Workshops of the EDBT/ICDT 2026 Joint Conference (March 24-27, 2026), Tampere, Finland
*Corresponding author.
luca.cagliero@polito.it (L. Cagliero); francesco.tarasconi@staff.aruba.it (F. Tarasconi)
0000-0002-7185-5247 (L. Cagliero); 0000-0003-4562-2463 (F. Tarasconi)
© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-04-09

CEUR-WS.org/Vol-4192/DARLIAP-paper8.pdf

This paper addresses the above-mentioned issues. Specifically, it not only studies the LLM capabilities to verify the correctness of a legal temporal ordering clause, but also classifies the type of temporal relationships between two legal entities. It also empirically compares chat- and instruct-based LLMs, LLMs with deep reasoning and not, and models with different sizes, context lengths, and inference costs.

The results achieved by Grok 4[1] under a few-shot learning setting surpasses the state of the art on the binary verification task (accuracy: Grok 4 85.3% vs. GPT 4 80.8%) and achieves robust performance on the multi-class event ordering classification task. Notably, the LLMs with deep reasoning capabilities achieve just marginal improvements or no improvements, likely because they incorporate a limited background in the legal domain.

The remainder of this paper is organized as follows. Section 2 formalizes the established and new temporal reasoning task. Section 3 presents the proposed methodology, while Section 4 summarizes the main experiments. Finally, Section **??** draws conclusions and discusses the future research developments.

2. Problem statement

Given a legal document d, we extract a context paragraph p in d mentioning a sequence of two legal events $\langle a, b \rangle$. Events a and b are either one implicit and one explicit event or two explicit events [10]. In compliance with [10], every event is defined by an occurrence or action triggered by a verb or noun taking place at a specific moment.

In the following we define the tasks addresses in this work.

Legal Event Temporal Ordering Verification Given an ordered sequence $\langle a, b \rangle$ consisting of events a and b and an arbitrary temporal relationship r, this task, hereafter denoted by LETOV for the sake of brevity, aims to verify whether the statement $a\ tr\ b$ (e.g. *event a precedes event b*) holds (target response: *yes*) or not (target response: *no*).

Legal Event Temporal Ordering Classfication Given an ordered sequence of events $\langle a, b \rangle$, and a predefined set of temporal relationships $\{tr_1, tr_2, tr_{ldots}, tr_n\}$ (e.g., *precedes*, *subsequent*, *contemporary*), this task, hereafter denoted by LETOC for the sake of brevity, has the goal of predicting the correct temporal relationship between events a and b.

With the goal of deepening the analysis of the LLMs' capabilities in legal temporal temporal reasoning, we introduce LETOC as a new task extending LETOV [10].

3. Methodology

To assess the LLMs' capabilities to address LETOV and LETOC we apply the following steps. Firstly, we enrich the statements originally included in [10] with different prompting styles, including chat- and instruct versions as well as zero-shot and few-shot learning settings. Based on the results observed in the preliminary experiments (see Section 4 for further details), we decided to employ only the instruct style from the second setting onward due to its little impact on overall performance.

Secondly, we design a testing framework that can uniquely identify a given prompt for a given model and that stores the history of experiments' outcomes.

Lastly, we collect the results on a grid search over multiple models, settings and prompting strategies. The grid search spans across the models, the number of shots $n \in \{0, 1, 3\}$, and the reasoning levels $r \in \{\text{low}, \text{medium}, \text{high}\}$

[1] https://x.ai/news/grok-4 latest access: January 7, 2026

Chat vs. Instruct-based models We experiment with two main classes of prompts: chat and instruct. The chat style is the most common way to prompt a LLM, as most interfaces are designed with this principle in mind. Recent works [14] have inspired the creation of models that perform best when dealing with instructions. Hence, we also experimented with this to compare their effect of legal temporal reasoning.

The instruct prompts selected for LETOV follow the following template:

```
You are a legal expert that never makes mistakes and that
never hallucinates.
Give your unbiased opinion on the following events about
their temporal relationship.
Do not make mistakes.

Consider these examples:

# Example 1
    Given this context: '$example_context1'
    For the statement '$example1'
    You should answer '$label1'

...(other examples or no examples at all)

In the context: $context
Verify the soundness of this statement: $question

Only answer with one word: if the statement is correct,
answer with the word "Entailment"; whereas if the
statement is wrong, answer with the word "Contradiction"
```

The selected LETOV chat prompt, instead, has the following structure.

```
    I am examining this paragraph from a legal context and
    I want to extrapolate the temporal relations between
    two events. I absolutely need these to be correct, no
    mistakes allowed.

    This is my context: $context
    This is my statement: $question

I need a one word answer: if the statement is correct,
answer with the word "Entailment"; whereas if the statement
is wrong, answer with the word "Contradiction"
```

For LETOC we focus on instruct prompts, identifying the following template:

```
    You are a legal expert that never makes mistakes
    and that never hallucinates. Give your unbiased
    opinion on the following events about their temporal
    relationship. You must pick one of three temporal
    relations from a set. Do not make mistakes.

Consider these examples:

# Example 1
    Given this context: '$example_context1'
    For the events:
      Event A: '$example_a1'
      Event B: '$example_b1'
```

You should answer '$label1'

...(other examples or no examples at all)

Question
Given context: $question_context
For the events:
Event A: '$question_a'
Event B: '$question_b'

Only answer with only one word representing the relation:
- If Event A follows event B, answer "follows"
- If Event A precedes event B, answer "precedes"
- If the two events happen at the same time,
answer "simultaneous"

Hardware resources and services We run our experiments using the LLM-As-A-Service OpenRouter platform[2]. The experiments took around 50 hours, and the overall cost was 173,88$.

To prepare the inputs and postprocess the results, we used a machine equipped with 16GB of RAM, an AMD Ryzen AI 7 PRO 350 CPU and 512 GB SSD and running Windows 11 Pro.

Dataset We adapt the LexTime open benchmark [10] to address both the LETOV and LETOC tasks.

LexTime is composed of a legal context taken from U.S. federal complaints between 2020 and 2024. They randomly sampled complaints categorized under the Nature of Suit (NOS) codes beginning with 7, which correspond to labor-related cases. Alongside the context, it contains a statement in natural language about two events. For each statement corresponds a binary label: "entailment" if the statement is sound, "contradiction" otherwise. Each statement also has some metadata about the nature of the couple of events: whether they are explicitly mentioned in the context, or if one of them can only be deduced by a legal expert, eventually marking it as implicit. Our study disregards the effect of metadata as mainly focuses on temporal relations between legal entities.

The dataset curation consisted of the following steps: firstly, we only selected the statements that are logically sound, as it is impossible to deduce the event relation from contradicting statements. Secondly, we used a regular expression to extrapolate each of the temporal relations that compose LexTime. Finally, we aggregate similar ones into three classes:

- *precedes*: for couples of events where the first happens before the second
- *follows*: for couples of events where the first happens after the second
- *simultaneous*: for couples of events where the first and the second happen at the same time.

Hereafter, we will refer to this smaller dataset as the multi-class dataset.

Models We benchmark the performance of the state-of-the-art LLMs reported in Table 1. For each model we also report its reasoning availability and whether or not the reasoning effort specification is supported, the cost expressed in $ per million of output inference tokens and finally if it is an instruct model or not. Opensource models are also reported.

In the experiments we explored the following dimensions of analysis:

- **Model openness**: We compared opensource and proprietary models. We focus on state-of-the-art model, testing a selection of models all released after April 2025.

[2]https://openrouter.ai/ latest access: January 10, 2026

Model Name	Deep Reasoning	Cost ($ per 10^6 tokens)	Instruct
Grok 4 [15]	Yes	15	No
Claude Sonnet 4.5 [16]	Yes	15	No
OpenAI GPT-5.2 [17]	effort specifiable	14	No
OpenAI o3 [18]	effort specifiable	8	No
Gemini 3 flash prev [19]	Yes	3	No
DeepSeek V3.2 [20]	Yes	0.38	No
OpenAI GPT OSS 120b [21]	effort specifiable	Opensource	No
Mistral Devstral 2 2512 [22]	No	Opensource	No
Qwen3 Instruct 2507 [23]	No	Opensource	Yes

Table 1
Selection of textual LLMs, including opensource and proprietary LLMs and LLMs with deep reasoning.

- **Model dimension and context length**: We tested models with context size ranging from 131.072 to 1.048.576. Extending the preliminary work presented in [10] and other works [24] that had already promoted the usefulness of large contexts in legal contexts, we aim to study the impact of very large context length on models' performance.
- **Effect of deep reasoning**: To test the impact of the reasoning capabilities, we consider models with and without this feature (see Section 4 for more details).
- **Instruct vs chat setting**: we compare chat vs. instruct-based LLMs. Given the recent LLMs' alignment to human preferences [14], we explore instruction tuning as an alternative to chat models.

Settings We test three different LEVOT settings. The first one is aimed at discovering the impact of the instruct style prompt, as well as the model's own preference towards a more friendly and conversational prompt or a more strict direct order.

In the second setting we verify whether content adaptation strategies are beneficial to enhance legal temporal reasoning performance. We also empirically verify if the reasoning models are better at generalizing from the examples and therefore applying the reasoning to the question.

In the last setting we try to change the number of tokens that the models can dedicate to reasoning by specifying an effort parameter. The effort parameter can be one of several values. We experiment with values 'low', 'medium', 'high'.

For LEVOC we test only the last two settings of the previous task with slight modifications. Firstly, we test and compare zero-, one-, and three shot learning. Lastly, we once again test how the reasoning effort affects the performance.

4. Experimental results

We measure the LETOV and LETOC performance of different combinations of models and settings in terms of classification accuracy (i.e., the percentage of correctly classified samples, similarly to [10]). Furthermore, we also evaluate the per-class performance in terms of precision, recall and F1-score. For LETOC we adopt the weighted versions of the metric to reduce the impact of class imbalance.

Results and discussion Table 2 reports the values of the performance scores for every run in the instruct style and a differential score Δ. Δ is defined by the performance gap between the classifier prompted with the instruct prompts and the same metric for one prompted with the chat style. For every metric $m \in \{\text{Accuracy}, \text{Precision}, \text{Recall}, \text{F1}\}$,

$$\Delta = m_{\text{instruct}} - m_{\text{chat}}$$

Based on the reported Δ values, the prompting technique appears to provide limited contributions. In addition, as shown by the F1-score results, most models marginally benefit from the instruction prompting style. For this reason, we then further explore the instruction-based LLMs. Qwen 3 Instruct [23] underperforms the large proprietary model, with Grok 4 [15] outperforming the other approach, except for Claude Sonnet 4.5 [16]. Devstral [22] instead achieves a very noticeable 92.37% recall, while getting lower precision scores. The LOVET performance on this task has improved compared to the state of the art(80.8 accuracy) [10].

Model	**Instruct % (Δ)**			
	Accuracy	Precision	Recall	F1
GPT OSS 120b	76.75 (+2.00%)	76.28 (+0.11%)	77.51 (-1.60%)	76.89 (-0.98%)
Mistral Devstral 2	72.75 (-4.20%)	66.28 (-10.37%)	**92.37 (+13.25%)**	77.18 (-0.69%)
DeepSeek v3.2	74.95 (+0.60%)	82.24 (-2.38%)	70.68 (+4.42%)	76.03 (+1.71%)
Claude Sonnet 4.5	82.16 (-0.61%)	84.48 (-1.42%)	78.71 (+0.40%)	81.50 (-0.43%)
Gemini 3	80.76 (=)	79.09 (-1.79%)	83.53 (+2.00%)	81.25 (+0.05%)
Qwen 3 Instruct	73.54 (+3.20%)	72.79 (-5.40%)	82.73 (+6.42%)	77.44 (+0.20%)
GPT-5.2	77.56 (+0.61%)	77.51 (+1.35%)	77.51 (-0.80%)	77.51 (+0.28%)
Grok 4	**83.96 (+0.19%)**	**89.30 (-1.88%)**	77.11 (-13.99%)	**82.76 (+0.64%)**

Table 2
LETOV task. Performance comparison between the chat and instruct-based model versions. The performance gap is quantified by Δ, which is computed as the difference between the results of the instruction and chat settings, respectively.

Table 3 reports the results for LETOC, where we focus on few-shot learning. For the sake of completeness and clarity, we also repeat the zero-shot instruct experiment from the previous setting. Reasoning models are expected to perform better in this task as the reasoning is further helped by the examples. In this setting, Grok 4 [15] proves once again to able to outperform all the other models, though with a limited extent. The performance achieved by Gemini 3 and Sonnet seems to closely follow the one by Grok, though it consistently lags behind. Devstral confirms its tendency to have high recall measures. Overall, the presence of one or few examples helps the model's generalization capabilities as expected. Finally, the top accuracy score of 84.48 achieved by Grok 4 in the three shots, further surpasses the one from the previous setting.

Model	**Zero Shot**				**One Shot**				**Three Shots**			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
Claude Sonnet 4.5	82.16	84.48	78.71	81.50	81.93	84.05	78.63	81.25	84.27	85.96	81.78	83.82
Gemini 3	80.76	79.09	83.53	81.25	**82.73**	80.68	85.88	**83.20**	84.27	83.39	85.42	**84.40**
Mistral Devstral 2	72.75	66.28	**92.37**	77.18	78.11	72.20	**91.12**	80.57	79.63	73.70	**91.90**	81.80
Grok 4	**83.96**	**89.30**	77.11	**82.76**	**82.73**	**88.21**	75.40	81.30	**84.48**	**89.35**	78.14	83.37

Table 3
LETOV task. Comparison between zero-, one-, and few-shot learning settings.

For the final binary classification task's setting, we report the results in Table 4. The effect of the reasoning seems to be limited. However, once more, Grok 4 surpasses the previous score, and we find our best result for the accuracy metric of 85.28. However, while the Grok 4 performance increased steadily, the latency in the response generation is significant, i.e., sometimes exceeding one minute of

Model	**Low**				**Medium**				**High**			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
o3	80.65	85.78	73.28	79.04	77.82	84.42	68.02	75.33	79.84	84.83	72.47	78.17
GPT-5.2	78.63	87.70	66.40	75.58	81.25	**89.69**	70.45	78.91	78.63	86.53	67.61	75.91
Grok 4	**84.07**	**90.78**	**75.71**	**82.56**	**84.48**	88.99	**78.54**	**83.44**	**85.28**	**90.65**	**78.54**	**84.16**

Table 4
LETOC task. Performance results achieved by varying the amount of computing effort and tokens dedicated to reasoning in a three-shot setting.

thinking and generation. This should be taken into account in the cost-benefit analysis.

Table 5 and 6 report the results of the LETOC task. As stated previously, this task is inherently more complex as the number of target classes is higher. Table 5 reports the difference between the zero-shot setting and the settings where the model's context is enriched with examples taken from the original dataset. Models facing this problem generally solved the task well in most cases. However, in this scenario we do not have a clear superior model: only small differences can be noted and the top performance is either shared between two models or it changes with the metric chosen. It is still clear that models that could be chosen in an industrial environment or where performances are of utmost importance are Sonnet 4.5 [16] and Grok 4 [15]. However, the small and opensource Devstral 2 LLM [22] achieves fairly good accuracy, especially in the three-shot setting. Hence, it could be selected for applications where low cost and fast inference is crucial.

Model	**Zero Shot**				**One Shot**				**Three Shots**			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
Claude Sonnet 4.5	**81.97**	**85.68**	**81.97**	**83.28**	**79.31**	84.04	**79.31**	**81.01**	80.00	85.70	80.00	82.14
Gemini 3	77.68	84.62	77.68	79.88	78.88	**84.79**	78.88	80.81	76.09	84.47	76.09	79.09
Mistral Devstral 2	78.541	81.53	78.54	75.26	**79.31**	80.25	**79.31**	76.61	**81.30**	82.06	**81.30**	81.09
Grok-4	78.97	84.07	78.97	80.73	78.88	82.39	78.88	80.20	80.87	**88.10**	80.87	**82.82**
GPT OSS 120b	77.68	79.87	77.68	78.40	75.43	77.05	75.43	76.00	76.96	79.26	76.96	77.89

Table 5
LETOC task. Performance of the instruct-based model tested while varying the number of shots in the LLM prompt.

The last result we want to discuss is once again the variation of the model's reasoning effort. All the measures are reported in table 6. Like in the previous reasoning variation setting, we do not see a steep increase in performance, just a small fluctuation. Once again Grok 4 achieves most of the top performances that we reported in bold. However, GPT 5.2 [17] seems to handle best the medium reasoning effort parameter compared to the others.

Model	**Low**				**Medium**				**High**			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
o3	76.52	82.59	76.52	78.74	78.70	85.04	78.70	80.98	76.96	82.70	76.96	79.02
GPT 5.2	78.26	84.75	78.26	80.44	**79.13**	86.02	**79.13**	**81.55**	79.56	86.70	79.56	81.88
Grok 4	**80.87**	**89.34**	**80.87**	**83.23**	78.26	**86.19**	78.26	80.50	**81.74**	**88.70**	**81.74**	**83.63**

Table 6
LETOC task. Performance of the instruct-based model tested while varying the the amount of tokens the model could use for the reasoning process.

Finally, we analyze the relation between the accuracy of each model (averaged between the various tasks), and the model's cost and context length. Figure 1 visually represents how the cost of the model and its context window length influences its accuracies in the various tasks. The accuracy reported as the dependent variable is macro aggregated using the mean of all accuracies in settings of the binary classification tasks. Those are the instruct zero shot accuracy, the one shot accuracy, the three shot accuracy and the three accuracies of the low-medium-high reasoning effort experiment. We only selected the accuracies of the first task because the number of examples only changes slightly (one to three less runs if the prompt contains examples) and so the mean aggregation method makes sense. As an independent variable, we show how cost and context window length affect the accuracy. If a positive correlation exists between the two variables, we would expect the points to be placed on the main diagonal. However, while this visualization suggests this to be the case for both the cost and the length variables, we can see some notable exceptions like Grok 4 [15] and Gemini 3 [19]. The first one shows a correlation between cost and accuracy but it seems to make the most of its short context length better than the rest. The second one, instead uses very high context lengths, which presumably translates to a higher power consumption, while still remaining quite inexpensive.

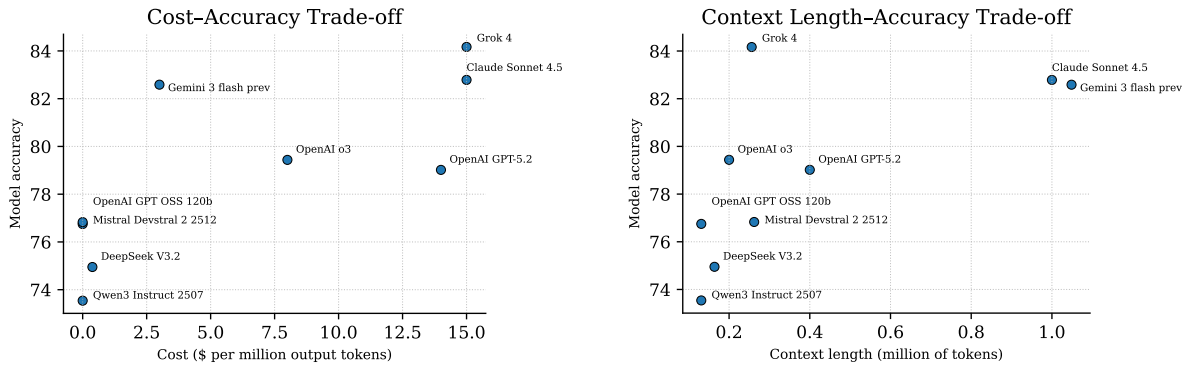


Figure 1: Cost-accuracy (left) and context-accuracy (right) trade-offs scatter plot representations.

Conclusions LLMs have proved to be effective in addressing temporal reasoning on legal documents, particularly in the understanding the temporal order between pairs of legal events. Among the tested models, Grok 4 [15] performs best in both downstream tasks, even in the absence of deep reasoning. As a drawback, the Grok 4's inference time often exceeds one minute, making it not applicable to real-time applications. As an alternative, LLMs like Claude Sonnet 4.5 [16], Gemini 3 [19] and Devstral 2 [22] offer fairly good performance with a more limited cost and inference time.

Future works We plan to extend the set of tested models and configuration settings, including models that are fine-tuned on in-domain sources. We would like to also dig deeper into the reasons behind models' failure by analyzing both the common mistakes and the questions that cause the most failures using Explainable AI techniques. To explore the effect of deep reasoning, we plan to also analyze the structure of the reasoning tokens. Finally, additional prompting techniques that are more specific to the task can be tested as well. For example, we can explain the steps that the model should follow when answering a time related question.

Limitations Due to the limited number of annotated samples, we mainly focus on zero- and few-shot learning rather than supervised fine-tuning. We plan to extend the set of labeled data in the future work.

Some of the LLMs might generate hallucinated content. For this reason, we cannot exclude the generation of unpredictable answers at inference time.

Grok 4 and GPT 5 have inference costs superior to all the other models. Due to budget limitations, we focused on this two very large proprietary LLMs.

Ethics statement

We are not aware of the methods that the providers of the OpenRouter platform employ in terms of data collection and model training. We made sure to disable every option that we could in the settings panel of the website to avoid model training on our queries and all sorts of data collections and we encourage the readers to do so as well. We strongly suggest to only use anonymous data or open source data when and if redoing these experiments and, ideally, we would advise running models on premise if possible.

Data and code availability

The code of the project is publicly available upon request to the authors.

Declaration on Generative AI

During the preparation of this work, the authors used Chat-GPT-5.2 in order to: Grammar and spelling check. After using this tool, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] M. Siino, M. Falco, D. Croce, P. Rosso, Exploring llms applications in law: A literature review on current legal nlp approaches, IEEE Access 13 (2025) 18253–18276. doi:10.1109/ACCESS.2025.3533217.

[2] A. B. Hou, O. Weller, G. Qin, E. Yang, D. Lawrie, N. Holzenberger, A. Blair-Stanek, B. Van Durme, CLERC: A dataset for U. S. legal case retrieval and retrieval-augmented analysis generation, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Findings of the Association for Computational Linguistics: NAACL 2025, Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 7898–7913. URL: https://aclanthology.org/2025.findings-naacl.441/. doi:10.18653/v1/2025.findings-naacl.441.

[3] M. Hindi, L. Mohammed, O. Maaz, A. Alwarafy, Enhancing the precision and interpretability of retrieval-augmented generation (rag) in legal technology: A survey, IEEE Access 13 (2025) 46171–46189. doi:10.1109/ACCESS.2025.3550145.

[4] S. Shaghaghian, L. Y. Feng, B. Jafarpour, N. Pogrebnyakov, Customizing contextualized language models for legal document reviews, in: 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 2139–2148. doi:10.1109/BigData50022.2020.9378201.

[5] I. Benedetto, A. Koudounas, L. Vaiani, E. Pastor, E. Baralis, L. Cagliero, F. Tarasconi, Boosting court judgment prediction and explanation using legal entities, in: Artificial Intelligence and Law, 2024. URL: https://doi.org/10.1007/s10506-024-09397-8. doi:10.18653/v1/2023.semeval-1.194.

[6] P. P. Kumari, G. R. Babu, A survey on legal judgement prediction using machine learning, in: Security Intelligence in the Age of AI: Navigating Legal and Ethical Frameworks, Emerald Publishing Limited, 2025. URL: https://doi.org/10.1108/978-1-83608-156-220251002. doi:10.1108/978-1-83608-156-220251002.

[7] V. Malik, R. Sanjay, S. K. Guha, A. Hazarika, S. K. Nigam, A. Bhattacharya, A. Modi, Semantic segmentation of legal documents via rhetorical roles, in: N. Aletras, I. Chalkidis, L. Barrett, C. Goanță, D. Preoțiuc-Pietro (Eds.), Proceedings of the Natural Legal Language Processing Workshop 2022, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Hybrid), 2022, pp. 153–171. URL: https://aclanthology.org/2022.nllp-1.13/. doi:10.18653/v1/2022.nllp-1.13.

[8] R. Jain, D. Sojitra, A. Acharya, S. Saha, A. Jatowt, S. Dandapat, Do language models have a common sense regarding time? revisiting temporal commonsense reasoning in the era of large language models, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 6750–6774. URL: https://aclanthology.org/2023.emnlp-main.418/. doi:10.18653/v1/2023.emnlp-main.418.

[9] X. Wu, Y. Bu, Y. Cai, T. Wang, Updating large language models' memories with time constraints, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2024, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 13693–13702. URL: https://aclanthology.org/2024.findings-emnlp.801/. doi:10.18653/v1/2024.findings-emnlp.801.

[10] C. Barale, L. Barrett, V. S. Bajaj, M. Rovatsos, Lextime: A benchmark for temporal ordering of legal events, 2025. URL: https://arxiv.org/abs/2506.04041. arXiv:2506.04041.

[11] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, G. Lample, Llama: Open and efficient

foundation language models., CoRR abs/2302.13971 (2023). URL: http://dblp.uni-trier.de/db/journals/corr/corr2302.html#abs-2302-13971.

[12] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, C. Berner, L. Bogdonoff, O. Boiko, M. Boyd, A.-L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann, B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fulford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn, H. Jun, T. Kaftan, Łukasz Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, J. H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, Łukasz Kondraciuk, A. Kondrich, A. Konstantinidis, K. Kosic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Martin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. Mély, A. Nair, R. Nakano, R. Nayak, A. Neelakantan, R. Ngo, H. Noh, L. Ouyang, C. O'Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. de Avila Belbute Peres, M. Petrov, H. P. de Oliveira Pinto, Michael, Pokorny, M. Pokrass, V. H. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. B. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, B. Zoph, Gpt-4 technical report, 2024. URL: https://arxiv.org/abs/2303.08774. arXiv:2303.08774.

[13] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de Las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, W. E. Sayed, Mistral 7b, CoRR abs/2310.06825 (2023). URL: https://doi.org/10.48550/arXiv.2310.06825. doi:10.48550/ARXIV.2310.06825. arXiv:2310.06825.

[14] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, R. Lowe, Training language models to follow instructions with human feedback, 2022. URL: https://arxiv.org/abs/2203.02155. arXiv:2203.02155.

[15] xAI, Grok-4 model card, https://data.x.ai/2025-08-20-grok-4-model-card.pdf, 2024. System card and technical documentation.

[16] Anthropic, Claude sonnet 4.5 system card, https://assets.anthropic.com/m/12f214efcc2f457a/original/Claude-Sonnet-4-5-System-Card.pdf, 2024. Model capabilities and safety report.

[17] OpenAI, Gpt-5.2 system card, https://assets.anthropic.com/m/12f214efcc2f457a/original/Claude-Sonnet-4-5-System-Card.pdf, 2025. Reasoning-effort configurable large language model.

[18] OpenAI, Openai o3 system card, https://cdn.openai.com/pdf/2221c875-02dc-4789-800b-e7758f3722c1/o3-and-o4-mini-system-card.pdf, 2024. Reasoning-oriented model with variable effort.

[19] Google DeepMind, Gemini 3 Flash: Fast and Efficient Multimodal Reasoning, Technical Report, Google DeepMind, 2024. URL: https://storage.googleapis.com/deepmind-media/gemini/gemini_3_flash_model_evaluation.pdf.

[20] DeepSeek-AI, A. Liu, A. Mei, Z. Zhang, Z. Qu, Deepseek-v3.2: Pushing the frontier of open large language models, 2025. URL: https://arxiv.org/abs/2512.02556. arXiv:2512.02556.

[21] OpenAI, S. Agarwal, L. Ahmad, J. Ai, S. Altman, E. Zhang, S. Zhao, gpt-oss-120b gpt-oss-20b model card, 2025. URL: https://arxiv.org/abs/2508.10925. arXiv:2508.10925.

[22] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, W. E. Sayed, Mistral 7b, 2023. arXiv:2310.06825.

[23] A. Yang, A. Li, B. Yang, Beichen, Z. Qiu, Qwen3 technical report, 2025. URL: https://arxiv.org/abs/2505.09388. arXiv:2505.09388.

[24] K. Wei, A. Gautam, R. Huang, Are llms good annotators for discourse-level event relation extraction?, 2025. URL: https://arxiv.org/abs/2407.19568. arXiv:2407.19568.